Co-operative Private Equality Test

Ronghua Li and Chuan-Kun Wu
(Corresponding author: Chuan-Kun Wu)

State Key Laboratory of Information Security, Institute of Software,
Chinese Academy of Sciences, Beijing 100080, P.R. China (Email: {lirh, ckwu}@is.iscas.ac.cn)

(Received June 21, 2005; revised and accepted July 4, 2005)

Abstract

We study the following problem: party A's secret input is a, party B's secret input is b, and party C's input is empty; they want to know if $a = b$ with restriction that A and B should not learn anything more than what is implied by their secret inputs and the comparison result, and C should not learn anything about a or b except if $a = b$. This problem can be seen as a variant of the socialist millionaires' problem. We propose a simple and efficient protocol for this problem from a semantically homomorphic encryption scheme. The protocol is fair if party C is semi-honest.

Keywords: Homomorphic encryption, secure multi-party computation

1 Introduction

Suppose there are three parties A, B, and C. Party A's secret input is a, party B's secret input is b, and party C's input is empty; they want to know if $a = b$ with restriction that A and B should not learn anything more than what is implied by the comparison result, and C should not learn anything about a or b except the equality of a and b. This can be seen as a variant of the socialist millionaires' problem [4] (also known as the private equality test (PET) problem [2]) where two parties each with a secret input want to know if they happen to possess the same secret without disclosing their inputs in case they do not. We call the problem studied in this paper the co-operative private equality test (CPET).

We propose a simple and efficient protocol for the CPET. The protocol makes use of a semantically secure homomorphic encryption scheme (i.e. Paillier's cryptosystem). The protocol requires 3 rounds of communications and calls the encryption primitive constant times. The message complexity is about $3c$ bits, where c is the ciphertext length of the encryption scheme used in the protocol. If the party C is semi-honest, then the protocol is fair for party A and party B.

Although previous protocols [1,7] with a third party for comparison problem can be modified to compute the CPET problem, they cannot achieve the efficiency of the protocol proposed in this paper. The protocols from [1,7] require essentially $\mathcal{O}(l)$ encryptions, and the message complexity is *mathcall* bits, where l is the length of the inputs.

Protocols for comparison problem are used in on-line private bidding and auctions, database query, privacy preserving data mining etc. Our protocol for CPET can be applied to such computations where a third party is used, and the comparison result is accessible to the third party.

2 Preliminaries

For an integer n, $\mathbb{Z}_n$ denotes the set of all integers modulo n. $\mathbb{Z}_n^*$ is a subset of $\mathbb{Z}_n$ such that elements in $\mathbb{Z}_n^*$ are relatively prime to n. An element $a \in \mathbb{Z}_{n^2}^*$ is called a n-th residue modulo n^2 if $x^n \equiv a\,(mod\ n^2)$ for some $x \in \mathbb{Z}_{n^2}^*$.

Computational Indistinguishability

A function $\epsilon(n)$ is called negligible if for every positive polynomial p, and all sufficiently large n, $\epsilon(n) < 1/p(n)$.

We use the notation of computational indistinguishability from [5]. An ensemble $X = \{X_n\}_{n\in N}$ is a sequence of random variables each ranging over binary strings. Two ensembles $X = \{X_n\}$ and $Y = \{Y_n\}$ are said to be computationally indistinguishable if for every PPT algorithm D and every $c > 0$ there exists an integer N such that for all $n > N$

$$|Prob(D(X_n) = 1) - Prob(D(Y_n) = 1)| < 1/n^c.$$

Let $X \approx^c Y$ denote that X and Y are computationally indistinguishable.

Homomorphic Encryption

We use the public-key probabilistic encryption scheme from [6]. Here we just give a brief description of the scheme.

