k out of *n* Oblivious Transfer Protocols from Bilinear Pairings

Qin Jing
School of Mathematics, Shandong University, Jinan, P.R.China
Email: qinjing@sdu.edu.cn

Zhao Hua-wei
School of Computer and Information Engineering, Shandong University of Finance, Jinan,P.R.China
Email: zhaohw@sdfi.edu.cn

***Abstract*—Two non-interactive three parties *k* out of *n* oblivious transfer protocols OT_n^k (where 0<*k*<*n*) from bilinear pairings are proposed in this paper. In these protocols, a sender can obliviously transfer *n* messages to a receiver without communication with him/her. The public keys that a sender used to encrypt his/her messages are made by a third party. The receiver can extract *k* out of *n* messages at his/her choices by using the corresponding secret keys. The choices of the receiver are unconditionally secure. The sender's secrecy is guaranteed if the receiver is semi-honest in the standard model since the decision bilinear Diffie-Hellman problem (DBDH) is hard and the sender's secrecy is achieved when the receiver is a malicious party in the random oracle model since the bilinear Diffie-Hellman problem assumption (BDHA) holds. When a sender is a cheating party, the receiver will detect him/her and halt the protocol. A precise proof of the security of the protocols is presented.**

***Index Terms*—oblivious transfer, malicious oblivious transfer, bilinear pairing, bilinear Diffie-Hellman problems**

I. INTRODUCTION

The concept of oblivious transfer (OT) was first introduced by M. Rabin in 1981[1]. Rabin's OT can be described as a game of two polynomial time parties S (sender) and R (receiver). S sends one secret s (one bit) to R, with 1/2 probability that R will receive s and with 1/2 probability that R will receive nothing; but S does not know which event will happen. Rabin's initiative has attracted a lot of attention. Various OT forms have been subsequently proposed.1-out-2 oblivious transfer protocol OT_2^1 [2], in the protocol, S has two messages m_1 and m_2, and would like R to obtain exactly one of them. In addition, S remains oblivious to R's choice. Non-interactive oblivious transfer protocol[3], in a slightly informal version, a **non-interactive oblivious transfer protocol** is a means that any S can obliviously transfer secret(s) to such a R, who without the recipient's having to take any action at all. 1-out-*n* oblivious transfer protocol OT_n^1 proposed soon after in the name "All-or-nothing disclosure of secrets" [4]. After that, OT_n^1 has become an important research topic in cryptographic protocols design. Some OT_n^1 are designed by invoking basis OT_2^1 schemes several times [5,6]. For *k*-out-*n* oblivious transfer protocol OT_n^k, Bellare and Micali [3] proposed an OT_n^{n-1}, Noar and Pinkas proposed a non-trival OT_n^k protocol [7].The scheme is built by invoking a basis OT_2^1 scheme several times. Chu and Tzeng [8] proposed another OT_n^k scheme based on DDH problem. Mu, Zhang and Varadharajan [9] designed some efficient OT_n^k schemes from cryptographic functions directly. Yao, Bao and Deng [10] pointed out some security issues in [9]. For oblivious transfer protocols, the most efficient one is non-interactive one.

Oblivious transfer protocol has been studied extensively and in many forms. Most of them consider the case that R chooses one message. In this paper, we consider *k*-out-*n* oblivious transfer for strings. In the setting, a sender S holds *n* messages $m_1, m_2, ...m_n$ and is willing to disclose exactly *k* of them to a receiver R. The indices

$$\alpha_1, \alpha_2, ...\alpha_k \in \{1, 2, ..., n\}$$

of the disclose messages $m_{\alpha_1}, m_{\alpha_2}, ...m_{\alpha_k}$ are determined by the receiver and must not be learned by the sender. It is further to be assured that the receiver S does not gain any information about m_j or their combinations, where $1 \le j \le n$ and $j \ne \alpha_1, \alpha_2, ..., \alpha_k$.

Oblivious transfer protocol can be used as stand-alone protocols, e.g. for trading digital information, or as a building block for more complex protocols, e.g. for secure two-party computations, privacy-preserving auctions and oblivious polynomial evaluation. It is used as a key component in many applications of cryptography. It has proven to be a powerful tool in the design of cryptographic protocols; also it is an important

doi:10.4304/jsw.5.1.65-72

sub-protocol of many cryptographic protocols since its introduction by Rabin in 1981 [11,12,13,14,15]. Moreover, oblivious is necessary and sufficient for secure multi-party computations[13, 16].

As a key component in many applications of cryptography, its computational requirements are quite demanding and they are likely to be the bottleneck in many protocols that invoke OT. The analysis of the construction relies on the assumption that determining quadratic residue is hard or decisional Diffie-Hellman problem is hard. In the literature, the sender and the receiver are honest or semi-honest parties (passive parties). Even in the malicious situation, the sender is honest or semi-honest[8, 17, 18]. In the investigation of OT we encountered two problems, one is the OT protocols' efficiency, and the other is the security.

A. Contributions

In this paper we first introduce a three parties OT, where the third party T in an oblivious transfer protocol is same as a Private Key Generator (PKG) in IBE[19], whose sole purpose is to give R a "personalized smart card" as his/her secret key and generate the public key corresponding to the secret key. Using the secret key R can extract k out of n strings that S obliviously transfers with his/her public key. Then we propose two secure and efficient OT_n^k protocols for any $n \geq 2$ from bilinear pairings over elliptic curves. R's secrecy is unconditional. S's secrecy is guaranteed if the decision bilinear Diffie-Hellman problem (DBDH) is hard when R is semi-honest. S's secrecy is achieved in the random oracle model since the bilinear Diffie-Hellman problem assumption (BDHA) holds when R is a malicious party. We do not need any commitment or zero-knowledge proof in the protocols. Therefore we avoid large workload of computations. Our results are along the current trend of research on the design of cryptographic protocols, which is to find provable-secure practical protocols.

