An Efficient Secure Oblivious Transfer

Hung-Min Sun[1], *Yalin Chen[2], Jue-Sam Chou[3]
[1]Dept. Computer Science, National Tsing Hua University
hmsun@cs.nthu.edu.tw
[2]Institute of Information Systems and Applications, National Tsing Hua University
*: corresponding author
d949702@oz.nthu.edu.tw
[3] Department of Information Management, Nanhua University
jschou@mail.nhu.edu.tw

Abstract

As traditional oblivious transfer protocols are treated as a cryptographic primitive, they are usually executed without the consideration of possible attacks, e.g., impersonation, replaying, and man-in-the-middle attacks. Therefore, when these protocols are applied in certain applications such as mental poker playing, some necessary mechanism must be executed first to ensure the security of subsequent communications. But doing this way, we found that almost all of the resulting mechanisms are not efficient enough in communicational cost which is a significant concern for commercial transactions. Inspired by these observations, we propose a novel secure oblivious transfer protocol based on bilinear pairing which not only can provide mutual authentication to resist malicious attacks but also is efficient in communicational cost, other than its original functions.

Keywords*: oblivious transfer, mutual authentication, ID-based cryptosystem, bilinear pairing, communicational cost, ECDLP, DLP, impersonation, MIMA, KCI*

1. Introduction

Oblivious transfer (OT) is an important tool for designing secure protocols and has been widely used in various applications like fairly signing contracts, obliviously searching database, playing mental poker games, privacy-preserving auctions, secure multiparty computations, and so on. It is a protocol by which a sender can send some messages to a receiver without the receiver's knowing which part of the messages can be obtained. In 1981, Rabin [1] first proposed the concept of interactive OT scheme, in which a sender sends an encrypted message to a receiver and the receiver can decrypt the message with probability 1/2. Rabin used the proposed OT to design a protocol, hoping that two parties can exchange their secrets fairly and simultaneously. In his protocol, the initiator signs each message flow to make the message flow be authentic to the receiver. In 1985, Even, Goldreich, and Lempel [2] presented a more generalized form of OT, naming 1-out-of-2 OT (OT_1^2), where a sender sends two

encrypted messages to a chooser with the chooser being able to decrypt one of them that he had chosen. Moreover, by evoking their OT_1^2 multiple times they presented a contract-signing protocol to achieve the goal that one party cannot obtain the other party's contract signature without showing his own. In 1986, Brassard and Crepeau [3] further extended OT_1^2 to 1-out-of-*n* OT (OT_1^n, also known as "all-or-nothing"), the case of sending *n* messages to a chooser with only one of them can be obtained by the chooser. Meanwhile, they pointed out that OT_1^n is a useful cryptographic primitive for implementing a multi-party mental poker game against player coalition. Except for the above interactive versions, Bellare and Micali [4] first proposed a non-interactive OT_1^2 in 1989. It is a means that one user can obliviously transfer something to another who is equipped with two public keys.

Based on interactive and non-interactive OT_1^2 schemes, Naor et al. proposed some related OT works [5-9] during 1999 to 2001 such as, adaptive OT [6], proxy OT [7], distributed OT [8] and how to construct an efficient OT_1^n or OT_k^n from OT_1^2 [5, 9]. Here, the OT_k^n scheme -- a chooser is allowed to privately choose *k* messages from *n* encrypted messages from a sender -- is the last form in OT spectrum. In 2002, Mu et al. [10] proposed three OT_k^n schemes which are directly constructed from RSA encryption, Nyberg-Rueppel signature, and ElGamal encryption scheme respectively, and claimed that their schemes induce a significant improvement in communicational cost. In 2004, Ogata and Kurosawa [11] proposed another OT_k^n scheme (which can be employed in an either adaptive or non-adaptive manner) based on RSA blind signature. They claimed that their scheme can be applied in oblivious key search -- a database supplier first commits *n* data (through a CD-ROM or DVD) to a customer, and the customer can then search a keyword on the pay-per-view basis without revealing the keyword to the database supplier. After that, three OT_k^n schemes were proposed [12-14] in 2005. Among them, Chu et al. claimed that their method [12] is the most efficient one because it needs only 2 rounds to send $1024k$ bits from the chooser to the sender and $1024*(k+1)+n*|\text{Data}|$ bits from the sender to the chooser, where |Data| presents the bit length of Data, a plaintext or ciphertext. In 2006, Parakh [15] proposed an elliptic-curve based algorithm to allow two parties exchanging their secrets in a non-simultaneous manner with one-quarter success probability. Further, they used the proposed algorithm to implement a novel OT_1^2 scheme, having the advantage of providing the same security level with only 160-bit key instead of 1024-bit key needed in RSA. In the same year, for coping with all possible attacks encountered in an open network, Kim et al. [16] modified Bellare-Micali non-interactive OT_1^2 scheme by appending the sender's signature to make the sender undeniable about his sent messages and be authenticated by the chooser. However, we found their protocol still suffers from the impersonation attack

(we will describe this in Section 3). In 2007, Halevi and Kalai [19] proposed another OT_1^2 scheme by using smooth projective hashing and showed that the used RSA-composite in their scheme needs not be a product of safe primes. Also in 2007, Camenish et al. and Green et al. proposed two related studies [17, 18] respectively, both focusing on the security of full simulatability for the sender and receiver which can resist against the selective-failure attack [6]. In 2009, Chang [20] presented an OT_k^n scheme using both the RSA blind signature and Chinese Remainder Theorem; however, we think that their scheme did not consider the overhead of using a public board; i.e. it needs an additional secure-access mechanism to keep malicious users from computing other's secrets on the public board.

