April 2003

Robust non-interactive oblivious transfer

Y. Mu
University of Wollongong, ymu@uow.edu.au

J. Zhang
Macquarie University

V. Varadharajan
Macquarie University

Y. X. Lin
University of Wollongong, yanxia@uow.edu.au

Robust non-interactive oblivious transfer

Abstract

We present a novel scheme of noninteractive m out of n oblivious transfer, which demonstrates significant improvement over the existing schemes in terms of completeness, robustness and flexibility. This scheme is useful for protection of user privacy in the Internet.

Keywords

Internet data privacy security of data telecommunication security

Publication Details

This paper originally appeared as: Mu, Y, Zhang, J, Varadharajan, V and Lin, YX, Robust non-interactive oblivious transfer, IEEE Communications Letters, April 2003, 7(4), 153-155. Copyright IEEE 2003.

Robust Non-Interactive Oblivious Transfer

Yi Mu, *Member, IEEE*, Junqi Zhang, Vijay Varadharajan, *Senior Member, IEEE*, and Yan-Xia Lin

***Abstract*—We present a novel scheme of noninteractive m out of n oblivious transfer, which demonstrates significant improvement over the existing schemes in terms of completeness, robustness and flexibility. This scheme is useful for protection of user privacy in the Internet.**

***Index Terms*—Data security, oblivious transfer.**

I. Introduction

THE concept of Oblivious Transfer (OT) was introduced by Rabin [1]. Rabin's OT can be considered as a game between two polynomial time parties, Alice and Bob. Alice sends a bit to Bob in such a way that with 1/2 probability Bob will receive the same bit and with 1/2 probability Bob will receive nothing. Alice does not know which event has happened. Rabin's initiative has attracted a lot of attentions. Various OT methods have been subsequently proposed (e.g., [2]–[7]), where most notable ones are one out of two OT and chosen one out of two OT. In a one out of two OT ($\binom{2}{1}$-OT), Alice sends two bits to Bob who receives one of these bits with equal probability and knows which bit he has received, while Alice does not know which bit Bob received. A chosen one out of two OT is similar to a normal one out of two OT; the different between them is that, in the former, Bob can choose an index c and receives bit b_c. Alice does not learn c.

One direct extension to $\binom{2}{1}$-OT is 1 out of n oblivious transfer [8], [9]. However, there has been little study in m out of n oblivious transfer, $\binom{n}{m}$-OT. The closest scheme is the $n-1$ out of n OT proposed by Bellare and Micali [10]. Roughly speaking, in an m out of n oblivious transfer, Bob can receive only m messages out of n messages ($n > m$) sent by Alice; and Alice has no idea about which ones have been received. The OT proposed by Bellare and Micali [10] is noninteractive. By noninteractive we mean that Bob does not need to communicate with Alice during an OT process. Santis and Persiano [8] also proposed a noninteractive OT protocol. Their scheme falls within the case of 1 out of n.

In this paper, we go one step further by giving a new noninteractive OT scheme that covers the complete OT spectrum. We called them m out of n OT, $\binom{n}{m}$-OT. Here, m is an arbitrary number in $1 \leq m < n$. The original OT scheme by Rabin can be considered as one of cases in our scheme. One important feature in our schemes is that the sender and the recipient can securely implement an OT process without the involvement of a trusted third party, because the security can be proved by both the sender and the recipient.

Manuscript received July 10, 2002. The associate editor coordinating the review of this letter and approving it for publication was Dr. L. Chen.

Y. Mu is with the School of Information Technology and Computer Science, The University of Wollongong, Wollongong, NSW 2500, Australia (e-mail: ymu@ics.mq.eu.au).

J. Zhang and V. Varadharajan are with the Department of Computing, Macquarie University, Sydney, NSW 2019, Australia.

Y.-X. Lin is with the School of Mathematics and Applied Statistics, The University of Wollongong, Wollongong, NSW 2500, Australia.

Digital Object Identifier 10.1109/LCOMM.2003.811213

II. System Setup

m of n oblivious transfer is defined as follows. Alice knows n messages and wants to send m of them to Bob. Bob gets m of them with probability $m!(n-m)!/n!$ and knows which ones he has got, but Alice has no idea about which m messages Bob has received.

