An Efficient Oblivious Transfer Protocol Using Residue Number System

Yanjun Liu [1, 2], Chin-Chen Chang[2, 3], and Shih-Chang Chang[4]
(Corresponding author: Chin-Chen Chang)

School of Computer Science and Technology, Anhui University[1]
No. 3, Feixi Rd., Hefei, 230039, China
Department of Computer Science and Information Engineering, Asia University[2]
No. 500, Lioufeng Rd., Wufeng, Taichung, 41354, Taiwan
Department of Information Engineering and Computer Science, Feng Chia University[3]
No. 100 Wenhwa Rd., Seatwen, Taichung, 40724, Taiwan
Department of Computer Science and Information Engineering, National Chung Cheng University
No.168, Sec. 1, University Rd., Min-Hsiung Township, Chiayi, 62102, Taiwan
(Email: alan3c@gmail.com)
(Received Dec. 20, 2012; revised and accepted Feb. 5, 2013)

Abstract

Because the *t*-out-of-*n* oblivious transfer (OT) protocol can guarantee the privacy of both participants, i.e., the sender and the receiver, it has been used extensively in the study of cryptography. Recently, Chang and Lee presented a robust *t*-out-of-*n* OT protocol based on the Chinese remainder theorem (CRT). In this paper, we use the Aryabhata remainder theorem (ART) to achieve the functionality of a *t*-out-of-*n* OT protocol, which is more efficient than Chang and Lee's mechanism. Analysis showed that our proposed protocol meets the fundamental requirements of a general *t*-out-of-*n* OT protocol. We also utilized BAN logic to prove that our proposed protocol maintains the security when messages are transmitted between the sender and the receiver.

Keywords: Aryabhata remainder theorem (ART), efficiency, oblivious transfer (OT), privacy, security

1 Introduction

Nowadays, the oblivious transfer (OT) protocol has become a significant cryptography tool due to its basic functionality of providing privacy for the involved participants. In 1981, Rabin [21] introduced the first OT protocol which contains two participants, i.e., the sender, Alice, and the receiver, Bob. Alice transmits a bit to Bob, and Bob has a 50% probability of receiving the same bit and a 50% probability of receiving nothing. Alice does not know which of the two choices Bob made.

Many extensions to Rabin's OT protocol have been proposed [2, 4, 12, 13, 14, 15, 17, 20, 23, 25, 28]. Two famous extensions are the one-out-of-two OT protocol [13] and the 1-out-of-*n* OT protocol [2]. The one-out-of-two OT protocol, which is denoted as OT_1^2, allows Alice to transport two bits to Bob, and Bob has a 50% probability of obtaining one of the two bits. Additionally, Alice does not know which bit was obtained by Bob. The 1-out-of-n OT protocol, denoted as OT_1^n, is a significant extension of OT_1^2 in which Alice owns *n* messages, and Bob can receive one of them. Similarly, Alice does not know which message has been received by Bob, and Bob gets no information other than the message that he received. Afterwards, a genuine extension to OT_1^n protocol, which enables Bob to securely obtain *t* out of n messages from Alice, is proposed. Such a protocol is called the t-out-of-n OT protocol, which is denoted as OT_t^n and must satisfy the following three fundamental requirements.

1). Correctness: If both the sender and the receiver follow the OT_t^n protocol, then the receiver can obtain the desired *t* messages after executing the protocol with the sender.

2). Privacy of the receiver: The sender cannot determine which *t* messages were selected by the receiver after conducting the OT_t^n protocol with the receiver.

3). Privacy of the sender: The receiver cannot obtain the other *n-t* messages after conducting the OT_t^n protocol with the sender.

Many OT_t^n protocols have been proposed [1, 7, 11, 16, 24, 26, 27]. In 2003, Mu et al. [19] proposed an OT_t^n protocol based on the concepts of the discrete logarithm problem (DLP) and non-interaction, which means Bob has no need to communicate with Alice during the OT process. They claimed that their OT_t^n mechanism was provably secure and more efficient than previous works.