After surveying all of the above-mentioned OT schemes, we found that only [1] and [16] does consider protection from all possible attacks. In other words, almost all of the traditional OT schemes lack the consideration of adding security features. Therefore, if we wish such OT protocols to resist against various attacks, we should run them through secure channels. This however would incur extra communicational overhead. For this reason, in this paper, we will propose a novel solution that combines an OT_k^n scheme with a secure mechanism encompassing some security features such as, mutual authentication, the resistance of man-in-the-middle (MIMA) attack, replaying attack, and key compromise impersonation (KCI) attack. We refer to the combination as "authentic OT". Except for the security enhancement, our authentic OT also can improve communicational efficiency. This is because the number of rounds needed by a traditional OT scheme running under a secure channel is at least three (one for the secure mechanism and two for the traditional OT scheme itself), whereas our secured authentic OT scheme just needs two. Thus, as compared with traditional OT schemes, our scheme promotes not only in the aspect of security but also in the communicational efficiency.

The rest of this paper is organized as follows. The introduction has been presented in Section 1 and the preliminaries will be shown in Section 2. In Section 3, we will review Kim et al.'s scheme and show their weaknesses. After that, we show our protocol in Section 4. Then, the security analysis and communicational cost comparisons among the related works are made in Section 5. Finally, a conclusion is given in Section 6.

2. Preliminaries

In this section, we will introduce the security requirements for an authentic OT, the axiom and principles of bilinear paring, and the intractable problems used in this article.

2.1 Security requirements of an authentic OT scheme

In a traditional OT, there are two parties, a sender S (who has n messages m_1, $m_2, ..., m_n$ to be obliviously transferred to a chooser, where $n \geq 2$) and a chooser C (who chooses σ_1, $\sigma_2, ..., \sigma_k$ messages among the transferred messages from the sender in advance, where $\{\sigma_1, ..., \sigma_k\} \subset \{1, 2, ..., n\}$, $k < n$). The protocol should meet the following security requirements:

(1) ***Correctness***: C should obtain the valid data which he had chosen at the end of the protocol if S and C properly run the protocol.

(2) ***Chooser's privacy***: In the protocol, the choices of the chooser should not be known to the sender or other third party. More precisely, the chooser's encrypted choice can be any clear choice with the same probability, i.e. for an encrypted message *y* and any clear message *x*, Pr[*x*|*y*] = Pr[*x*]. This property is known as *Shannon perfect secrecy*.

(3) ***Sender's privacy***: At the end of the protocol run, the chooser cannot get any knowledge about the messages that he did not choose. More formally, the ciphertext sent by the sender is semantically secure (i.e. the ciphertext can not be distinguished non-negligibly better than random guessing).

Except for the above three concerns, to guard against security threats, the following attacks must be considered in our authentic OT scheme.

(1) ***Impersonation attack***: In the protocol, each party must authenticate the counterpart as the intended party as claimed. That is, it should be a mutual-authentication OT.

(2) ***Replaying attack***: An adversary could obtain plaintexts by only replaying old messages that a chooser had sent to a sender before.

(3) ***Man-in-the-middle attack*** (***MIMA***): MIMA is an attack that an adversary slinkingly intercepts the communicational line between two communicating parties and uses some means to make them believe that they are talking to the intended party. But indeed, they each other are talking to the adversary.

(4) ***KCI attack***: KCI attack means that when *S*(*C*)'s private key is compromised by an adversary, the adversary can impersonate *C*(*S*) to communicate with *S*(*C*).

2.2 Bilinear paring

Let G_1 and G_2 be two groups of order q, where q is a large prime, G_1 be a subgroup of the additive group of points on an elliptic curve $\boldsymbol{E}/\boldsymbol{F}_p$, and G_2 be a subgroup of the multiplicative group of a finite field $\boldsymbol{F}_{p^2}^*$. A bilinear mapping is defined as $e: G_1 \times G_1 \to G_2$ between these two groups. The mapping must satisfy the following properties:

(1) Bilinear: A mapping $e: G_1 \times G_1 \to G_2$ is bilinear if $e(aP, bQ) = e(P, Q)^{ab}$ for all P, $Q \in G_1$ and all $a, b \in Z_q^*$.

(2) Non-degenerate: The mapping does not map all pairs in $G_1 \times G_1$ to the identity in G_2.

(3) Computable: There is an efficient algorithm to compute $e(P, Q)$ for any $P, Q \in G_1$.

(4) If P is a generator for G_1 then $e(P, P)$ is a generator for G_2.

(5) Commutative: For all $P_1, P_2 \in G_1$, $e(P_1, P_2) = e(P_2, P_1)$.

(6) Distributive: For all $P_1, P_2, P_3 \in G_1$, $e(P_1 + P_2, P_3) = e(P_1, P_3)e(P_2, P_3)$.

2.3 Diffie-Hellman problems

Let $a, b, c \in_R Z_q^*$ and $G =< g >$, $G_1 =< P >$, and $G_2 =< g\ (= e(P, P)) >$, each be a group of prime order q. In the following, we describe some well known intractable Diffie-Hellman problems that we will use in this paper.

(1) **The Computational Diffie-Hellman (CDH) problem**: CDH problem is that in G, given (g, g^a, g^b), finding the element $C = g^{ab}$.

(2) **The Decisional Diffie-Hellman (DDH) problem**: DDH problem is that in G, given (g, g^a, g^b, g^c), deciding whether $c=ab$.

(3) **The Bilinear Computational Diffie-Hellman (BCDH) problem**: BCDH problem is that given (P, aP, bP, cP) in G_1, finding $e(P, P)^{abc}$ in G_2. According to Boneh and Frank's study [23], the BCDH problem is no harder than the CDH problem in G_1 or G_2.

(4) **Chosen-Target CDH (CTCDH) problem**: Let $H: \{0,1\}^* \to G$ be a hash function, $T(\cdot)$ be a target oracle which returns a random element in G, and $(\cdot)^c$ a helper oracle, where c is a random integer from Z_q^*. Also let q_t be the number of queries for $T(\cdot)$ and q_h the number of queries for $(\cdot)^c$. The CTCDH problem is to find l pairs, $(j_1, v_1), \ldots, (j_l, v_l)$, satisfying $v_i = (T(j_i))^c$, for $1 \le i \le l$ and $q_h < l \le q_t$. Without loss of generality, we let q_h

and q_t be l-1 and l, respectively. The CTCDH problem can then be rephrased as that knowing $T(j_1),\ldots, T(j_l)$ via querying the $T(\cdot)$ oracle, and $(j_1, v_1),\ldots, (j_{l-1}, v_{l-1})$ via querying the helper oracle $(\cdot)^c$, try to find the l^{th} pair (j_l, v_l).