We will define malicious oblivious transfer protocol and give two protocols that satisfy this kind of definition. When R is an adversary (passive or active), he/she can not get any information on the not-chosen messages. When S is a cheating party, who bluffs in the protocol, R will detect him/her and stop execution of the protocol. To the best of our knowledge, our methods have not been considered previously in the literature.

B. Outline

The rest of the paper is organized as follows: The next section briefly introduces the definition of OT_n^k, non-interactive OT_n^k and three parties OT, the bilinear maps and the Deffie-Hellamn problem assumptions. Section III will give a detailed description of our non-interactive three parties oblivious transfer protocol OT_n^k and we will analyze the security of our protocols OT_n^k in Section IV, we will give the conclusions of the paper.

II. PRELIMINARIES

This paper focuses on variants of k-out-n oblivious transfer protocol OT_n^k.

A. Parties

The involved parties of an OT_n^k protocol are a sender S and a receiver R. Both are polynomial-time bounded probabilistic parties. A party is semi-honest (or passive) if he/she does not deviate from the steps defined in the protocol, but tries to compute extra information from the received messages. A party is malicious (or active) if he/she can deviate from the specified steps in any way in order to get extra information.

A malicious sender may change the order or contents of his/her messages. We call this kind of malicious sender cheating party.

B. Definition of OT_n^k

In OT_n^k protocol, S has n secrets $m_1, m_2, ...m_n$ and is willing to disclose k of them to R at R's choices $\alpha_1, \alpha_2, ...\alpha_k \in \{1,2,...,n\}$. We say that a protocol is OT_n^k if it satisfies the following three conditions.

1. Correctness : If both S and R follow the protocol, R shall get $m_{\alpha_1}, m_{\alpha_2}, ...m_{\alpha_k}$ after executing the protocol with S, where $\alpha_1, \alpha_2, ...\alpha_k \in \{1,2,...,n\}$ are R's choices.

2. Receiver's security: After executing the protocol with R, S shall not get any information about R's choices $\alpha_1, \alpha_2, ...\alpha_k \in \{1,2,...,n\}$.

3. Sender's security: After executing the protocol with S, R gets no information about other m_j or their combinations, where $1 \leq j \leq n$ and $j \neq \alpha_1, \alpha_2, ..., \alpha_k$.

In fact, OT_n^k is a functionality

$$OT_n^k\left((m_1, m_2, ..., m_n), \alpha_1, \alpha_2, ...\alpha_k\right) = \left(\perp, \left(m_{\alpha_1}, m_{\alpha_2}, ...m_{\alpha_k}\right)\right).$$

The symbol $\perp$ meanings S's outputs is nothing.

A non-interactive oblivious transfer protocol is a means by which any sender S can obliviously transfer messages to such an R who does not take any action at all. A little more formally, non-interactive oblivious transfer protocol is defined as follows: S has n secrets $m_1, m_2, ...m_n$, R has some public keys that S uses to compute messages $c_1, c_2, ..., c_n$ and sends them to R while R does not send any messages to S. R can extract k out of them from these messages at his/her choices. S will not know which of them R got.

A three parties OT is defined as that: there are three parties, a sender, a receiver and a private key generator (PKG). The keys that S used to encrypt his/her messages are made by the third party PKG.

C. Bilinear maps and the Bilinear Diffie-Hellman Assumption

G_1 and G_2 are two groups of order q, where q is some large prime. G_1 is an additive group and G_2 is a multiplicative group. 0 is an identity for G_1 and 1 is an

identity for G_2 . $\hat{e}: G_1 \times G_2 \to G_2$ is a map satisfying the following properties:

1. Bilinearity:

$$\hat{e}\left(P+P^{'},Q\right)=\hat{e}\left(P,Q\right)\hat{e}\left(P^{'},Q\right)$$

and

$$\hat{e}\left(P,Q+Q^{'}\right)=\hat{e}\left(P,Q\right)\hat{e}\left(P,Q^{'}\right)$$

for all P , $P^{'}$, Q , $Q^{'} \in G_1$, especially

$$\hat{e}\left(aP,bQ\right)=\hat{e}\left(P,Q\right)^{ab}$$

for all $P,Q \in G_1$ and all $a,b \in Z$.

2. Non-degeneracy: The maps $\hat{e}$ does not send all pairs in $G_1 \times G_1$ to the identity in G_2 .That is: $\hat{e}\left(P,Q\right) \neq 1$ for some $P,Q \in G_1$. Since G_1 and G_2 are groups of prime order, this implies that if P is a generator of G_1 , then $\hat{e}\left(P,Q\right)$ is a generator of G_2 .

3. Computability: There is an efficient algorithm to compute $\hat{e}\left(P,Q\right)$ for any $P,Q \in G_1$, this algorithm is described in [20]. Its running time is comparable to exponentiation in F_p .

We note that Weil or Tate pairings associated with supers in singular elliptic curves can be modified to create such bilinear maps. (See [20, 21] for more details)

CDH (The computational Diffie-Hellman problem): Given a triple $\left(P,aP,bP\right) \in G_1^3$ for $a,b \in Z_q^*$, find the element abP.