Unfortunately, in 2009, Chang and Lee [8] pointed out one design weakness in Mu et al.'s protocol. They demonstrated that Mu et al.'s approach cannot ensure the privacy of Alice, i.e., Bob can recover more than t messages from Alice. Hence, Bob could disclose the messages freely that Alice did not want him to know and Alice would not know that he had done so. In order to overcome this drawback, Chang and Lee proposed a robust OT_t^n protocol based on the Chinese remainder theorem (CRT) [5, 6, 9, 10, 18]. They claimed that their OT_t^n scheme meets the basic requirements of a general OT_t^n protocol and also reduces computation overhead with low bandwidth.

Inspired by Chang and Lee's protocol, we propose an efficient OT_t^n protocol based on the Aryabhata remainder theorem (ART) [22]. Because the ART used in our proposed protocol has less time complexity than that of the CRT used in Chang and Lee's protocol, our protocol is more efficient than Chang and Lee's protocol and thus is more practical in applications of cryptography. In addition, a BAN analysis is given to prove that our proposed protocol can maintain the security when messages are transmitted between the sender and the receiver.

The rest of this paper is organized as follows. In Section 2, we briefly review Chang and Lee's protocol and the ART utilized in our OT_t^n protocol. In Section 3, we propose our OT_t^n protocol on the basis of the ART. Section 4 presents analyses of the proposed protocol and the comparisons between the proposed protocol and Chang and Lee's protocol. Finally, the conclusions are made in Section 5.

2 Related Work

First, we review Chang and Lee's protocol [8] and then describe the ART, which is a main construction element of our novel OT_t^n protocol.

2.1 A Review of Chang and Lee's Protocol

In Chang and Lee's OT_t^n protocol, Alice uses *n* positive, pairwise co-prime integers, $q_1, q_2, \ldots, q_n$, and *n* messages to compute an integer *X* by the CRT. Then, she conveys *X* and the information associated with q_i for $1 \le i \le n$ to Bob. Bob communicates with Alice and uses the received information to compute q_j' for *j* = 1, 2, …, *t*, where $q_j' \in \{q_1, q_2, \ldots, q_n\}$. Finally, Bob recovers the *t* messages that he chose by using *X* and q_j' according to the CRT.

Next, we introduce the CRT that is used in Chang and Lee's method. The CRT can be described as follows. Assume that there are *n* positive, pairwise co-prime moduli, $q_1, q_2, \ldots, q_n$, and *n* positive integers, $x_1, x_2, \ldots, x_n$. A number *X* can be represented as $\{x_1, x_2, \ldots, x_n\}$, where $x_i \equiv X \bmod q_i$ for *i* = 1, 2, …, *n*. According to the CRT, the unique solution *X* in Z_Q can be computed as follows, where $Q = \prod_{i=1}^{n} q_i$.

$X = \sum_{i=1}^{n} q_i' \cdot q_i'' \cdot x_i \pmod{Q}$, where $q_i' = Q/q_i$, and $q_i' \cdot q_i'' \equiv 1 \pmod{q_i}$.

Example 2.1 demonstrates how the CRT works.

Example 2.1. Use the CRT to compute a positive integer *X* = $\{x_1, x_2, x_3\}$ = {2, 4, 3} with the moduli set $\{q_1, q_2, q_3\}$ = {5, 7, 13}.

According to the CRT, we can obtain $q_1' = 91$, $q_2' = 65$, $q_3' = 35$, $q_1'' = 1$, $q_2'' = 4$, and $q_3'' = 3$. Therefore,

$$X = \sum_{i=1}^{n} q_i' \cdot q_i'' \cdot x_i \pmod{Q}$$
$$= 91 \cdot 1 \cdot 2 + 65 \cdot 4 \cdot 4 + 35 \cdot 3 \cdot 3 \pmod{455} = 172.$$

2.2 Aryabhata Remainder Theorem

In this subsection, we review the ART [10] used in our proposed protocol. The ART, proposed by Rao and Yang in 2006, has two versions, i.e., ART with two moduli and ART with *n* moduli. Here, we only consider the computation process of the ART with *n* moduli. Assume that there are *n* positive, pairwise co-prime moduli, $q_1, q_2, \ldots, q_n$, and *n* positive integers, $x_1, x_2, \ldots, x_n$. A number *X* can be represented as $\{x_1, x_2, \ldots, x_n\}$, where $x_i \equiv X \bmod q_i$ for *i* = 1, 2, …, *n*. According to the ART, the unique solution *X* that satisfies $0 \le X < \prod_{i=1}^{n} q_i$ can be computed by the iterative algorithm, which is quite different from that stated in the CRT, as shown below.