The CTCDH problem is proposed and considered as a hard problem by Boldyreva in 2002 [21]. The former version of this problem in RSA is proved by Bellare et al. in [22].

3. Review of Kim et al.'s protocol

In this section, we first review Kim et al.'s protocol in Section 3.1 then cryptoanalyze the security weaknesses of their protocol in Section 3.2.

3.1 Kim et al.'s protocol

In 2006, Kim et al. proposed a secure verifiable non-interactive OT (NIOT) protocol based on RSA. Their method, improved from Bellare-Micali non-interactive OT, hopes to enable a receiver to authenticate the sender and prevent the sender from denying what he had sent. The protocol is described as follows and also illustrated in Fig 1.

In the initialization phase of the protocol, Bob makes the choice via setting the OT public key (β_0, β_1); i.e. he will obtain m_0 if (β_0, β_1) = (g^x, c/g^x), or m_1 if (β_0, β_1) = (c/g^x, g^x), where c is a public constant. In the oblivious transfer phase, for the security requirement of non-repudiation, Alice would use her private key d_A and Bob's public key (n_B, e_B) to create the credential C_A. Bob then verifies C_A by using his private key d_B and Alice's public key (n_A, e_A). They claimed that a valid credential can make Alice unable to repudiate the fact that she had sent the message.

3.2 Cryptanalysis of Kim's NIOT scheme

Although, Kim et al. claimed that in their scheme, Bob can authenticate Alice and prevent from Alice's denial of what she had sent by checking whether $M_A = (C_A^{\,d_B} \bmod n_B)^{e_A} \bmod n_A$ holds or not. However, the encrypted message, X_A, has never been signed by Alice. Hence, it exposes a serious vulnerability that an adversary E can impersonate Alice to communicate with Bob. We describe this impersonation attack as follows：

(1) When E intercepts X_A, C_A and M_A sent from Alice to Bob, he can compute another couple $(X_0', X_1')(= X_A')$ for a pair of arbitrary messages (m_0', m_1') in the same manner as specified in Fig. 1. Then, he sends X_A', M_A, and C_A to Bob.

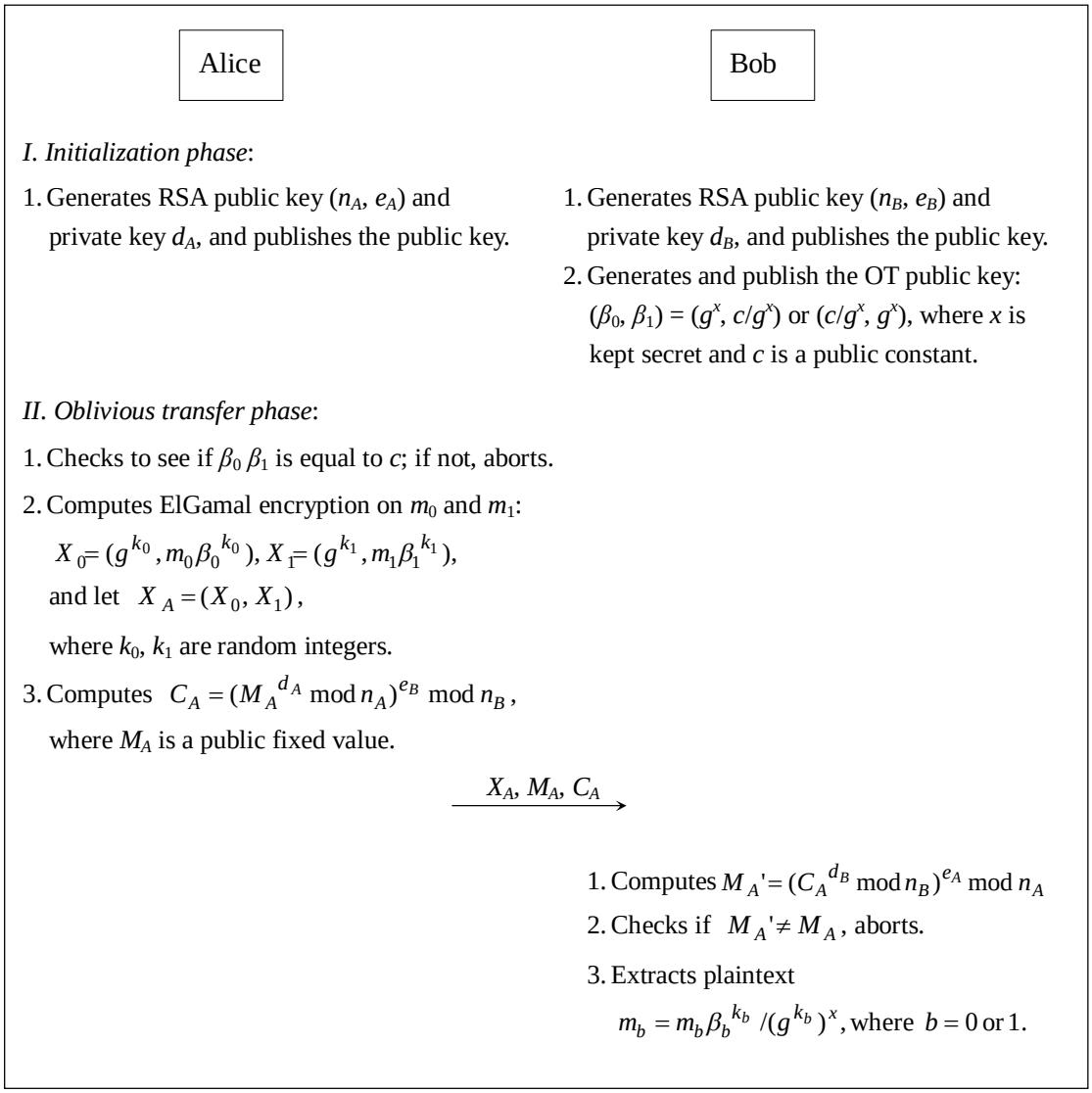

Fig 1: Kim et al.'s scheme

(2) After receiving X_A', M_A, and C_A from E, Bob will accept the received message X_A' since the received M_A is equal to his newly computed $(C_A^{d_B} \bmod n_B)^{e_A} \bmod n_A$. In other words, since the credential C_A is independent of ciphertext X_A' (and even X_A), the adversary E can thus easily impersonate Alice to communicate with Bob.