Assume that Alice intends to send n messages, $M_1, \ldots, M_n \in \mathbb{Z}_p$, to Bob and knows for sure that Bob can receive m of them. Which ones will be received by Bob is unknown to Alice. We now describe the Bob's public key generation algorithm that will be used to our $\binom{n}{m}$-OTs.

Let p be a large prime number, $\mathbb{Z}_p^*$ be a multiplicative group, $g \in \mathbb{Z}_p^*$ be the generator of order $q = p-1$, and $0 \neq x_i \in \mathbb{Z}_p$, $i = 1, \ldots, n$, be a set of integers. All these data are pre-agreed and made public. For simplicity, we omit modulus p in the rest of the presentation.

The public key setup is done by Bob who selects m private keys $s_i \in \mathbb{Z}_q$ and then computes $y_i = g^{s_i}$, $i = 1, \ldots, m$ ($m < n$). Given x_i, the n public keys are constructed by using a set of m linear equations with respect to $a_1, \ldots, a_m$,

$$a_1 x_i + a_2 x_i^2 + \cdots + a_m x_i^m = y_i,\ i = 1, \ldots, m. \quad (1)$$

The corresponding linear equations in a matrix form are as follows:

$$\begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{m-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{m-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_m & x_m^2 & \cdots & x_m^{m-1} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} = \begin{pmatrix} y_1' \\ y_2' \\ \vdots \\ y_m' \end{pmatrix}. \quad (2)$$

Here $y_i' = y_i / x_i$. The coefficient matrix A is a so-called Vandermonde matrix. The determinant of A is not equal to zero

$$\det\ A = \prod_{1 \leq i < k \leq m} (x_i - x_k) \neq 0,$$

i.e., A is a nonsingular matrix, because x_i Bob chose are distinct and no element $(x_i - x_k)$ in this product equals zero. Since the determinant of the coefficient matrix is nonzero, the equations have a unique solution over the field $\mathbb{Z}_p$.

After Bob has got the unique solution $a_1, \ldots, a_m$, he can calculate other $n-m$ "public keys" (their discrete logs are unknown), using the following formula:

$$y_j = a_1 x_j + \cdots + a_m x_j^m,\ m < j \leq n.$$

As a result, he has n public keys $\{x_i, y_i\}_{i=1}^n$.

Bob shuffles his public keys such that the order is known to himself only. The shuffled public keys are then made public. For

convenience, we denote by $\mathbb{U}$ the subset of public key indices whose associated public key discrete logs are unknown to Bob and by $\mathbb{K}$ those known. Since the pubic keys will always come with a shuffled form, we still denote by $\{x_i, y_i\}_{i=1}^n$ the shuffled public key set.

The public keys can be easily verified without knowing the corresponding private keys. Given the public key set $\{x_i, y_i\}_{i=1}^n$, we can choose **any m** of public keys from the public key set, and then calculate $\hat{a}_i$ for $i = 1, \ldots, m$ with respect to the m public keys, where $\hat{a}_i \equiv a_i$ if the public keys are genuine. With the resultant $\hat{a}_i$, we can verify the rest of $(n-m)$ public keys,

$$\hat{y}_j = \hat{a}_1 x_j + \cdots + \hat{a}_m x_j^m, \qquad m < j \leq n.$$

Here, $\hat{y}_j \subset \{y_i\}_{i=1}^n$ have not been used in computation of $\hat{a}_i$.

A. Claim 1

Given x_i, Bob cannot cheat by pre-selecting y_i.

The explanation is as follows. After Bob found the unique coefficient set $\{a_i\}_{i=1}^m$, he can compute y_i for $i \in \mathbb{U}$ in terms of the given $\{x_i\}_{i=m+1}^n$. However, it is infeasible for him to compute the discrete logs of these values in poly-time. Bob should not be able to cheat by pre-selecting $\{y_i\}_{i=m+1}^n$ and then try to find $\{a_i\}_{i=1}^m$ that satisfies all n equations. To fix this potential problem, we give the following lemma.