Input: $(\{x_1, x_2, \ldots, x_n\}, \{q_1, q_2, \ldots, q_n\})$
Output: ***X***
1. $Q_1 \leftarrow 1$, $X_1 \leftarrow x_1$.
2. for *i* = 2 to *n* do
3. $Q_i \leftarrow Q_{i-1} \cdot q_{i-1}$.
4. $X_i \leftarrow Q_i \cdot (((x_i - X_{i-1}) \cdot Q_i^{-1}) \bmod q_i) + X_{i-1}$, where $Q_i^{-1} \bmod q_i$ is the multiplicative inverse of Q_i modulo q_i.
5. end for.
6. Return X_n.

After executing this algorithm, we can get the unique solution X_n. Example 2.2 uses the same values of $\{x_1, x_2, \ldots, x_n\}$ and $\{q_1, q_2, \ldots, q_n\}$ in Example 2.1 to illustrate the computation process of the ART.

Example 2.2. Find a positive integer $X = \{x_1, x_2, x_3\} = \{2, 4, 3\}$ with the moduli set $\{q_1, q_2, q_3\} = \{5, 7, 13\}$ by the ART.

According to the ART, the computation process consists of the three steps shown below:

Step 1:

$Q_1 = 1,\ X_1 = x_1 = 2.$

Step 2:

$Q_2 = Q_1 \cdot q_1 = 1 \cdot 5 = 5.$

$X_2 = Q_2 \cdot (((x_2 - X_1) \cdot Q_2^{-1}) \bmod q_2) + X_1$

$= 5 \cdot (((4-2) \cdot 5^{-1}) \bmod 7) + 2 = 32.$

Step 3:

$Q_3 = Q_2 \cdot q_2 = 5 \cdot 7 = 35.$

$X_3 = Q_3 \cdot (((x_3 - X_2) \cdot Q_3^{-1}) \bmod q_3) + X_2$

$= 35 \cdot (((3-32) \cdot 35^{-1}) \bmod 13) + 32 = 172.$

From the computation process of both the CRT and the ART, we can infer that if the same values of $\{x_1, x_2, \ldots, x_n\}$ and $\{q_1, q_2, \ldots, q_n\}$ are given, the same solution of the number X will be obtained. Hence, the only thing to distinguish the CRT from the ART is the method of computing X. In the CRT, we must compute a modular operation with a large number, Q, in the last step, which will increase the computation time. On the contrary, the ART computes a modular operation with a smaller number in each iteration and thus is more efficient than the CRT [10]. Because the method of computing X dominates the sender's computation cost of an OT_t^n protocol, we can substitute the ART for the CRT to design an OT_t^n protocol with higher efficiency.

3 The Proposed Protocol

In this section, we describe the proposed OT_t^n protocol that is based on the ART. First, we give the notations that are important in the development of our protocol:

- Alice is the sender;
- Bob is the receiver;
- $q_1, q_2, \ldots, q_n$ are n positive integers that are relatively prime in pairs;
- $g_1, g_2, \ldots, g_n$ are the n messages owned by Alice, where the values of m_i are positive integers for i = 1, 2, …, n;
- (e, L) is the public key of Alice, where L is a product of two large prime numbers and $\gcd(e, \phi(L)) = 1$;
- d is the private key of Alice and $ed \equiv 1 (\bmod \phi(L))$;
- ID_i is the identity of the message g_i for i = 1, 2, …, n;
- $a_1, a_2, \ldots, a_n$ are the values published on the bulletin board for Bob to choose the desired messages;
- $g_{k_1}, g_{k_2}, ..., g_{k_t}$ are the t messages required by Bob with the corresponding pair (ID_{k_j}, a_{k_j}) for j = 1, 2, …, t, where $g_{k_j} \in \{g_1, g_2, \ldots, g_n\}$.