Moreover, other than the above described drawback, their scheme has another shortcoming. If $n_A > n_B$, then message M_A cannot be recovered by Bob. This is known as the reblocking problem [23] and can make legal Alice can not be authenticated by Bob.

4. Proposed protocol

For the simplicity of key distribution and management, an ID-based public key cryptosystem is often suggested in user identity authentication. In this section, we present our ID-based authentic OT_k^n protocol based on bilinear paring (proposed by Boneh and Franklin in 2001 [24]).

Our scheme also consists of the two phases: (1) initialization phase, and (2) oblivious transfer phase. We describe them as follows.

In phase (1), we adopt the same system parameters as used in [24]. In addition, there exists a key generation center (KGC) who initially chooses an additive group $G_1 =< P >$ of order q, a multiplicative group $G_2 =< e(P,P) >$ of the same order, where e is a bilinear mapping, i.e. $e: G_1 \times G_1 \to G_2$, and three one-way hash functions: $H: \{0,1\}^* \to \{0,1\}^l$, $H_2: G_1 \to \{0,1\}^l$, and H_1 which maps a string (a user's ID) to an element in G_1, i.e. $H_1: \{0,1\}^* \to G_1$. Moreover, KGC also selects $s \in Z_q^*$ as its private master key and computes the corresponding system public key as $P_{pub} = sP$. Then, KGC publishes the system parameter set $\{G_1, G_2, q, e, P, P_{pub}, H, H_1, H_2\}$. After that, when a user U (sender/chooser) registers his identifier ID_U to KGC, KGC can compute a public/private key pair U_{pub}/U_{priv} for him, where $U_{pub}=H_1(ID_U)$ and $U_{priv} = sU_{pub}$.

In phase (2), when a sender who has public/private key pair S_{pub}/S_{priv} and possesses n messages, m_1, m_2,..., and m_n, wants to obliviously transfer k messages of them, m_{σ_1}, m_{σ_2},..., and m_{σ_k}, to a chooser (whose public/private key pair is C_{pub}/C_{priv}), where σ_1, σ_2,..., and σ_k are the k choices selected by the chooser in advance, they will execute the following steps. We also depict them in Fig.2.

Step (1): The chooser randomly chooses two integers a, $b \in Z_q^*$, and computes $V = abC_{pub}$, and $V_j = bH(\sigma_j)C_{priv}$, where j=1, 2,..., k. After that, he generates a signature on V by computing $h = H_2(V)$ and $Sig = hC_{priv}$ Then, he sends ID_R, V, V_1,..., V_k together with Sig to the sender.