Lemma 1: To prevent Bob from cheating by pre-selecting all $\{y_i\}_{i=m+1}^n$, the rank of matrix A' must be $m+1$, where

$$A' = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{m-1} & y_1' \\ 1 & x_2 & x_2^2 & \cdots & x_2^{m-1} & y_2' \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{m-1} & y_n' \end{pmatrix},$$

which is an $n \times (m+1)$ matrix.

Proof: Consider equation:

$$A' \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Because the rank of A' is $m+1$, by making row transformations, we have a nonsingular matrix C such that

$$CA' = \begin{pmatrix} B \\ 0 \end{pmatrix}$$

where B is an $(m+1) \times (m+1)$ nonsingular upper triangle matrix. Therefore

$$CA' \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \\ -1 \end{pmatrix} = \begin{pmatrix} B \\ 0 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

This is equivalent to

$$B \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Thus, we have

$$\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \\ -1 \end{pmatrix} = B^{-1} \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}.$$

Because B^{-1} is nonsingular, it implies that the equations with respect to a_i have no nonzero solution at all. In other words, Bob cannot find a nonzero solution if he wants to cheat by precomputing y_i. □

Therefore, in the verification of public keys, we also need to check if or not the rank of A' is equal to $m+1$.

III. Non-Interactive m Out of n OT

Using the setup phase given in Section II-A, Bob obtains his private keys s_i for $i = 1, \ldots, m$ and his public keys $\{x_i, y_i\}_{i=1}^n$ where the discrete logs of y_i for $i \in \mathbb{U}$ are not known. The protocol is described as follows.

Alice	**Bob**
$x_i, y_i, i = 1, \ldots, n$	$x_i, y_i, s_i, i = 1, \ldots, n$
$t_1, \ldots, t_n \in_R \mathbb{Z}_q$	
$\alpha_i = g^{t_i}$	
$z_i = M_i y_i^{t_i}$	
$\xrightarrow{\alpha_i, z_i}$	
	$\frac{z_i}{\alpha_i^{s_i}} = M_i, i \in \mathbb{K}$

- Alice:
  - randomly chooses $t_1, \ldots, t_n \in_R \mathbb{Z}_q$;
  - calculates $\alpha_i = g^{t_i}$, $i = 1, \ldots, n$;
  - generates the order of messages at random;
  - based on the order, calculates $z_i = M_i y_i^{t_i}$ for $i = 1, \ldots, n$;
  - then sends to Bob $\alpha_1, \ldots, \alpha_n$ and $z_1, \ldots, z_n$.
- Bob: decrypts z_i to recover m messages, $z_i/\alpha_i^{s_i} = M_i$, $i \in \mathbb{K}$.

A. Claim 2

(Completeness) If Alice correctly follows the procedure, Bob can recover m out of n messages, $1 \leq m < n$.

This is obvious. Note the facts that the order of the public keys are not changed and Bob knows their indices. Bob has m private decryption keys s_i, $i \in \mathbb{K}$, and knows which ones to decrypt. The encryptions done by Alice are based on the standard ElGamal encryption scheme. [11]

B. Claim 3

(Soundness) Both Alice and Bob cannot cheat.

Alice does not know which public keys are associated with Bob's m private keys, so she cannot know which messages Bob can decrypt and has no control over which messages Bob will receive. Bob cannot cheat by manipulating his public keys. This is because Alice can check the correctness of Bob's public key using the method described earlier in this paper. The security is, however, based on the assumption that our system is poly-time-bounded. Bob cannot solve the discrete log problem in poly-time.