Next, we describe our proposed OT_t^n protocol in detail.

Step 1. Bob sends a request for obtaining t messages to Alice.

Step 2. Alice chooses n positive, pairwise co-prime integers, $q_1, q_2, \ldots, q_n$.

Step 3. Alice uses $\{q_1, q_2, \ldots, q_n\}$ and the messages $\{g_1, g_2, \ldots, g_n\}$ that she owns to generate the congruence system $X \equiv g_i (\bmod q_i)$ for i = 1, 2, …, n. Then, Alice calculates the number X by using the ART.

Step 4. Alice uses her public key (e, L) to compute $a_i = q_i^e \bmod L$ for i = 1, 2, …, n.

Step 5. Alice publishes X and $\{(ID_i, a_i)\}_{1 \le i \le n}$ on the bulletin board, from which Bob chooses t pairs of (ID_{k_j}, a_{k_j}) for j = 1, 2, …, t.

Step 6. Bob randomly selects t numbers, $r_1, r_2, \ldots, r_t$, and utilizes Alice's public key (e, L) to compute $m_j = r_j^e \cdot a_{k_j} \bmod L$ for j = 1, 2, …, t. Then, Bob transmits $\{m_1, m_2, \ldots, m_t\}$ to Alice.

Step 7. Alice computes $b_j = m_j^d \bmod L$ for j = 1, 2, …, t by using her private key d and the messages $\{m_1, m_2, \ldots, m_t\}$ sent by Bob, and then transports $\{b_1, b_2, \ldots, b_t\}$ to Bob.

Step 8. Upon receiving $\{b_1, b_2, \ldots, b_t\}$ sent by Alice, Bob generates $q'_j = r_j^{-1} \cdot b_j \bmod L$ for j = 1, 2, …, t.

Step 9. Bob uses X and q'_j for j = 1, 2, …, t to reconstruct the t desired messages by using $g_{k_j} = X \bmod q'_j$ for j = 1, 2, …, t.

4 Analyses

In this section, we prove that our proposed OT_t^n protocol meets the essential requirements of a general OT_t^n protocol, and then give the security and efficiency analyses of our protocol.

4.1 Analysis of the Essential Requirements

Correctness:

Assume that Alice and Bob cannot cheat each other. After Step 8 is completed, Bob computes:

$$q'_j = r_j^{-1} \cdot b_j \bmod L = r_j^{-1} \cdot (m_j^d \bmod L) \bmod L$$
$$= r_j^{-1} \cdot ((r_j^e \cdot a_{k_j} \bmod L)^d \bmod L) \bmod L$$
$$= r_j^{-1} \cdot ((r_j^e \cdot (q_{k_j}^e \bmod L))^d \bmod L) \bmod L$$
$$= (r_j^{-1} \cdot (r_j \cdot q_{k_j})^{ed}) \bmod L = q_{k_j} \bmod L.$$

Since q'_j is equivalent to q_{k_j} for j = 1, 2, …, t, Bob can definitely use X that is sent by Alice and q'_j to reconstruct the t messages that he wanted via the congruence system, i.e., $g_{k_j} \equiv X (\bmod q'_j) \equiv X (\bmod q_{k_j})$ for j = 1, 2, …, t. As a result, the proposed protocol meets this requirement.

Privacy of the receiver:

During Step 7, Alice computes:

$$b_j = m_j^d \bmod L = ((r_j^e \cdot a_{k_j} \bmod L))^d \bmod L$$
$$= (r_j^e \cdot ((q_{k_j}^e \bmod L)) \bmod L)^d \bmod L$$
$$= r_j \cdot q_{k_j} \bmod L.$$

The expression of b_j for j = 1, 2, …, t contains $(r_j \cdot q_{k_j})$, which prevents Alice from getting q_{k_j} separately and using q_{k_j} to obtain the t messages that Bob chose by the equation $g_{k_j} = X \bmod q_{k_j}$ for j = 1, 2, …, t. Hence, the second requirement is met by our proposed protocol.