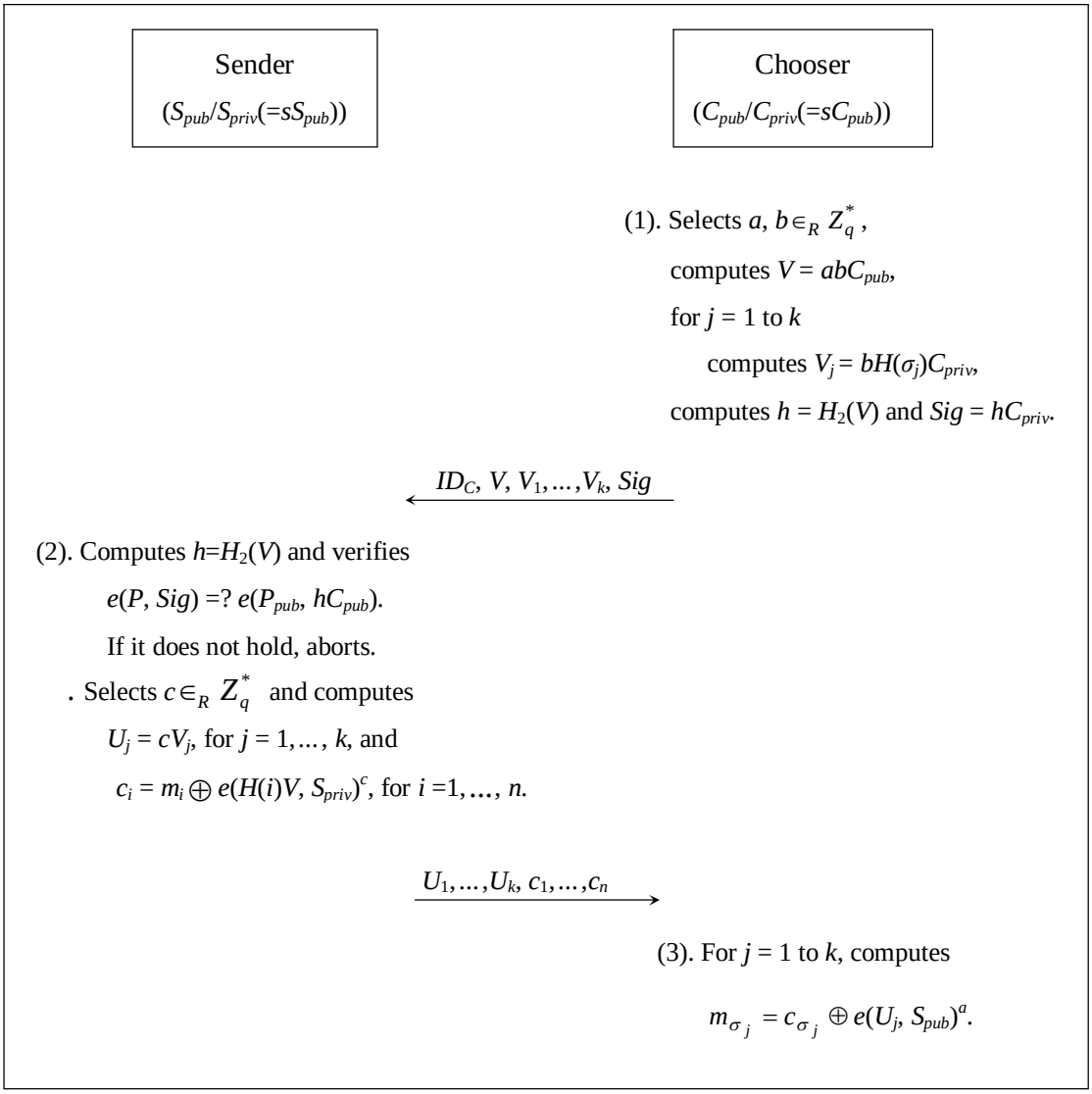

Fig. 2: The proposed k-out-of-n authentic OT protocol

Step (2): After receiving ID_R, V, $V_1, \ldots, V_k$ and Sig from the chooser, the sender computes $h = H_2(V)$ and verifies the chooser's signature by checking whether the equation $e(P,Sig) = e(P_{pub}, hC_{pub})$ holds or not. If it holds, he believes that the chooser is the intended party as claimed. Then, the sender randomly chooses an integer $c \in Z_q^*$, and computes $U_j = cV_j$ and $c_i = m_i \oplus e(H(i)V, S_{priv})^c$, where $j = 1,\ldots,k$ and $i = 1,\ldots,n$. He then sends $U_1, \ldots, U_k, c_1, \ldots,$ and c_n to the chooser.

Step (3): After receiving the message $U_1, \ldots U_k, c_1, \ldots,$ and c_n from the sender, the chooser can obtain the intended plaintexts by computing $m_{\sigma_j} = c_{\sigma_j} \oplus e(U_j, S_{pub})^a$

for $j = 1,...,k$.

5. Security analysis

In this Section, we use the following claims to show that our protocol not only can achieve mutual authentication, chooser's privacy and sender's privacy but also can resist against active attacks such as relay attack, KCI attack and man-in-the-middle attack.

Claim 1: *The proposed protocol is correct.*

Proof: After the protocol run, the chooser can obtain the exact k messages which he selected by computing

$$
\begin{aligned}
& c_{\sigma_j} \oplus e(U_j, S_{pub})^a \\
&= c_{\sigma_j} \oplus e(cbH(\sigma_j)C_{priv}, S_{pub})^a \\
&= c_{\sigma_j} \oplus e(H(\sigma_j)bcsC_{pub}, S_{pub})^a \\
&= c_{\sigma_j} \oplus e(H(\sigma_j)abC_{pub}, sS_{pub})^c \\
&= c_{\sigma_j} \oplus e(H(\sigma_j)V, S_{priv})^c = m_{\sigma_j}.
\end{aligned}
$$

Claim 2: *The proposed protocol can achieve mutual authentication.*

Proof: First, it can be easily seen that the sender can authenticate the chooser by verifying the chooser's signature, *Sig*, as described in Step (2). Next, we show how the chooser can authenticate the sender. For that the ciphertext c_i ($= m_i \oplus e(H(i)V, S_{priv})^c$) contains the sender's private key S_{priv} ($= sS_{pub}$), the chooser can compute the plaintext m_{σ_j} only via using the sender's public key S_{pub} (also refer to the equation in Claim 1). This means that only the true sender can produce the right c_is and thus the chooser can authenticate the sender.

Claim 3: *The proposed protocol can achieve the chooser's privacy.*

Proof: For each of the chooser's choices $\sigma_j \in \{1,2,...,n\}$, it is randomized and signed as $V_j = bH(\sigma_j)C_{priv}$ by C in Step (1), where b is a random number and C_{priv} is the chooser's private key. We argue that nobody except for the chooser can know the choice σ_j because even an attacker can steal the chooser's private key C_{priv}, he cannot obtain $bH(\sigma_j)$ from V_j due to the hardness of ECDLP. That is, he can not figure out $bH(\sigma_j)$, not to mention σ_j. More precisely, let $A = \{(b, \sigma_j) \in Z_q^* Z_n$:

$bH(\sigma_j)C_{priv}= V_j\}$; that is, A consists of all the possible ordered pairs satisfying the equation. If we are given a value V_j, then under the fixed value C_{priv}, there only exists an unique value $bH(\sigma_j)$ satisfying the equation. Under a collision-free one-way hash function, once σ_j has been determined, the value of b is determined as well. That is, the relationship between b and σ_j is one-to-one. Having this observation and the dimension of σ_j is n, we can see that there are n (b, σ_j) pairs in A. In other words, $\Pr[\sigma_j|V_j] = \Pr[\sigma_j] = 1/n$. This achieves the *Shannon perfect secrecy*. Therefore, the proposed protocol has chooser's privacy.

Claim 4: *The proposed scheme can achieve the sender's privacy.*

Proof: Assume that there exists a valid but malicious chooser $\hat{C}$ who plays the role of the chooser wants to obtain more than k plaintexts in the protocol. If he could succeed, then the sender's privacy is violated. However, we will prove that it is computationally infeasible for $\hat{C}$ to obtain the $(k+1)^{th}$ plaintext . We use the following two arguments, (I) and (II), to show that $\hat{C}$ can obtain at most k plaintexts after running the proposed protocol. In argument (I), we discuss that $\hat{C}$ must follow the protocol to form the values V and V_js; otherwise, he can not obtain the k plaintexts that he had chosen. In argument (II), we show that if $\hat{C}$ intends to obtain the $(k+1)^{th}$ plaintext, then he will face the intractable CTCDH problem under the assumption that $H(.)$ is a random hash function.