DDH (The decision Diffie-Hellman problem): Given a quadruple $\left(P,aP,bP,abP\right) \in G_1^4$ for $a,b,c \in Z_q^*$, decide whether $c = ab\left(\mathrm{mod}\, q\right)$ or not.

Joux and Nguyen[20]give examples $\hat{e}: G_1 \times G_2 \to G_2$ where CDH in G_1 is believed to be hard but DDH in G_1 is easy. Our protocols are based on variant assumptions of CDH and DDH, called the bilinear Diffie-Hellman problem (BDH) and the decision bilinear Diffie-Hellman problem (DBDH), respectively.

BDH (The Bilinear Diffie-Hellman problem): The problem in $\left\langle G_1,G_2,\hat{e}\right\rangle$ is as follows:

Given $\left(P,aP,bP,cP\right) \in G_1^4$ for some $a,b,c \in Z_q^*$, compute

$$W=\hat{e}\left(P,P\right)^{abc} \in G_2 .$$

G_{en} is called a BDH parameter generator if

1. G_{en} takes a security parameter $\kappa \in Z^+$,
2. G_{en} runs in polynomial time in κ ,
3. G_{en} outputs a prime number q, the description of two groups G_1 and G_2 of order q, and the description of a bilinear map $\hat{e}$ described above.

BDHA (The bilinear Diffie-Hellman Assumption): Let G_{en} be a BDH parameter generator. We say that G_{en} satisfies the BDHA if for any randomized polynomial time (in sufficiently large κ) algorithm A that

$$Adv_{G_{en},A}\left(k\right)=\Pr[A\left(q,G_1,G_2,\hat{e},P,aP,bP,cP\right)=\hat{e}\left(P,P\right)^{abc}:$$
$$\left\langle q,G_1,G_2,\hat{e}\right\rangle \leftarrow G_{en}\left(1^k\right),P \leftarrow G_1^*,a,b,c \to Z_q^*]$$

is a negligible function. When G_{en} satisfies the BDHA we say that BDH is hard in groups generated by G_{en} . (See [19] for more details).

DBDH (The decision of bilinear Diffie-Hellman problem): Given $\left(P,aP,bP,cP,h^{\omega}\right)$ decide whether $abc \equiv \omega \bmod q$, where $h=\hat{e}\left(P,P\right)$, $a,b,c \in Z_q^*$. We say DBDH problem is hard if it is not possible to distinguish the following two distribution ensembles with a non-negligible advantage in polynomial time[22]:

$$Y_1=\left\{\left(P,aP,bP,cP,h^{abc}\right),h=\hat{e}\left(P,P\right),a,b,c \in Z_q^*\right\}$$

and

$$Y_2=\left\{\left(P,aP,bP,cP,h^{\omega}\right),h=\hat{e}\left(P,P\right),a,b,c \in Z_q^*\right\}$$

where the advantage is that

$$Adv_{G_{en},A}\left(k\right)=|\Pr[\,A\left(q,G_1,G_2,\hat{e},P,aP,bP,cP,h^{abc}\right)=1:$$
$$\left\langle q,G_1,G_2,\hat{e}\right\rangle \leftarrow G_{en}\left(1^k\right),P \leftarrow G_1^*,a,b,c \to Z_q^*]-$$
$$\Pr[A\left(q,G_1,G_2,\hat{e},P,aP,bP,cP,h^{\omega}\right)=1:$$
$$\left\langle q,G_1,G_2,\hat{e}\right\rangle \leftarrow G_{en}\left(1^k\right),P \leftarrow G_1^*,a,b,c \to Z_q^*]|$$

D. Security Model

In order to define security we discuss separately protecting the sender S and the receiver R. Since we can not offer both S and R unconditional protection, we provide only computational protection for one of them. S is protected computationally and R unconditionally. Only honest third party T is considered in this paper. In $OT_n^k - I$ we suppose S is a semi-honest party. In $OT_n^k - II$, we suppose S is a cheating party. In the literature, to prevent the cheat, S is often asked to commit the messages in a bulletin. When S sends the encrypted messages to the receiver during the execution of an OT protocol, he/she needs to tag a zero-knowledge proof of showing equality of committed messages and the encrypted messages.

In $OT_n^k - II$, when S can bluff in the protocol R will detect him/her and stop execution of the protocol without commitment and zero-knowledge proof. In both $OT_n^k - I$ and $OT_n^k - II$ R is semi-honest or a malicious party.

An OT_n^k with security against semi-honest S and R should meet following requirements:

R's security: If for any two different sets of choices $\alpha = \{\alpha_1,\alpha_2,...,\alpha_k\}$ and $\alpha' = \{\alpha_1',\alpha_2',...,\alpha_k'\}$ and any probabilistic polynomial time S^* executing the sender's part. The message that S^* receives in the case where R tries to obtain $\left\{m_{\alpha_1},m_{\alpha_2},...,m_{\alpha_k}\right\}$ and $\left\{m_{\alpha_1'},m_{\alpha_2'},...,m_{\alpha_k'}\right\}$ are identically distributed, the choices of R are unconditionally secure.

S's security: If for any choices set $\alpha = \{\alpha_1,\alpha_2,...,\alpha_k\}$, the encryption of the un-chosen messages are indistinguishable from the random ones if

DBDH is hard, the un-chosen messages of S are computationally secure.

An OT_n^k with security against semi-honest S and malicious R should meet the following requirements:

R's security: Same as the case of the semi-honest receiver.