Privacy of the sender:

We assume that both Alice and Bob are honest. Because Bob cannot get Alice's private key d to calculate $b_j = m_j^d \bmod L$ for $t+1 \le j \le n$, it is impossible for him to compute $q'_j = r_j^{-1} \cdot b_j \bmod L$ and the other n-t messages $g_{k_j} = X (\bmod q'_j)$ for $t+1 \le j \le n$. Therefore, our protocol guarantees the third requirement.

4.2 Security Analysis

This subsection uses BAN logic [3] to verify our OT_t^n protocol. According to the analytical procedures of BAN logic, each round of the protocol must be transformed into the idealized form. Next, we briefly describe basic notations of BAN logic as follows:

- $P \mid\equiv X$: P believes X, or P would be entitled to believe X.
- $P \lhd X$: P sees X. Someone has sent a message containing X to P, who can read and repeat X.
- $P \mid\sim X$: P once said X. The principal P at some time sent a message including the statement X.
- $P \mid\Rightarrow X$: P has jurisdiction over X. The principal P is an authority on X and should be trusted on this matter.
- $\#(X)$: The formula X is fresh. This is usually true for nonce, which includes a timestamp or a random number.
- $P \overset{K}{\leftrightarrow} Q$: P and Q may communicate with each other using the shared key K. The key K will never be discovered by any principal except P or Q.
- $P \overset{X}{\Leftrightarrow} Q$: The formula X is a secret known only to P and Q. Only P and Q may use X to prove their identities to one another.
- $\overset{K}{\mapsto} P$: P has K as a public key. The matching secret key (denoted as K^{-1}) will never be discovered by any principal except P.
- $\{X\}_K$: This represents the formula X encrypted under the key K.
- $\langle X \rangle_Y$: This represents the formula X combined with the formula Y.

The details of our protocol are shown as below. Two messages are used to maintain the security when messages are transmitted between the sender and the receiver in our protocol. Here, we present Alice denoted as A and Bob denoted as B.

Message 1. $B \rightarrow A$: $m_j = r_j^e \cdot a_{k_j} \bmod L$ for $j = 1, 2, ..., t$.

Message 2. $A \rightarrow B$: $b_j = m_j^d \bmod L$ for $j = 1, 2, ..., t$.

Before analyzing our protocol, we first make the following assumptions:

A 1. $B \mid\equiv \overset{e}{\mapsto} A$.

A 2. $B \mid\equiv (A \mid\Rightarrow a_{k_j})$.

A 3. $B \mid\equiv \#(r_j)$.

A 4. $B \mid\equiv (A \mid\Rightarrow L)$.

A 5. $A \mid\equiv (B \mid\Rightarrow \#(r_j))$.

A 6. $A \mid\equiv a_{k_j}$.

A 7. $A \mid\equiv L$.

A 8. $A \mid\equiv \overset{e}{\mapsto} A$.

A 9. $B \mid\equiv (A \mid\Rightarrow e^{-1} = d)$.

Then, we analyze the idealized form of our proposed protocol using the above assumptions and rules of BAN logic. Details of the logic proof are presented as follows.

A receives *Message* 1. The rule shows that

$A \lhd \{ m_j = r_j^e \cdot a_{k_j} \bmod L \text{ for } j = 1, 2, ..., t \}$. (Statement 1)

We break conjunctions and produce

$A \lhd B \mid\sim r_j^e \cdot a_{k_j}$ (Statement 2)

and

$A \lhd B \mid\sim L$. (Statement 3)

By *A 7* and Statement 3, we apply the nonce-verification rule to deduce

$A \mid\equiv L$. (Statement 4)

By *A 6* and Statement 2, we apply the message-meaning rule to derive

$A \mid\equiv B \mid\sim r_j^e$. (Statement 5)

By *A 8* and Statement 5, the message-meaning rule applies and yields

$A \mid\equiv B \mid\sim r_j$. (Statement 6)

By *A 5* and Statement 6, we apply the nonce-verification rule to deduce

$A \mid\equiv r_j$. (Statement 7)

Then, *B* receives *Message 2*. The annotation rule yields that

$B \lhd \{ b_j = m_j^d \bmod L \text{ for } j = 1,2,...,t \}$. (Statement 8)