Argument (I): $\hat{C}$ must follow the protocol to form the values of V $(= ab\hat{C}_{pub})$ and V_j $(= bH(\sigma_j)\hat{C}_{priv})$; otherwise, he can not obtain the k plaintexts, $m_{\sigma_1}, ..., m_{\sigma_j}$, which he had chosen.

In the following, we further divide this argument into two cases: **(a)** $\hat{C}$ fakes V but forms V_js honestly, and **(b)** $\hat{C}$ fakes the values of V and V_js.

(a) If $\hat{C}$ is dishonest in forming V but forms V_js as in the original protocol. For example, without loss of generality, he replaces V with a specified $X \in G_1$ and computes $V_j = bH(\sigma_j)\hat{C}_{priv}$. Then, the sender will compute $U_j = cV_j$, $c_i = m_i \oplus e(H(i)X, S_{priv})^c$, and send them back to $\hat{C}$. As a result, $\hat{C}$ can not decrypt c_{σ_j} $(c_{\sigma_j} = m_{\sigma_j} \oplus e(U_j, S_{pub})^a)$ to obtain the k plaintexts since $e(U_j, S_{pub})^a$ is obviously not equal to $e(H(\sigma_j)X, S_{priv})^c$. For obtaining the k plaintexts, $\hat{C}$ may try another way and compute $e(H(i)X, S_{priv})^c$. But this is computationally infeasible since $\hat{C}$ doesn't know both the sender's private key S_{priv} and one-time

secrecy c. Moreover, to extract c from U_j is an ECDLP.

(b) Similarly, without loss of generality, except for the replacement of V with X, $\hat{C}$ fakes V_j as $H(\sigma_j)X$. Under this construction, the value of U_j computed by the sender would be $U_j = cV_j = cH(\sigma_j)X$ and the target ciphertext c_{σ_j} would be $m_{\sigma_j} \oplus e(H(\sigma_j)X, S_{priv})^c$, or equivalently, $c_{\sigma_j} = m_{\sigma_j} \oplus e(cH(\sigma_j)X, S_{priv})$. Although, $\hat{C}$ knows the value of $cH(\sigma_j)X$ (since it just equal to U_j received from the sender), $\hat{C}$ still can not compute $e(cH(\sigma_j)X, S_{priv})$ without the knowledge of S_{priv}. From above description, we know that when the setting of V_j is $H(\sigma_j)X$, $\hat{C}$ can not obtain m_{σ_j}. Not to mention, $\hat{C}$ sets V_j as $H(\sigma_j)Y$, where $Y(\neq X)$ is an element in G_1. In summary, $\hat{C}$ can not obtain the k plaintexts under the violation of setting the values, V and V_js .

Argument (II): If $\hat{C}$ intends to obtain the $(k+1)^{\text{th}}$ plaintext, then he will face the intractable CTCDH problem under the assumption that $H(.)$ is a random hash function.

That $\hat{C}$ wants to obtain plaintext m_i implies $\hat{C}$ would have the knowledge of $e(H(i)V, S_{priv})^c$ $(=e(U_j, S_{pub})^a)$ (In fact, $\hat{C}$ could know k of the n values, $e(H(i)V, S_{priv})^c$, for i=1 to n, since $e(H(i)V, S_{priv})^c = e(U_j, S_{pub})^a$, for $i = \sigma_j$ and j =1 to k.) Let $e(H(i)V, S_{priv})^c = y^{(i)}$ be an element in G_2. According to argument (I), for obtaining k plaintexts, $\hat{C}$ can not change the structures of $V(=abC_{pub})$ and $V_j(=bH(\sigma_j)\ C_{priv})$. In this situation, $y^{(i)}$ only can be decomposed as $y^{(i)} = e(H(i)abC_{pub}, S_{priv})^c = e(abH(i)C_{priv}, S_{pub})^c$. Moreover, under the assumption that $H(.)$ is a random hash function and the fact that $\hat{C}$ has the knowledge of a, b, C_{priv}, and S_{pub}, $y^{(i)}$ can be represented as $(g_i)^c$, where $g_i = e(abH(i)C_{priv}, S_{pub})$ and is a random element in G_2. Consequently, the problem $\hat{C}$ really faces is that knowing k pairs of $(\sigma_1, (g_{\sigma_1})^c)$, $(\sigma_2, (g_{\sigma_2})^c)$,..., and $(\sigma_k, (g_{\sigma_k})^c)$, where $(g_{\sigma_j})^c = e(U_j, S_{pub})^a$, find the $(k+1)^{\text{th}}$ pair $(\sigma_{k+1}, (g_{\sigma_{k+1}})^c)$ without the knowledge of sender's one-time secrecy c. This is known as the intractable CTCDH problem introduced in Section 2.3. Therefore, the chooser can not obtain the $(k+1)^{\text{th}}$ plaintext; he can just obtain the exact k plaintexts that he had chosen.

According to arguments I and II, we have proven Claim 4 that our scheme has the sender's privacy.

Claim 5: *The proposed scheme can resist against replay attack.*

Proof: Suppose that an adversary eavesdropped on a chooser's OT request (containing ID_C, V, V_js, and Sig) and replayed it later. After receiving the sender's new response ($U_1,...,U_k$, $c_1,...,c_n$) computed from the replayed V and V_js, the adversary can not obtain the selected k plaintexts by computing $m_{\sigma_j} = c_{\sigma_j} \oplus e(U_j, S_{pub})^a$ since he did not know the value of a embedded in the replayed message V. More precisely, it is computationally infeasible for the adversary to extract the value a from the eavesdropped $V = abC_{pub}$ due to the hardness of ECDLP.