Let p and q be large primes, $n = pq$. Let g be a random element in $\mathbb{Z}_{n^2}^*$ whose order is a non-zero multiple of n. Consider (n, g) as public parameters and (p, q) as

private parameters. Let $\lambda(n) = lcm(p-1, q-1)$. For any $w \in \mathbb{Z}^*_{n^2}$, $L(w^\lambda \, (mod \; n^2)) = \lambda[w]_{1+n} \, (mod \; n)$, where $[w]$ denote the class of w. For $w \in \mathbb{Z}^*_{n^2}$, $n-th$ residuosity class of w with respect to g is the unique integer $x \in \mathbb{Z}_n$, for which there exists $y \in \mathbb{Z}^*_n$ such that $g^x y^n = w \; (mod \; n^2)$. The cryptosystem is described as below.

Encryption E:

plaintext $m < n$
a random integer $r < n$
ciphertext $c = g^m r^n \, (mod \, n^2)$

Decryption D:

ciphertext $c < n^2$
plaintext $m = \frac{L(c^\lambda \, (mod \; n^2))}{L(g^\lambda \, (mod \; n^2))} \, (mod \; n)$

The encryption scheme is semantically secure (see Theorem 5 in [6]). For two known messages m_0, m_1, c is the ciphertext of either m_0 or m_1. Without the private keys, it is computationally intractable to decide whether c is the ciphertext of m_0 or c is the ciphertext of m_1.

The scheme has an additive homomorphic property, i.e. $\forall m_1, m_2 \in \mathbb{Z}_n$ and $k \in \mathbb{N}$,

$$D(E(m_1)E(m_2) \; mod \; n^2) = m_1 + m_2 \; mod \; n,$$

$$D(E(m)^k \; mod \; n^2) = km \; mod \; n.$$

3 Model of Secure Multi-party Computation

3.1 Multi-party Computation

In the setting of secure multi-party computation, n parties P_1, P_2, ..., P_n each with a secret input $x_i (i = 1, ..., n)$ want to evaluate some function $f(x_1, x_2, ..., x_n) = (y_1, y_2, ..., y_n)$. The i-th party P_i gets the i-th output y_i. The function should be computed in a way protecting privacy of their inputs and correctness of the result.

We represent the function computed by the three parties in the multiparty computation model. The goal of the protocol is that at the end of the protocol A and B and C know the same result — if $a = b$. Thus we simply define the function as a single-output one. When the protocol ends, all parties know the value of $f(a, b, \varepsilon)$. The function is described as below.

$$f : \mathcal{A} \times \mathcal{B} \times \mathcal{C} \rightarrow \{0, 1\}$$

where

$$\mathcal{A} = \mathcal{B} = [0, 2^l - 1], \mathcal{C} = \{\varepsilon\},$$

and

$$f(a, b, \varepsilon) = \begin{cases} 1 & if \quad a \neq b \\ 0 & if \quad a = b \end{cases}$$

We aim at designing a protocol for the CPET problem which preserves the following properties.

1) Correctness: all the three parties will be convinced of the correctness of the result;

2) Privacy: neither A or B learns more information about the other party's secret inputs than what is implied by the output of $f(a, b, \varepsilon)$, and C learns the output of $f(a, b, \varepsilon)$ and nothing else;

3) Fairness: if A gets the value of $f(a, b, \varepsilon)$, then the protocol guarantees that B gets it, too.

3.2 Model of Adversary and Communication

Cheating party in protocols is modelled as an adversary running in probabilistic polynomial time (PPT). An adversary may corrupt some party and get all the information held by this party, including all information about all actions and messages the party has received.

An adversary can be either passive or active. A passive adversary obtains the complete information held by the corrupted parties, but the parties still act correctly. An active adversary takes full control of the corrupted parties, the adversary can determine the messages sent out and the outputs produced by the corrupted parties.

An adversary can also be either static or dynamic. A static adversary decides which parties to corrupt before the execution of the protocol. While a dynamic adversary can corrupt parties at any time when the protocol is running.