We break conjunctions and produce as follows:

$B \lhd A \mid\sim m_j^d$ (Statement 9)

and

$B \lhd A \mid\sim L$. (Statement 10)

By *A 4* and Statement 10, we apply the nonce-verification rule to obtain

$B \mid\equiv L$. (Statement 11)

By *A 9* and Statement 9, we apply the message-meaning rule to deduce

$B \mid\equiv A \mid\sim m_j$. (Statement 12)

By *Message* 1, the message-meaning rule applies and yields

$B \mid\equiv A \mid\sim r_j^e \cdot a_{k_j}$. (Statement 13)

By *A 2* and Statement 13, we apply the message-meaning rule to derive

$B \mid\equiv A \mid\sim r_j^e$. (Statement 14)

By *A 1* and Statement 14, the message-meaning rule applies and yields

$B \mid\equiv A \mid\sim r_j$. (Statement 15)

By *A 3* and Statement 15, we apply the nonce-verification rule to deduce

$B \mid\equiv r_j$. (Statement 16)

As the above mentioned, we prove our proposed protocol can keep the security.

4.3 Efficiency Analysis

In this subsection, we analyze the efficiency of our protocol and compare it with Chang and Lee's protocol.

The computation cost of an OT_t^n protocol contains two parts, i.e., the computation cost of the sender and the receiver. Our protocol and Chang and Lee's protocol have the same computation cost of the receiver, but our protocol requires less computation cost for the sender. In our protocol, we substitute the ART for the CRT to compute the number *X*. As discussed in Subsection 2.2, the method to compute *X* dominates the sender's computation cost of an OT_t^n protocol. Therefore, we analyze that the ART used in our protocol requires less time complexity than the CRT used in Chang and Lee's protocol.

According to the CRT, $X = \sum_{i=1}^{n} q_i' \cdot q_i'' \cdot x_i \pmod{Q}$, where $q_i' = Q/q_i$, and $q_i' \cdot q_i'' \equiv 1 \pmod{q_i}$. Here, $q_i' \cdot q_i''$ can be pre-computed. Hence, there are *n* multiplications, (*n*-1) additions, and one modular operation. Assuming that q_i is allocated *s* digits, the multiplication and addition of two moduli require s^2 and *s* bit operations, respectively. Furthermore, an *s*-bit modular operation requires s^2 bit operations. Hence, the computation cost of the CRT is about $n \cdot s^2 + (n-1) \cdot s + (n \cdot s)^2$ bit operations, where $n \cdot s$ is the number of digits in *Q*. Thus, the time complexity of the CRT is $O(n^2 s^2)$.

According to the Aryabhata Remainder Theorem (ART), $X_i = Q_i \cdot (((x_i - X_{i-1}) \cdot Q_i^{-1}) \bmod q_i) + X_{i-1}$, which performs (*n*-1) rounds. Here, $Q_i \cdot (Q_i^{-1} \bmod q_i)$ can be pre-computed. So, there are one multiplication, one subtraction, one addition, and one modular operation in every round. Assume that the division and subtraction of two moduli require s^2 and *s* bit operations, respectively. As a result, after performing (*n*-1) rounds, the computation cost of the ART is about $(n-1) \cdot (s^2 + s + s + s^2)$ bit operations. Therefore, the time complexity of the ART is $O(ns^2)$, which indicates that the ART has lower time complexity than that of the CRT. As a result, our proposed protocol is more efficient than Chang and Lee's protocol.

5 Conclusions and Future Work

In this article, we proposed an OT_t^n protocol that substitutes the ART for the CRT used in Chang and Lee's protocol. Our analysis indicated that our protocol reduces the computation cost, thus is more efficient than Chang and Lee's protocol. We also proved that the security can be kept when messages are transmitted between the sender and the receiver. The generalized Aryabhata remainder theorem (GART) is an extension of the ART in which an additional integer *k* is provided during the computation process. Is there a more efficient OT_t^n method that is based on the GART? This appears to be a useful and interesting issue of inquiry.