Claim 6: *The proposed scheme can resist against man-in-the-middle attack (MIMA).*

Proof: MIMA is an attack that an adversary E slinkingly intercepts the communication line between two communicating parties and uses some means to make them believe that they are talking to the intended party as claimed. But indeed, they each are talking to E. Fig. 3 illustrates the scenario of such a MIMA. We first argue that the adversary E cannot succeed in this scenario since he can not generate the valid message (2), (ID_C, V', V_1',..., V_k', Sig'), in Fig 3. More clearly, he can not forge a valid signature Sig' in message (2) to be verified successfully by the sender without the knowledge of chooser's private key C_{priv}. In addition, it is also hard for E to forge valid message (4), (U_1',..., U_k', c_1',..., c_n'), to be accepted by the chooser. Since that for embedding a meaningful m_i' into c_i', E must have the knowledge of $e(H(i)V, S_{priv})^c$. Though, E can choose another random nonce c' such that $U_j' = c'V_j$, he still has to know the sender's private key S_{priv} to form the valid $c_i'(=m_i \oplus e(H(i)V, S_{priv})^{c'})$. Therefore, without the knowledge of S_{priv}, E can not launch a MIMA attack.

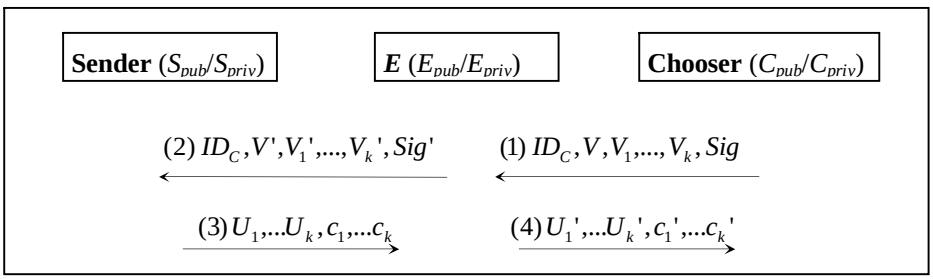

Fig. 3: The scenario of MIMA attack

Claim 7: *The proposed scheme can resist KCI attack.*

Proof: We show this claim by using the following two directions. (i) and (ii).

(i) Suppose that a sender's private key S_{priv} $(=sS_{pub})$ had been compromised by an adversary E and E tries to impersonate chooser C to communicate with the

sender. It can be easily seen that E would fail since that although C can forge V' as abC_{pub}, however, without the chooser's private key, E could not forge V_j' and Sig' to be verified successfully by the sender. Therefore, E could not succeed in this kind of attack.

(ii) Suppose that a chooser's private key C_{priv} (=sC_{pub}) had been compromised by an adversary E and E tries to impersonate sender S to communicate with chooser C. We argue that E would fail in such an attack since he could not know the sender's private key S_{priv} to compute the valid ciphertext c_i.

5.2 Communicational cost comparisons

In this paper, we focus our comparisons on communicational cost of our non-adaptive authentic OT_k^n protocol with the same type of other existing OT_k^n protocols. They are Chu et al.'s [12] (which is, to our best knowledge, the most efficient OT_k^n scheme to date), Mu et al.'s [10], Naor et al.'s [5], and recent works [13, 14, 18, 20]. It is well known that when considering the communicational cost of a protocol, the number of rounds is always a dominant factor while compared with its needed data size in transmission. From this point of view, we know that our scheme is the most efficient for it just requires two rounds, including its security features. Moreover, except for the reduction in number of rounds our scheme makes, we go a step further to analyze the needed size of transmitted messages both from the chooser to the sender and from the sender to the chooser, respectively. Before the comparison, we describe some underlying facts and used notations.

Facts: Generally, for the same security level, a RSA cryptosystem requires key length of 1024 bits while an ElGamal or ECC-based cryptosystem only needs 160 bits. The ciphertext for RSA, ElGamal and ECC-based cryptosystem is 1024 bits, 1024 bits, and 160 bits, correspondingly.

Notations: We use |*string*/*action*| to present the bit length of a *string*, or the needed bit length an *action* performs.

After the description of the facts and notations, we can now use them to estimate the needed transmission data size (NTDS) of our scheme and the above-mentioned protocols. In our protocol, each of the variables V, $V_1, \ldots, V_k$, Sig, $U_1, \ldots, U_k$ transmitted between the chooser and sender is an ECC point. Thus, the NTDS from a chooser to a sender is estimated as $160*(k+2)$ bits and $160k+n*|ciphertext|$ bits from

Table 1: Needed rounds and data size comparisons among protocols

Protocol	Rounds	Size of message: $C \rightarrow S$ (bits)	Size of message: $S \rightarrow C$ (bits)	Authen-tication
Ours	2	$160*(k+2)$	$160k+n*\|ciphertext\|$	yes
Naor et al. [5]	$k*\log n\,\mathrm{OT}_1^2$	depends on OT_1^2	depends on OT_1^2	no
Mu et al.'s scheme(1) [10]	3	$1024k$	$1024n+nk*\|ciphertext\|$	no
Mu et al.'s scheme(2) [10]	2	$1024*2n$	$n*\|ciphertext\|$	no
Chu et al. [12]	2	$1024k$	$1024*(k+1)+ n*\|ciphertext\|$	no
Zhang et al. [13]	2	$1024*(k+3)$	$1024n+ n*\|ciphertext\|$	no
Huang et al. [14]	3	$1024k$	$(n+k)*\|ciphertext\|$	
Green et al. [18]	3	$\|Pok\|+k*\|BlindExtract\|$	$n*\| ciphertext \|+\|Pok\|+ k*\|BlindExtract\|$	no
Chang et al. [20]	4	$1024k$	$(n+2k+2)*1024$	no

the sender to the chooser. Naor et al.'s scheme [5] has the most expensive communicational cost since they construct their OT_k^n scheme by evoking an OT_1^2 primitive $\log n$ times. Thus, the NTDS in their scheme is about $\log n$ times of an OT_1^2's work. As for the non-adaptive scheme of Green et al.'s [18] (in it, there are two types of schemes, adaptive and non-adaptive), the communicational cost is expensive as well due to the complication of the protocol. It is because in the transfer phase, the sender first sends n ciphertexts to the chooser, and then the sender and the chooser together run a proof-of-knowledge (Pok) sub-protocol for assuring the correctness of the ciphertexts. If the proof is valid, the chooser and the sender run the BlindExtract sub-protocol k times for extracting the k blind choices, cooperatively. The chooser can then use the k extracted blind choices to decrypt the ciphertexts. The NTDS of other studies can be estimated in the same manner. We show the comparison results in Table 1.