S's security: Comparison with the ideal-model. In the ideal-model, the sender sends all messages and the receiver sends his/her choices to the trusted third party TTP. Then TTP sends the chosen messages to the receiver. The receiver can not get extra information from the sender in the ideal-model and the sender can not know receiver's choices. We say that the sender S's security is achieved if for any receiver R in the real OT_n^k, there is another PPTM R^* (called simulator) in the ideal-model such that the outputs of R and R^* are indistinguishable if BDHA holds.

III. NON-INTERACTIVE THREE PARTIES OBLIVIOUS TRANSFER PROTOCOLS

In this section, we will give the details of two non-interactive three parties oblivious transfer protocols OT_n^k from bilinear pairings. A sender S obliviously transfers n messages to a receiver R. R can extract k out of n messages at his/her choices by using his/her private keys without communication with S. One important typical feature of our schemes is that the keys which S uses to encrypt the messages are made by the third party T. Thus we have improved the sender S's security and we give detailed proof of this claim. Another important typical feature of our schemes is that we give an OT_n^k protocol when the sender S is a cheating party.

A. $OT_n^k - I$

We suppose that S is a semi-honest party, R is a semi-honest party or a malicious party.

Input: S's input are $m_1, m_2, ..., m_n \in \{0,1\}^l$. R's input are $\alpha_1, \alpha_2, ..., \alpha_k \in \{1,2,...,n\}$

Output: R's output are $m_{\alpha_1}, m_{\alpha_2}, ..., m_{\alpha_k}$, that is he/she gets the information of his/her choices. S's output is nothing.

Initialization: The third party T, as a Private Key Generator (PKG) in IBE [19], using the security parameter κ, generates G_1, G_2 of order of q, G_1 is an additive group and G_2 is a multiplicative group, and a bilinear map: $\hat{e}: G_1 \times G_2 \to G_2$. Choose a cryptographic hash function H_1: $G_2 \to \{0,1\}^l$. T picks random $s \in Z_q^*$ and sets $Q = sP$, where $P \in G_1$ is a random generator. We call s master-key.

Key Generation: R picks his/her choices

$$\alpha_1, \alpha_2, ..., \alpha_k \in \{1,2,...,n\}$$

and finds a bi-jection

$$f: i \to x_i' = f(i) \in Z_q^*, i = 1,2,...,n .$$

R sends $x_i' (i = 1,2,...,n)$ and $\alpha_1, \alpha_2, ..., \alpha_k \in \{1,2,...,n\}$ to T. For every $x_i' \in Z_q^*$, T chooses $x_i'' \in Z_q^*$ and let $x_i = x_i' + x_i''$, $x_1, x_2, ..., x_n$ are distinct, then computes

$$P_{pub_i} = x_i P, i = 1,2,...,n .$$

T sends $x_{\alpha_j} (j = 1,2,...,k)$ to R as his/her private keys.

$$q, G_1, G_2, \hat{e}, l, H_1, P, Q, P_{pub_i} (i = 1,2,...,n)$$

are public.

Protocol:

Step1: S chooses random $r_i \in Z_q^* (i = 1,2,...,n)$ and sets the cipher-texts to be

$$c_i = \langle u_i, v_i \rangle$$

For $i = 1,2,...,n$, where

$$u_i = r_i P; v_i = m_i \oplus H_1(g_1^{r_i}) \text{ and } g_i = \hat{e}(Q, P_{pub_i}) \in G_2^* .$$

Then S sends $c_i (i = 1,2,...,n)$ to R.

Step2: On receiving $c_1, c_2, ..., c_n$, R uses his/her secret keys $j = 1,2,...,k$, to compute

$$v_{\alpha_j} \oplus H_1(\hat{e}(u_{\alpha_j}, x_{\alpha_j} Q)) .$$

He/she then gets m_{α_j}, $j = 1,2,...,k$, the secrets at his/her choices.

B. Analysis of $OT_n^k - I$

Correctness : For $j = 1,2,...,k$, we have

$$\hat{e}(u_{\alpha_j}, x_{\alpha_j} Q) = \hat{e}(r_{\alpha_j} P, x_{\alpha_j} sP) = \hat{e}(Q, P_{pub_i})^{r_{\alpha_j}} = g_{\alpha_j}^{r_{\alpha_j}} .$$

Therefore

$$v_{\alpha_j} \oplus H_1(\hat{e}(u_{\alpha_j}, x_{\alpha_j} Q)) =$$

$$m_{\alpha_j} \oplus H_1(g_{\alpha_j}^{r_{\alpha_j}}) \oplus H_1(\hat{e}(u_{\alpha_j}, x_{\alpha_j} Q)) = m_{\alpha_j}$$

R's Security: Note that only honest third part T is considered. For every tuple $(x_1', x_2', ..., x_k')$, corresponding to the choices $\alpha_1', \alpha_2', ..., \alpha_k'$, there is a tuple $(x_1'', x_2'', ..., x_k'')$ that satisfies $P_{pub_{\alpha_j}} = (x_i' + x_j'')P$ for $i = 1,2,...,k$. So the messages that S^* receives in the case R tries to obtain $\{m_{\alpha_1}, m_{\alpha_2}, ..., m_{\alpha_k}\}$ and $\{m_{\alpha_1'}, m_{\alpha_2'}, ..., m_{\alpha_k'}\}$ are identically distributed, thus R's choices $\alpha_1, \alpha_2, ...\alpha_k$ are computationally secure even though S has unlimited computing power.

S's security: For almost OT or OT_n^k schemes[1,3, 6,7,8,17,18], the key that S used to encrypt his/her messages was made by R himself/herself. Therefore if R is a malicious party, he/she may get more information from the cipher-texts S sent to him/her. So S's security is trustless. In our protocol, the key that S used to encrypt his/her messages was made by the third party T, so we enhance the degree of S's security.