There are two basic models of communication. In the cryptographic model, the adversary is assumed to have access to all messages sent, however he cannot modify messages exchanged between honest parties. In the information-theoretic model, the parties can communicate via secure channels, and the adversary gets no information about messages exchanged between honest parties. This model is also known as secure channels or private channels model.

In this paper, we assume that parties communicate with one another via secure channels, and the adversary is passive. We assume that party C cannot collude with A or B, so the adversary is static since he can corrupt only one party. That is, either A, or B, or C is cheating passively in the protocol.

3.3 Security Definition

We will consider the security with respect to correctness, privacy, and fairness in the ideal vs. real world paradigm [3]. In the real world the parties run a real protocol and the adversary attacks it. A passive adversary V has access to the internal view of A, or B, or C. At the end of the protocol, the adversary outputs an arbitrary function of its view.

In the ideal world there is a trusted party U, all parties $\overline{A}$, $\overline{B}$, and $\overline{C}$ send their inputs to U. U computes $f(a, b, \epsilon)$ and sends the result to each party. A passive adversary $\overline{V}$ has access to the view of $\overline{A}$, or $\overline{B}$, or $\overline{C}$. All parties output

the value received from U and the adversary outputs an arbitrary function of its view.

Definition 3.1. *A protocol for the CPET problem is secure if for every passive real adversary V there is a passive ideal adversary $\overline{V}$ whose running time is bounded by a polynomial in the running time of V such that for all $a \in \mathcal{A}$, $b \in \mathcal{B}$, the outputs of A, B, C, V in the real model and those of $\overline{A}$, $\overline{B}$, $\overline{C}$, $\overline{V}$ in the ideal process are computationally indistinguishable.*

In particular, A should not gain more information about b than what follows from a and $f(a, b, \epsilon)$; likewise, B should not gain more information about a than what follows from b and $f(a, b, \epsilon)$. C learns $f(a, b, \epsilon)$.

4 Protocol Development

Setup:
Let **S** be a homomorphic public-key cryptosystem as described above. Let k be a security parameter (normally, $k \geq 80$), and $|p| = |q| = k$. If the bit-length l is larger than k, then we just set $k = l$.

Before the execution of the protocol, party C generates a public key / secret key pair (pk, sk) and sends E_{pk} to party A and party B. Party A and party B jointly generate a random number $r_{ab} \in [0, 2^k - 1]$.

Step 1. A encrypts his secret input a as $x = E_{pk}(r_{ab}a)$, and sends x to B.

Step 2. B encrypts his secret input b as $y = E_{pk}(-r_{ab}b)$, and computes $y' = E_{pk}(r_{ab}a)E_{pk}(-r_{ab}b)$, then B sends y' to C.

Step 3. C decrypts y': $z = D_{sk}(y')$. If $z = 0$, then C sends $z' = 0$ to A and B. Otherwise C sends $z' = 1$ to A and B.

Result extraction:
C tests if $z = 0$. If it holds, then C outputs 0 (C concludes $a = b$), otherwise C outputs 1 (C concludes $a \neq b$).

A tests if $z' = 0$. If it holds, then A outputs 0 (A concludes $a = b$), otherwise A outputs 1 (A concludes $a \neq b$). B tests if $z' = 0$. If it holds, then B outputs 0 (B concludes $a = b$), otherwise B outputs 1 (B concludes $a \neq b$).

5 Security Analysis

Theorem 5.1. *Under the security assumption for the public-key cryptosystem S, the protocol above is a secure tri-party protocol for the CPET problem.*

Proof. (Sketch)
Correctness:
Firstly we prove that the protocol is correct.

Party C:
Suppose $a = b$, then C gets

$$\begin{aligned} z &= D_{sk}(y') \\ &= D_{sk}(E_{pk}(r_{ab}a)E_{pk}(-r_{ab}b)) \\ &= D_{sk}(E_{pk}(r_{ab}a - r_{ab}b)) \\ &= D_{sk}(E_{pk}(0)) = 0. \end{aligned}$$

So C knows $a = b$.