From Table 1, we can conclude that other than possessing authentication mechanism, our protocol is the most efficient both in communicational cost and NTDS while compared with the other work.

6. Conclusion

An OT scheme which is secure and efficient in both communicational cost and NTDS is essential and eager for commercial applications. After reviewing most of the

OT schemes, we found that it is necessary to combine an OT scheme with some security services to improve the communicational efficiency in certain security needed application environments such as, mental poker playing, oblivious key searching. Therefore, in this paper, we proposed a novel *k*-out-of-*n* authentic oblivious transfer protocol based on bilinear pairing to reach the two goals (security and efficiency). We have proved that our scheme not only can satisfy the mutual authentication, the sender's privacy, and the chooser's privacy but also can resist against replaying, impersonation, MIMA, and KCI attacks. Further, we have compared our scheme with other non-adaptive *k*-out-of-*n* OT schemes in the aspect of needed rounds, NTDS, and authentication. We have shown that our scheme is the most efficient not only in communicational cost but also in data size needed in transmission. In addition, it is the sole solution that has mutual authentication in OT scheme nowadays.

Reference

[1] M. O. Rabin, "How to exchange secrets with oblivious transfer, "*Technical Report TR-81*, Aiken Computation Lab, Harvard University, 1981.

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

[3] G. Brassard, C. Crepeau, and J.-M. Robert, "All-or-nothing disclosure of secrets," *Proc. Advances in Cryptology: CRYPTO'86, LNCS 263*, pp. 234-238, 1986.

[4] M. Bellare and S. Micali, "Non-interactive oblivious transfer and applications," *Proc. of Advances in Cryptology: CRYPTO'89, LNCS 435*, pp.547-557, 1989.

[5] M. Naor and B. Pinkas, "Oblivious transfer and polynomial evaluation," *Pro. of the 31th Annual ACM Symposium on the Theory of Computing (STOC'99)*, pp.245-254, ACM, 1999.

[6] M. Naor and B. Pinkas, "Oblivious transfer with adaptive queries," *Proc. Advances in Cryptology: CRYPTO'99, LNCS 1666*, pp. 573-590, 1999.

[7] M. Naor, B. Pinkas and R. Sumner, "Privacy preserving auctions and mechanism design, " *Proc. of the 1st ACM Conference on Electronic Commerce*, 1999.

[8] M. Naor and B. Pinkas, "Distributed oblivious transfer," *Proc. Advances in Cryptology: ASIACRYPT'00, LNCS 1976*, 2000.

[9] M. Naor and B. Pinkas, "Efficient oblivious transfer protocols," *SODA'01*, pp.448-457, 2001.

[10] Y. Mu, J. Zhang, and V. Varadharajan, "*m* out of *n* oblivious transfer," *Proc. of the 7th Australasian Conference on Information Security and Privacy (ACISP'02), LNCS 2384*, pp. 395-405, 2002.

[11] W. Ogata and K. Kurosawa, "Oblivious keyword search," *Journal of Complexity*,

20(2-3), pp.356-371, 2004.

[12] C. K. Chu, W. G. Tzeng, "Efficient *k*-out-of-*n* oblivious transfer Schemes with adaptive and non-adaptive queries," *PKC 2005, LNCS 3386*, pp. 172-183, 2005.

[13] J. Zhang, Y. Wang, "Two provably secure *k*-out-of-*n* oblivious transfer schemes," *Applied Mathematics and Computation*, vol. 169, pp. 1211-1220, 2005.

[14] H. F. Huang, C. C. Chang, "A new design for efficient *t*-out-*n* oblivious transfer scheme," *Advanced Information Networking and Applications 2, ANIA 2005*, pp. 28-30, 2005**.**

[15] A. Parakh, "Oblivious transfer using elliptic curves," *Proc. of the 15th International Conference on Computing*, IEEE, pp. 323-328, 2006.

[16] S. Kim and G. Lee, "Secure verifiable non-interactive oblivious transfer protocol using RSA and Bit commitment on distributed environment," *Future Generation Computer Systems*, 2006.

[17] J. Camenish, G. Neven, and A. Shelat, "Simulatable adaptive oblivious transfer," *EUROCRYPT 2007, LNCS 4515*, pp. 573-590, 2007.

[18] M. Green, S. Hohenberger, "Blind identity-based encryption and simulatable oblivious transfer," Cryptology ePrint Archive 2007/235, 2007.

[19] S. Halevi, Y. T. Kalai, "Smooth projective hashing and two-message oblivious transfer," *Cryptology ePrint Archive*, 2007/118, 2007.

[20] C. C. Chang and J. S. Lee, "Robust *t*-out-of-*n* oblivious transfer mechanism based on CRT," *Journal of Network and Computer Applications, 32(2009)*, pp.226-235, 2009.

[21] A. Boldyreva, "Threshold signatures, multisignatures and blind signatures based on the gap-diffie-hellman-group signature scheme," *Proc. of the Public-Key Cryptography (PKC'03)*, pp.31-46, 2003.

[22] M. Bellare, C. Namprempre, D. Pointcheval, and M. Semanko, "Power of RSA inversion oracles and the security of Chaum's RSA-based blind signature scheme," *Proceedings of Financial Cryptography (FC'01), LNCS 2248*, pp.319-338, 2001.

[23] L.M. Kohnfelder, "On the signature reblocking problem in public-key cryptography," *Communications of the ACM, vol. 21(2)179*, 1978.

[24] D. Boneh and M.K. Franklin, "Identity-based encryption from the Weil Pairing," *Proc. of Advances in Cryptology: CRYPTO'01, LNCS 2139*, pp. 213-229, 2001.