About the $OT_n^k - I$, when R is a semi-honest party we have:

Theorem 3.1: $OT_n^k - I$ meets the sender's security requirement. That is, by the DBDH assumptions, if R is semi-honest, he/she shall obtain no information about m_j ($1 \le j \le n$ and $j \ne \alpha_1, \alpha_2, ..., \alpha_k$).

Proof of Theorem 3.1: We will show that c_j for all $j \in \{1,2,...,n\} \setminus \{\alpha_1, \alpha_2, ..., \alpha_k\}$ look random if DBDH is hard.

First we define the random variables for the encryptions of the un-chosen messages as

$$c_j = (u_j, v_j), j \in \{1,2,...,n\} \setminus \{\alpha_1, \alpha_2, ..., \alpha_k\},$$

Since

$$u_j = r_j P, v_j = m_j \oplus H_1\left(\hat{e}\left(Q, P_{pub_j}\right)^{r_j}\right)$$

are known by R. $H_1, P, Q = sP, P_{pub_j} = x_j P$ are public, so we simplify c_j as $c_j' = \left(P, a_j P, b_j P, cP, h^{a_j b_j c}\right)$ where $h = \hat{e}(P,P)$, $a_j, b_j, c \in Z_q^*$. We just need to show that if the following two distributions

$$c_j' = \left(P, a_j P, b_j P, cP, h^{a_j b_j c}\right) \text{ and } X = \left(P, P_1, P_2, P_3, W\right)$$

where $P_1, P_2, P_3 \in G_1^*, W \in G_2$ are distinguishable by a polynomial-time distinguisher D, we can construct another polynomial-time machine D', which takes D as a procedure, to solve the DBDH problem.

D':

Input : (P, P_1, P_2, P_3, W) (Which is either from Y_1 or Y_2).

If $W \ne 1$, let $W = h^w$; otherwise, output 1.

Feed (P, aP, bP, cP, W) to D. $D(P, aP, bP, cP, W) = 1$ if and only if output 1.

We can see that if

$$(P, P_1, P_2, P_3, W) = \left(P, aP, bP, cP, h^{abc}\right)$$

is from Y_1, (P, aP, bP, cP, W) has the right form from c_j'.

If

$$(P, P_1, P_2, P_3, W) = \left(P, aP, bP, cP, h^{\omega}\right)$$

is from Y_2, (P, aP, bP, cP, W) has the right form from X.

Therefore, if D distinguishes c_j' and X with non-negligible advantage ε, then D' can solve the DBDH problem at least non-negligible advantage $\varepsilon - 1/q$.

Communication complexity of $OT_n^k - I$: The protocol uses two passes. The first pass sends $(n+k)$ messages (R to T) and the second pass sends 2n messages (S to R).

If we suppose that the hash function H_1 is a random oracle, then $OT_n^k - I$ is security for S when R is a malicious player, who may not follow the protocol dutifully. That is

Theorem 3.2: If R is a malicious player, $OT_n^k - I$ meet the requirement of the sender S by assuming that BDH problem is hard and H_1 is a random oracle.

In fact, if R wants to know un-chosen messages m_j, he/she should decrypt c_j ($1 \le j \le n$ and $j \ne \alpha_1, \alpha_2, ..., \alpha_k$), so R must computes $g_j^{r_j} = \hat{e}(P,P)^{r_j x_j s}$. That is: from

$$u_j = r_j P, P_{pub_j} = x_j P, Q = sP$$

to compute $\hat{e}(P,P)^{r_j x_j s}$. It is the BDH problem. In another words, if R can get the information about m_j ($1 \le j \le n$ and $j \ne \alpha_1, \alpha_2, ..., \alpha_k$), we can find an algorithm to solve the BDH problem.

Proof of Theorem 3.2: We will show that for each possible malicious R, we can construct a simulator R* in the ideal-model such that the outputs of R and R* are indistinguishable if BDH problem is hard and H_1 is a random oracle.

The simulator R* simulates both the sender S (externally, without knowing $m_1, m_2, ..., m_n$) and R.

R* works as follows:

1. R* simulates R to obtain $x_{\alpha_1}^*, x_{\alpha_2}^*, ..., x_{\alpha_k}^*$.
2. R* simulates S (externally, without knowing $m_1, m_2, ..., m_n$) to obtain $r_1^*, r_2^*, ..., r_n^*$ and $g_1^*, g_2^*, ..., g_n^*$.
3. R* randomly chooses $c_1^*, c_2^*, ..., c_n^*$.
4. R* monitors the queried of R closely, if R requires H_1 on some $v_j = \left(g_j^*\right)^{r_j^*}$, R* send j to the TTP to obtain m_j and return $c_j^* \oplus m_j$ as the hash value $H_1\left(\left(g_j^*\right)^{r_j^*}\right)$, otherwise, returns a random value (consistent with previous queries).
5. Outputs $\left(x_{\alpha_1}^*, x_{\alpha_2}^*, ..., x_{\alpha_k}^*; r_1^*, r_2^*, ..., r_n^*; c_1^*, c_2^*, ..., c_n^*\right)$.

If R obtains k+1 (i.e. more than k) decryption keys, R* does not know which k indices are really chosen by R. The simulation would fail. Therefore we show that R can obtain at most k decryption keys by assuming the hardness of BDH problem. In fact, if R knows more than k decryption keys, this in turn can be used to solve BDH problem when H_1 is a random oracle.