C. Non-Malleable Encryption

In the scheme presented in the preceding section, Alice was assumed to be honest in that she always uses Bob's public keys in encryption. The assumption is reasonable, since Alice wants Bob to receive m out of n messages she sent. However, if the order of the public keys or the order of the ciphertext is changed by accident (or by an adversary) during the transmission, Bob will not be able to find the fraud in the case that the messages consist of unrecognizable strings. We now modify the scheme so that he can check if or not the encrypted messages sent by Alice are correctly constructed. We now construct a nonmalleable encryption by reconstructing the private keys: select private keys, $s_i \in \mathbb{Z}_q$, and some integers, $s_i' \in \mathbb{Z}_q$, such that they satisfy $s_i s_i' \bmod q = s_i$. It is not hard to find that we can select s_i and s_i' that satisfy $s_i(s_i' - 1) = q$ and $s_i' \neq 1$. Bob needs to keep s_i and s_i' secret. His public keys are still the same. The correctness of the encryptions can then be verified during the decryption. Bob now decrypts the obliviously transferred messages using two different methods: for message M_i,

Method 1: Compute $z_i^{s_i'}/\alpha_i^{s_i} = M_i^{s_i'}$ and then remove s_i'.

Method 2: Compute $z_i/\alpha_i^{s_i} = M_i$.

Bob then checks the equality of two messages. The completeness is straightforward. To prove the soundness, we assume that Alice has not correctly used Bob's public keys in her encryptions, but uses g^{σ_i}. Bob can immediately find the fraud.

Method 1: Compute $(M_i g^{t_i\sigma_i})^{s_i'}/\alpha_i^{s_i} = M_i^{s_i'} g^{t_i(s_i'\sigma_i - s_i)}$. Remove s_i' from the message, Bob then gets $M_i g^{t_i(\sigma_i - s_i s_i'^{-1})}$.

Method 2: Compute $M_i g^{t_i\sigma_i}/\alpha_i^{s_i} = M_i g^{t_i(\sigma_i - s_i)}$.

Obviously, they are not equal.

In conclusion, we have proposed a new non-interactive OT protocol that is provably secure. Because m can vary from 1 to $n-1$, our scheme covers the entire spectrum of noninteractive OT. This new setting has potential applications for protection of user privacy in the Internet.

REFERENCES

[1] M. O. Rabin, "How Exchange Secrets by Oblivious Transfer," Computer Science Lab., Harvard Univ., Cambridge, MA, TR-81, 1981.

[2] S. Even, O. Goldreich, and A. Lempel, "A randomized protocol for signing contracts," in *Proc. Advances in Cryptology, CRYPTO'82*, Berlin, Germany, 1982, pp. 205–210.

[3] C. Crepeau, "Equivalence between two flavors of oblivious transfers," in *Proc. Adances in Cryptology, CRYPTO'87*, vol. 304, Berlin, Germany, 1987, pp. 350–354.

[4] C. Crepeau and J. Kilian, "Weakening security assumptions and oblivious tranfer," in *Proc. Advances in Cryptology, CRYPTO'88*, vol. 403, Berlin, Germany, 1988, pp. 2–7.

[5] G. Brassard, C. Crepeau, and J.-M Robert, "Information theoretic reductions among disclose problem," in *Proc. Symp. on Foundations of Computer Science*, 1986, pp. 168–173.

[6] B. den Boer, "Oblivious transfer protecting secrecy," in *Proc. Advances in Cryptology, EUROCRYPT 90*, vol. 473, Berlin, Germany, 1990, pp. 31–46.

[7] G. Brassard and C. Crepeau, "All or nothing disclosure of secrets," in *Proc. Advances in Cryptology, CRYPTO 86*, vol. 263, Berlin, Germany, 1987.

[8] A. D. Santis and G. Persiano, "Public-randomness in public-key cryptography," in *Proc. Advances in Cryptology, EUROCRYPT 90*, vol. 473, Berlin, Germany, 1990, pp. 46–61.

[9] B. Aiello, Y. Ishai, and O. Reingold, "Priced oblivious transfer: How to sell digital goods," in *Proc. Adances in Cryptology, EUROCRYPTO 2001*, vol. 2045, Berlin, Germany, 2001, pp. 119–135.

[10] M. Bellare and S. Micali, "Non-interactive oblivious transfer and application," in *Proc. Advances in Cryptology, Proc. CRYPTO 89*, vol. 435, Berlin, Germany, 1989, pp. 547–557.

[11] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," in *Proc. Advances in Cryptology, Proc. CRYPTO 84*, vol. 196, Berlin, Germany, 1985, pp. 10–18.