Suppose $a \neq b$, then C gets

$$\begin{aligned} z &= D_{sk}(y') \\ &= D_{sk}(E_{pk}(r_{ab}a)E_{pk}(-r_{ab}b)) \\ &= D_{sk}(E_{pk}(r_{ab}a - r_{ab}b)) \\ &= D_{sk}(E_{pk}(r_{ab}(a - b))). \end{aligned}$$

For r_{ab} is a random element, $D_{sk}(E_{pk}(r_{ab}(a - b)))$ is a non-zero random element in $\mathbb{Z}_n$ too. So C knows $a \neq b$.

Party A or B:
In case $a = b$, A (B) gets $z' = 0$, and A (B)knows $a = b$. In case $a \neq b$, A (B) gets $z' \neq 0$, and A (B)knows $a \neq b$.

Thus, in both cases, the three parties A, B, and C will get the same correct result. This completes the proof of correctness of the protocol.

Privacy:
Suppose the protocol is not secure. Then there is a PPT real adversary V such that no corresponding PPT ideal adversary $\overline{V}$ exists that achieves $(a, b, A(a), B(b), C(\epsilon), V) \approx^c (a, b, \overline{A}(a), \overline{B}(b), \overline{C}(\epsilon), \overline{V})$ for all a and b.

We say that A cannot cheat in the protocol. The information held by party A consists of

$$a, r_{ab}, x = E_{pk}(r_{ab}a), f(a, b, \epsilon),$$

from which A can learn if $a = b$ and nothing else.

Party C cannot cheat in the protocol, either. The only information about the secrets a and b held by party C is an encryption of the bit 0 or a random element in $\mathbb{Z}_n$. Thus C only knows the comparison result and nothing else.

In the following, we only analysis the cheating activity of party B.

We model party B as an adversary V who has access to the view of party B. What V knows includes

$$\begin{aligned} b, r_{ab}, x &= E_{pk}(r_{ab}a), \\ y &= E_{pk}(-r_{ab}b), \\ y' &= E_{pk}(r_{ab}a)E_{pk}(-r_{ab}b), f(a, b, \epsilon). \end{aligned}$$

If V can do what an ideal adversary $\overline{V}$ cannot do, that is V can get more information about a than what is implied by the comparison result, then with out lose of generality, V can distinguish the following two cases by observing the view of B. In one case, A's input is a' and B's input

is b, where $a', b \in [0, 2^l - 1]$ and $a' \neq b$. In the other case, B's input is also b, however A's input is a'', where $a'' \in [0, 2^l - 1]$, and $a'' \neq a', a'' \neq b$.

In both cases, B's input and the random number r_{ab} are identical, and $f(a', b, \epsilon) = f(a'', b, \epsilon)$. So V cannot distinguish the two cases from

$$b, r_{ab}, y = E_{pk}(-r_{ab}b), f(a, b, \epsilon).$$

This means V distinguishes the cases only from

$$x = E_{pk}(r_{ab}a),\ y' = E_{pk}(r_{ab}a)E_{pk}(-r_{ab}b).$$

More precisely, V distinguishes the cases only from the encryption of A's input a

$$x = E_{pk}(r_{ab}a).$$

We can construct an adversary D who can break the semantic security of the encryption scheme, using the adversary V. The adversary D runs the protocol, and controls the party A.

The adversary D chooses randomly two messages $a', a'' \in \{0,1\}^l$, and takes $r_{ab}a', r_{ab}a''$ as the two messages in the attack (see the proof of Theorem 5 in [6]) and sends them to the encryption oracle. The encryption oracle answers D with a ciphertext x which is either an encryption of $r_{ab}a'$ or an encryption of $r_{ab}a''$. Then D makes the party A send x to party B, and the parties A, B, and C continue the protocol. The adversary D outputs what the adversary V outputs.