At any time, R may issue queries to the random oracle H_1. To respond to these queries R* maintains a list of tuples called the H_1^{list}. Each entry in the list is a tuple of the form (v_j, H_j). Initially, the list is empty. To respond to query v_j, R* does the following:

If the query v_j already appears on the H_1^{list} in a tuple (v_j, H_j), then responds with $H_1(v_j) = H_j$. Otherwise, R* just picks a random string $H_j \in \{0,1\}^l$ and adds the tuple (v_j, H_j) to the H_1^{list}. It responds to R with $H_1(v_j) = H_j$.

If R queries H_1 on legal v_{j_i} for all $1 \le i \le k+1$, we can output k+1 tuples (v_{j_i}, H_{j_i}). From the above simulation, R* can get know all k+1 equations

$$H_1(v_{j_i}) = H_1\left(\left(g_{j_i}^*\right)^{r_{j_i}^*}\right) = H_1\left(\hat{e}(P,P)^{r_{j_i}^* x_{j_i}^* s}\right).$$

At this point R* picks some or other random tuple $\left(v_{j_i}, H_{j_i}\right)$ from the H_1^{list} and outputs $\left(j_t \neq \alpha_1, \alpha_2, \ldots, \alpha_k\right)$ as the solution to $\hat{e}(P,P)^{r_{j_i}^* x_{j_t}^* s}\left(j_t \neq \alpha_1, \alpha_2, \ldots, \alpha_k\right)$. Therefore, the BDH problem is solved. It is impossible. So R obtains at most *k* decryption keys assuming that BDH problem is hard.

Let $\alpha_1, \alpha_2, \ldots, \alpha_k$ be the *k* choices of R. For the queried legal v_{α_i}'s, c_{α_i} is consistent with the returned hash values, for *i*=1,2,…,*k*. Since no other

$$\left(g_{j_i}^*\right)^{r_{j_i}^*}\left(j_t \neq \alpha_1, \alpha_2, \ldots, \alpha_k\right)$$

can be queried to the H_1, c_{j_i} has the right distribution (due to the random oracle model). Thus the output distribution of R* is indistinguishable from that of R.

C. $OT_n^k - II$

We will present an $OT_n^k - II$ protocol when a sender S is a cheating party, but S need not do any commitment and zero-knowledge proof.

Input: S's input are $m_1, m_2, \ldots, m_n \in \{0,1\}^l$. R's input are $\alpha_1, \alpha_2, \ldots, \alpha_k \in \{1,2,\ldots,n\}$

Output: R's output are $m_{\alpha_1}, m_{\alpha_2}, \ldots, m_{\alpha_k}$, that is he/she gets the information of his/her choices. S's output is nothing.

Initialization: The third party T, as a Private Key Generator (PKG) in IBE [19], using the security parameter κ, generates G_1, G_2 of order of *q*, G_1 is an additive group and G_2 is a multiplicative group, and a bilinear map: $\hat{e}: G_1 \times G_2 \to G_2$. Choose a cryptographic hash function H_1: $G_2 \to \{0,1\}^l$. T picks random $s \in Z_q^*$ and sets $Q = sP$, where $P \in G_1$ is a random generator. We call s master-key. In addition, T picks a hash function

$$H_2 : \{0,1\}^l \times \{0,1\}^l \to Z_q^*$$

and a hash function

$$H_3 : \{0,1\}^l \to \{0,1\}^l .$$

H_2, H_3 are public also.

Key Generation: R picks his/her choices

$$\alpha_1, \alpha_2, \ldots, \alpha_k \in \{1,2,\ldots,n\}$$

and finds a bi-jection

$$f : i \to x_i' = f(i) \in Z_q^*, i = 1,2,\ldots,n .$$

R sends $x_i'\left(i = 1,2,\ldots,n\right)$ and $\alpha_1, \alpha_2, \ldots, \alpha_k \in \{1,2,\ldots,n\}$ to T. For every $x_i' \in Z_q^*$, T chooses $x_i'' \in Z_q^*$ and let $x_i = x_i' + x_i''$, $x_1, x_2, \ldots, x_n$ are distinct, then computes

$$P_{pub_i} = x_i P, i = 1,2,\ldots,n .$$

T sends $x_{\alpha_j}\left(j = 1,2,\ldots,k\right)$ to R as his/her private keys.

$$q, G_1, G_2, \hat{e}, l, H_1, P, Q, P_{pub_i}\left(i = 1,2,\ldots,n\right)$$

are public.

Protocol:

Step1: S chooses random $\sigma_i \in \{0,1\}^l$ and sets $r_i = H_2(\sigma_i, m_i), i = 1,2,\ldots,n$. Let the cipher-texts be

$$c_i = \langle u_i, v_i, w_i \rangle .$$

where

$$u_i = r_i P ;\ v_i = \sigma_i \oplus H_1\left(g_i^{r_i}\right),\ w_i = m_i \oplus H_3(\sigma_i)$$

and

$$g_i = \hat{e}\left(Q, P_{pub_i}\right) \in G_2^* .$$

S sends $c_i\left(i = 1,2,\ldots,n\right)$ to *R*.

Step2 : On receiving $c_1, c_2, \ldots, c_n$, R used his secret keys $x_{\alpha_j}\left(j = 1,2,\ldots,k\right)$ to

1. Compute $v_{\alpha_j} \oplus H_1\left(\hat{e}\left(u_{\alpha_j}, x_{\alpha_j} Q\right)\right)$, it is equal to σ_{α_j} for *j*=1,2,…,*k*.