Acknowledgments

This study was supported in part by the National Nature Science Foundation of China (grant number: 61202228) and the College Natural Science Key Project of Anhui Province of China (grant number: KJ2012A008). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] B. Aiello, Y. Ishai, and O. Reingold, "Priced oblivious transfer: How to sell digital goods," in *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques*, vol. 2045, pp. 119-135, Aarhus, Denmark, May 2001.

[2] G. Brassard, C. Crepeau, and J. M. Robert, "Information theoretic reductions among disclose problem," in *Proceedings of Symposium on Foundations of Computer Science*, pp. 168-173, Toronto, Canada, Oct.1986.

[3] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *Transactions on Computer Systems*, vol. 8, no. 1, pp. 18-36, 1990.

[4] J. Camenisch, M. Dubovitskaya, and G. Neven, "Oblivious transfer with access control," in *Proceedings of the 16th ACM conference on Computer and communications security*, pp. 131-140, Chicago, USA, Nov. 2009.

[5] C. C. Chang, Y. P. Hsieh, and C. C. Liao, "A visual secret sharing scheme for progressively restoring secrets," *Journal of Electronic Science and Technology*, vol. 9, no. 4, pp. 325-331, 2011.

[6] C. C. Chang and Y. P. Lai, "A fast modular square computing method based on the generalized Chinese remainder theorem for prime module," *Applied Mathematics and Computation*, vol. 161, no. 1, pp. 181-194, 2005.

[7] C. C. Chang and Y. P. Lai, "Efficient *t*-out-of-*n* oblivious transfer schemes," in *Proceedings of the 2008 International Conference on Security Technology*, pp. 3-6, Hainan, China, Dec. 2008.

[8] C. C. Chang and J. S. Lee, "Robust *t*-out-of-*n* oblivious transfer mechanism based on CRT," *Journal of Network and Computer Applications*, vol. 32, no. 1, pp. 226-235, 2009.

[9] C. C. Chang, B. Li, and J. S. Lee, "Secret sharing using visual cryptography," *Journal of Electronic Science and Technology*, vol. 8, no. 4, pp. 289-299, 2010.

[10] H. B. Chen, Y. H. Lai, K. W. Chen, and W. B. Lee, "Enhanced delegation based authentication protocol for secure roaming service with synchronization," *Journal of Electronic Science and Technology*, vol. 9, no. 4, pp. 345-351, 2011.

[11] C. K. Chu and W. G. Tzeng, "Efficient *k*-out-of-*n* oblivious transfer schemes," *Journal of Universal Computer Science*, vol. 14, no. 3, pp. 397-415, 2008.

[12] Y. Z. Ding, "Oblivious transfer in the bounded storage model," in *Proceedings of Advances in Crypto'01*, pp. 155-170, Santa Barbara, USA, Aug. 2001.

[13] S. Even, O. Goldreich, and A. Lempel, "A randomized protocol for signing contracts," *Communications of the ACM*, vol. 28, no.6, pp. 637-647, 1985.

[14] M. Green and S. Hohenberger, "Practical adaptive oblivious transfer from simple assumptions," *Theory of Cryptography*, vol. 6597, pp. 347-363, 2011.

[15] D. Hu and Q. Li, "Asymmetric fingerprinting based on 1-out-of-*n* oblivious transfer," *IEEE communications letters*, vol. 14, no. 5, pp. 453-455, 2010.

[16] A. Jain and C. Har, "A new efficient protocol for *k*-out-of-*n* oblivious transfer," *Cryptologia*, vol. 34, no. 4, pp. 282-290, 2010.

[17] M. Kumar, "A new secure remote user authentication scheme with smart cards," *International Journal of Network Security*, vol. 11, no. 3, pp. 112-118, 2010.

[18] Y. P. Lai and C. C. Chang, "Parallel computational algorithms for generalized Chinese remainder theorem," *Computers and Electrical Engineering*, vol. 29, no. 8, pp. 801-811, 2003.

[19] Y. Mu, J. Zhang, V. Varadharajan, and Y. X. Lin, "Robust non-interactive oblivious transfer," *IEEE Communications Letters*, vol. 7, no. 4, pp. 153-155, 2003.

[20] C. Peikert, V. Vaikuntanathan, and B. Waters, "A framework for efficient and composable oblivious transfer," *Advances in Cryptology*, vol. 5157, pp. 554-571, 2008.