This means, if the adversary V can distinguish whether he is in the protocol with a', b, ϵ as the parties' inputs or the protocol with a'', b, ϵ as the parties' inputs, then the adversary D can distinguish whether the ciphertext x is the encryption of $r_{ab}a'$ or the encryption of $r_{ab}a''$. So we construct an adversary which breaks the semantic security of the encryption scheme, which implies a contradiction to the security assumption of the encryption scheme.

This completes the proof of the Theorem 5.1. □

6 Conclusion

We considered a variant of the private equality test problem. The problem involves three parties A, B, and C. A has a secret a, B has a secret b, and C has an empty input. The three parties want to compare if $a = b$ without leaking anything about a and b more than what is implied by each party's input and the comparison result. We proposed an efficient protocol for this problem based on Paillier's cryptosystem. The protocol requires 3 rounds of communication and calls the encryption primitive constant times. The protocol is fair if the special party C is semi-honest.

References

[1] C. Cachin, "Efficient private bidding and acutions with an oblivious third party," in *6th ACM Conference on Computer and Communications Security*, pp. 120-127, ACM Press, 1990.

[2] R. Fagin, M. Naor, and P. Winkler, "Comparing information without leaking it," *Communications of the ACM*, vol. 39, no. 5, pp. 77-85, 1996.

[3] O. Goldreich, *Secure Multi-party Computation (Working Draft)*, http://citeseer.ist.psu.edu/goldreich98secure.html, 1998.

[4] M. Jakobsson and M. Yung, "Proving without knowing: on oblivious, agnostic and blindfolded provers," in *Advances in Cryptology - Crypto'96*, Springer-Verlag, LNCS 1109, pp. 186-200, 1996.

[5] K. Kurosawa and O. Watanabe, "Computational and statistical indistinguishabilities," in *Proceedings of the 3rd International Symposium on Algorithms and Computation*, LNCS 650, pp. 430-438, 1992.

[6] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology - EUROCRYPT'99*, LNCS 1592, Springer-Verlag, pp. 223-238, 1999.

[7] J. Qin, Z. F. Zhang, D. G. Feng, and B. Li, "A protocol of comparing information without leaking," *Journal of Software*, vol. 15, no. 3, pp. 421-427, 2004.

Ronghua Li born in 1977, as a PH.D student at the State Key Laboratory of Information Security (SKLOIS or LOIS), Graduate School of Chinese Academy of Sciences (GSCAS), Beijing, China. Her research interests are in the area of secure multi-party computation.

Chuan-Kun Wu born in 1964, awarded Bachelor of Science, Master of Science, and PhD of Engineering degrees in 1985, 1988 and 1994 respectively. Since January 1988, he was teaching at Xidian University. He was promoted as a Lecturer in 1990, an Associate professor in 1992, and a full professor in 1995. In September 1995, he became a postdoctoral fellow at Queensland University of Technology in Australia, then from 1997 a research fellow at University of Western Sydney, and from 2000 a Lecturer at Australian National University. From January 2003, he has joined the Institute of Software, Chinese Academy of Sciences. Dr. Wu has been involved in many research projects. He has got many academic awards while he was in China, including 2nd and 3rd prizes for Science and Technology Improvement from China Education Committee, excellent scientific youth honoured by the Department of Mechanics and Electronics of China, and China Government Special Subsidy awarded in 1993. He has served as a Program Chair for 2001, 2002 and 2003 International Workshop on Cryptology and Network Security.

He is now a senior member of IEEE, a member of International Association for Cryptologic Research (IACR). He also serves as an Associate Editor for IEEE COMMUNICATIONS LETTERS. His research has been mainly in the areas of spectrum analysis of cryptographic properties of Boolean functions; applying generalized inverses of matrices to cryptographic design; applying Boolean permutations to cryptosystems; and design and analysis of network security protocols. His current interests are in the area of cryptography and network security. They also include secure electronic commerce and information hiding.