2. Compute $w_{\alpha_j} \oplus H_3\left(\sigma_{\alpha_j}\right)$, then get m'_{α_j}, for *j*=1,2,…,*k*.

3. Set $r_{\alpha_j} = H_2\left(\sigma_{\alpha_j}, m'_{\alpha_j}\right)$, test that $u_{\alpha_j} = r_{\alpha_j} P$. If it is, then $m'_{\alpha_j} = m_{\alpha_j}$. If it is not, output the special symbol $\perp$ and the protocol halts.

About the protocol $OT_n^k - II$, we have:

Correctness : For *j*=1,2,…,*k*, since

$$\hat{e}\left(u_{\alpha_j}, x_{\alpha_j} Q\right) = \hat{e}\left(r_{\alpha_j} P, x_{\alpha_j} sP\right) = \hat{e}\left(Q, P_{pub_{\alpha_j}}\right)^{r_{\alpha_j}} = g_{\alpha_j}^{r_{\alpha_j}} ,$$

So

$$v_{\alpha_j} \oplus H_1\left(\hat{e}\left(u_{\alpha_j}, x_{\alpha_j} Q\right)\right) =$$

$$\sigma_{\alpha_j} \oplus H_1\left(g_{\alpha_j}^{r_{\alpha_j}}\right) \oplus H_1\left(\hat{e}\left(d_{\alpha_j}, u_{\alpha_j}\right)\right) = \sigma_{\alpha_j}$$

and

$$w_{\alpha_j} \oplus H_3\left(\sigma_{\alpha_j}\right) = m_{\alpha_j} \oplus H_3\left(\sigma_{\alpha_j}\right) \oplus H_3\left(\sigma_{\alpha_j}\right) = m_{\alpha_j} .$$

From Step2 (3) we know that if S is a cheating party, the test $u_{\alpha_j} = r_{\alpha_j} P$ were not passed, so R can find that S has bluffed in the protocol and stops the protocol.

R's security is as protocol $OT_n^k - I$ **.**

About S's security there is

Theorem 3.3: Suppose the hash functions H_1, H_2 and H_3 are random oracles, the $OT_n^k - II$ meets the S's security requirement by assuming that BDH problem is hard, if R is a malicious player.

The proof is analogous to Theorem 3.2. We omit it.

IV. CONCLUSIONS

In this paper, we construct two three parties OT_n^k protocols from bilinear maps between groups against semi-honest receivers in the standard model and malicious receivers, cheating senders in the random oracle model without any commitment or zero-knowledge proof. The current trend of research on the design of cryptographic protocols is to find provably-secure practical protocols. Our results are along this direction.

There are numerous applications of OT and OT_n^k [1, 3, 12, 23]. Straightforward applications of the proposed OT_n^k are in electronic commerce. For example, we can construct an online video shop based on our OT_n^k protocols, which achieves the private transaction (which hiding from vendors what items clients are buying, or even whether at a given moment they are buying anything at all, being unable to learn for vendors what the clients' current balance is or when it actually runs out of its funds) of digital products, such as films, music, software and etc.. In this scenario, S is now the vendor who wants to sell videos over the Internet, while his/her clients who are considered as R can get what they want without revealing which ones they have selected.

Aiello,Ishai and Reingold [23], Christian Tobias [24] presented techniques to reach this goal by using oblivious transfer protocols.

ACKNOWLEDGEMENT

This work is supported by the National Nature Science Foundation of China under Grant No: 60873041; the Nature Science Foundation of Shandong Province of China under Grant No:Y2007G15.

REFERENCES

[1] M. Rabin. How to exchange secrets by oblivious transfer. Technical report TR81, Aiken Computation Laboratory, Harvard University,1981.

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

[3] M. Bellare, S. Micali. Non-interactive oblivious transfer. In Proceedings of Advances in Cryptology-Crypto 89, Lecture Notes in Computer Science 435, Springer-Verlag, 1990, pp.547-557.

[4] G. Brassard, C.Crépeau, J.-M. Robert. All-or nothing disclosure of secrets. In Proceedings of Advances in Cryptology-CRYPTO'86, volume 263 of LNCS, Springer-Verlag,1986, pp.234-238.

[5] G. Brassard, C.Crépeau, J.-M. Robert. Information theoretic reduction among disclosure problems. In Proceedings of the 28th Annual Symposium on Foundations of Computer Science (FOCS'87), IEEE, 19867, pp.427-437.

[6] G. Brassard, C.Crépeau,M.Sántha. Oblivious transfers and intersecting codes. IEEE Transactions on Information Theory, volum42(6),1996, pp.1769-1780.

[7] M. Noar, B. Pinkas. Oblivious transfer with adaptive queries. In Proceedings of Advances in Cryptology-CRYPTO'99, volume 1666 of LNCS, Springer-Verlag,1999, pp.573-590.

[8] Cheng-kang Chu,Wen-Guey Tzeng. Efficient *k*-out-of-*n* oblivious transfer schemes with adaptive and non-adaptive queries.Public Key Cryptography,5th International Workshop onPractice and Theory in Public Key Cryptosystems. Paris,2005,pp.172-183.

[9] Y. Mu, J. Zhang, V. Varadharajan. m out of n Oblivious transfer. ACISP'02, Lecture Notes in Computer Science, Vol.2384, 2002, pp.395-405.

[10] G.Yao, F.Bao, Robert Deng. Security analysis of three oblivious transfer protocols. Workingshop on coding, Cryptography and Combinatorics. Huangshan City, China, 2003.