[21] M. O. Rabin, "How to exchange secrets by oblivious transfer," Technical Report TR-81, Aiken Computation Laboratory, Harvard University, May 1981.

[22] T. R. N. Rao and C. H. Yang, "Aryabhata remainder theorem: relevance to public-key crypto-algorithms," *Circuits, Systems, and Signal Processing*, vol. 25, no. 1, pp. 1-15, 2006.

[23] R Srinivasan, V Vaidehi, R Rajaraman, *et al.*, "Secure group key management scheme for multicast networks," *International Journal of Network Security*, vol. 11, no. 1, pp. 30-34, 2010.

[24] T. Tassa, "Generalized oblivious transfer by secret sharing," *Designs, Codes and Cryptography*, vol. 58, no. 1, pp. 11-21, 2011.

[25] W. G. Tzeng, "Efficient 1-out-of-*n* oblivious transfer protocols with universally usable parameter," *IEEE Transactions on Computers*, vol. 53, no. 2, pp. 232-240, 2004.

[26] Q. H. Wu, J. H. Zhang, and Y. M. Wang, "Practical *t*-out-*n* oblivious transfer and its applications," *Information and Communications Security*, vol. 2836, pp. 226-237, 2003.

[27] B. Zeng, C. Tartary, P. Xu, J. Jing, and X. Tang, "A practical framework for *t*-out-of-*n* oblivious transfer with security against covert adversaries," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 465-479, 2012.

[28] F. Zhu, M. W. Mutka, and L. M. Ni, "Private entity authentication for pervasive computing environments," *International Journal of Network Security*, vol. 14, no. 2, pp. 86-100, 2012.

Yanjun Liu was born in Anhui Province, China, in 1982. She received the B.S. degree from Anhui University, Hefei, China, in 2005 and the Ph.D. degree from the University of Science and Technology of China (USTC), Hefei, China, in 2010, both in computer science. She is currently serving in Anhui University. Meanwhile, she is a post doc at Asia University, Taichung, Taiwan. Her current research interests include information security and computer cryptography.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His first degree is Bachelor of Science in Applied Mathematics and master degree is Master of Science in computer and decision sciences. Both were awarded in National Tsing Hua University. Dr. Chang served in National Chung Cheng University from 1989 to 2005. His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from Feb. 2005. Prior to joining Feng Chia University, Professor Chang was an associate professor in Chiao Tung University, professor in National Chung Hsing University, chair professor in National Chung Cheng University. He had also been Visiting Researcher and Visiting Scientist to Tokyo University and Kyoto University, Japan. During his service in Chung Cheng, Professor Chang served as Chairman of the Institute of Computer Science and Information Engineering, Dean of College of Engineering, Provost and then Acting President of Chung Cheng University and Director of Advisory Office in Ministry of Education, Taiwan. Professor Chang has won many research awards and honorary positions by and in prestigious organizations both nationally and internationally. He is currently a Fellow of IEEE and a Fellow of IEE, UK. And since his early years of career development, he consecutively won Outstanding Talent in Information Sciences of the R. O. C., AceR Dragon Award of the Ten Most Outstanding Talents, Outstanding Scholar Award of the R. O. C., Outstanding Engineering Professor Award of the R. O. C., Distinguished Research Awards of National Science Council of the R. O. C., Top Fifteen Scholars in Systems and Software Engineering of the Journal of Systems and Software, and so on. On numerous occasions, he was invited to serve as Visiting Professor, Chair Professor, Honorary Professor, Honorary Director, Honorary Chairman, Distinguished Alumnus, Distinguished Researcher, Research Fellow by universities and research institutes. His current research interests include database design, computer cryptography, image compression and data structures.

Shih-Chang Chang received his B.S. degree in 2005 and his M.S. degree in 2007, both in Department of Information Engineering and Computer Science from Feng Chia University, Taichung, Taiwan. He is currently pursuing his Ph.D. degree in Computer Science and Information Engineering from National Chung Cheng University, Chiayi, Taiwan. His current research interests include electronic commerce, information security, computer cryptography, and mobile communications.