[11] F.Bao, Robert Deng. An efficient and practical scheme for privacy protection in the E-commerce of digital goods[C]. Proc. ICICS'2000. LNCX 2015, Springer Verlag, 2001. pp.162-170.

[12] O. Goldreich, R. Vainish. How to solve any protocol problem: an efficient improvement. In Proceedings of Advances in Cryptology-Crypto 87, Lecture Notes in Computer Science293, Springer-Verlag, 1988, pp.73-86.

[13] O. Goldreich. Secure multi-party computation, Available at htttp://theory.lcs.mit.edu/~oded.

[14] Qin Jing, Li Li, Li Bao. An Efficient Non-interactive OT_n^k Protocol and its Application[J], J.of Beijing Univ. of Posts and Telecommunications. Vol.31,No.4. pp.1-5. 2008.8.(Chinese with English abstract).

[15] R.Cramer.Introduction to secure computation. Available http:www.inf.ethz.ch/personal/cramer.

[16] Qin Jing, Zhang Zhen-feng, Feng Deng-guo, Li Bao. A Protocol of Comparing Information without Leaking it[J], Journal of Software,Vol.15 No.3,pp.421-427,2004.(Chinese with English abstract)

[17] M. Noar, B. Pinkas. Efficient oblivious transfer protocols. In proceedings of SODA 01,2001, pp. 448-457.

[18] W-G. Tzeng. Efficient oblivious transfer schemes. Public key cryptography, 5th International Workshop on practice and theory in public key cryptosystems, Paris, France, February, PKC 2002.

[19] D. Boneh, M. Franklin. Identity-Based Encryption from the Weil Pairing. Advances in Cryptology-Crypto 2001, LNCS 2139, Springer-Verlag, 2001, pp.213-229.

[20] A. Joux, K. Nguyen. Separating Decision Diffie-Hellman from Diffie-Hellman in cryptographic groups. J. Cryptology 16(4), 2003, pp.239-247.

[21] C. Lawrence Washington. Elliptic Curves, Number theory and cryptography. A CRC Press Company, Boca Raton London New York Washington, D.C., 2003.

[22] Mao. W. Modern cryptography: Theory and Practice[M]. Publishing house of electronics industry, Beijing,2004, pp.85-136.

[23] B. Aiello, Y.Ishai and O. Reigold. Priced oblivious transfer: how to sell digital goods. Advances in Cryptology-Eurocrypto 2001. Lecture Notes in Computer Science 2045.Springer-Verlag, Berlin, 2001. pp.119-135.

[24] C.Tobias. Practical oblivious transfer protocols, Lecture Notes In Computer Science, volum2578, 2002, pp.415 - 426.

Qin Jing, female, was born in Jinan, Shandong province, P.R.China, in November 1960. In July 1982, she was graduated from Information Engineering University, Zhenzhou, P.R.China and got the Bachelor Degree in science. In June 2004 she got Doctor Degree of science from School of Mathematics in Shandong University, P.R.China. Her major field of study is information security and number theory for computing.

From June 1982 to February 1986, she was an assistant in Information Engineering University. Since March 1986, she works at School of Mathematics in Shandong University. Now she is a full professor in School of Mathematics, Shandong University. In recent years, her major research results include:

[1] Qin Jing, Li Li, Li Bao. An Efficient Non-interactive OT_n^k Protocol and its Application [J], Journal of Beijing University of Posts and Telecommunications. Vol.31,No.4. pp.1-5. 2008.8.

[2] Qin Jing, Zhang Zhen-feng, Feng Deng-guo, Li Bao. A Protocol of Comparing Information without Leaking it[J], Journal of Software,Vol.15 No.3,pp.421-427,2004.

[3] Qin Jing, Zhang Zhen-feng, Feng Deng-guo, Li Bao. A protocol of specific secure two-party computation[J], Journal of China Institute of Communications,Vol.25 No.11,pp.35-42,2004.

In the period of PH.D. Degree candidate, her research interest focused on design and analysis of security about cryptographic protocols, especially the protocols of secure multi-party computations and now her research interest is also in this field and especially in oblivious transfer protocol.

Dr. Qin is a senior member of Chinese Association for Cryptographic Research (CACR) and a famous teacher in the 4th of Shandong province.

Zhao huawei, male, was born in Liaocheng, Shandong province P.R.China in May 1977. In 2000, he wasgraduated from School of Computer Science and Technology in Shandong University of Technology, P.R.China and got the Bachelor Degree in Computer Science. In 2002 and 2006, he got Master Degree of Computer Application and Doctor Degree of Computer software and Theory respectively from School of Computer Science and Technology and Institute of Network and Information security in Shandong University, P.R.China. His major field of study is network and information security.

Since 2006, he works on Shandong University of Finance. Now he is a vice-professor and the director of network and information institute belongings to School of Computer and Information Engineering. In recent years, his major research results include:

[1] Zhao huawei, Li daxing. Security protocols' analytic approach of reconciling two views. Computer Research and Development,2006,43(7),pp:1260-1266.

[2] Zhao huawei, Zhang wenyu. Definitions and analysis of integrity in strand spaces model. Proceedings of the International Symposium on Electronic Commerce and Security, ISECS 2008,pp:153-158.

In the period of Master Degree candidate, his research interest focused on Public Key Infrastructure; in the period of PH.D. Degree candidate, his research interest focused on formal analysis of security protocols and now his research interest is security of wireless sensor network.

Now, Dr. Zhao is the member of Chinese Association for Cryptographic Research (CACR), and the member of standardization working group on sensor networks.