Efficient 1-Out-of-n Oblivious Transfer Schemes with Universally Usable Parameters

Wen-Guey Tzeng

Abstract—In this paper, we propose efficient and secure (string) oblivious transfer (OT_n^1) schemes for any $n \geq 2$. We build our OT_n^1 scheme from fundamental cryptographic techniques directly. The receiver's choice is unconditionally secure and the secrecy of the unchosen secrets is based on the hardness of the decisional Diffie-Hellman problem. Some schemes achieve optimal efficiency in terms of the number of rounds and the total number of exchanged messages for the case that the receiver's choice is unconditionally secure. The distinct feature of our scheme is that the system-wide parameters are independent of n and *universally usable*, that is, all possible receivers and senders use the same parameters and need no trapdoors specific to each of them. We extend our OT_n^1 schemes to distributed oblivious transfer schemes. Our distributed OT_n^1 schemes take full advantage of the research results of secret sharing. For applications, we present a method of transforming any (single-database) PIR protocol into a symmetric PIR protocol by slightly increasing the communication cost only.

Index Terms—Oblivious transfer, distributed oblivious transfer, private information retrieval.

1 Introduction

Rabin [40] proposes the concept of the two-party oblivious transfer (OT) scheme in the cryptographic scenario. It has many flavors, such as original oblivious transfer (OT), 1-out-of-2 oblivious transfer (OT_2^1), introduced in [23], and 1-out-of-n oblivious transfer (OT_n^1), introduced in [10]. For OT, the sender, S, has only one secret, m, and would like to have the receiver, R, obtain m with probability 0.5. On the other hand, R does not want S to know whether it gets m or not. For OT_2^1, S has two secrets, m_1 and m_2, and would like to give R one of them at R's choice. Again, R does not want S to know which secret it chooses. OT_n^1 is a natural extension of OT_2^1 to the case of n secrets, in which S has n secrets $m_1, m_2, \ldots, m_n$ and is willing to disclose exactly one of them to R at R's choice. OT_n^1 is also known as "all-or-nothing disclosure of secrets (ANDOS)" in which R is not allowed to gain combined information of the secrets, such as their exclusive-or. Essentially, all these flavors are equivalent in the information theoretic sense [9], [12], [17]. Oblivious transfer is a fundamental primitive for cryptography and secure distributed computation [29], [32] and has many applications, such as private information retrieval (PIR), fair electronic contract signing, oblivious secure computation, etc. [6], [16], [23].

A general approach for constructing an OT_n^1 scheme is that we first construct a basis OT_2^1 scheme and then build the OT_n^1 scheme by (explicitly or implicitly) invoking the basis OT_2^1 scheme for many runs, typically, n or $\log_2 n$ runs [9], [11], [34]. Another approach is to build an OT_n^1 scheme from basic techniques directly [37], [38], [41], [46].

In this paper, we propose efficient string OT_n^1 schemes for any $n \geq 2$. We build our OT_n^1 schemes from fundamental cryptographic techniques directly. The receiver's choice α is unconditionally secure and the secrecy of the unchosen secrets m_i, $i \neq \alpha$, is based on the hardness of the decisional Diffie-Hellman problem. Our OT_n^1 schemes are very efficient in computation and achieve optimal efficiency in terms of the number of rounds and the total number of exchanged messages for the case that R's choice is unconditionally secure. In the OT_n^1-I scheme, R needs to compute two modular exponentiations only, no matter how large n is, and S needs to compute $2n$ modular exponentiations. By the speedup techniques in [31], S's computation time can be much reduced. If we assume the random oracle model, in the scheme OT_n^1-III, R needs to compute two modular exponentiations and S needs to compute three modular exponentiations only. The distinct feature of our schemes is that the system-wide parameters are independent of n and *universally usable*, that is, all possible receivers and senders use the same parameters and need no trapdoors (e.g., factorization of $N = pq$) specific to each of them.

We combine our OT_n^1 schemes with any secret sharing scheme to form efficient distributed OT_n^1 schemes [36]. In this setting, there are p servers. Each server holds partial information about the secret m_is. If R contacts t (the threshold) or more servers, it can compute m_α of its choice; otherwise, it cannot get any information about the secrets. Our threshold OT_n^1 schemes take full advantage of the research results of secret sharing. In particular, we construct an access-structure distributed OT_n^1 scheme (Γ-OT_n^k).

For applications, we present a method of transforming any (single-database) PIR protocol into a symmetric PIR (SPIR) protocol by slightly increasing communication cost, two extra messages and one extra round at most. As SPIR is equivalent to OT_n^1, this method provides an efficient reduction from PIR to OT_n^1. In particular, any computational PIR [33] in which the receiver's choice is computationally

• *The author is with the Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan 30050. E-mail: tzeng@cis.nctu.edu.tw.*

Manuscript received 25 Feb. 2002; revised 16 Dec. 2002; accepted 6 June 2003. For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number 115948.

secure with efficient communication complexity can be transformed into a communication-efficient OT_n^1 scheme with R's choice being computationally secure. Some communication-efficient PIR schemes have been proposed [14], [33].

1.1 Previous Work and Comparison

Oblivious transfer has been studied in various flavors and security models extensively (cf. [2], [4], [7], [9], [11], [13], [18], [20], [23], [27], [34], [38], [41], [44], [46]). In particular, bit OT_2^1 (where m_1 and m_2 are only one-bit) attracts much attention from researchers since it is the basis oblivious transfer scheme to which string OT_2^1 and OT_n^1 schemes are reduced. Most previous oblivious transfer schemes are based on hardness of factorization or quadratic residuosity problems.

The reduction approach is studied in [8], [9], [11], [17], [34]. For example, a k-bit string OT_2^1 scheme can be achieved by invoking βk runs of a bit OT_2^1 scheme for some β, $2 \leq \beta \leq 18$, [8], [9], [11]. In [34], a string OT_n^1 scheme is constructed by invoking $\log_2 n$ runs of a string OT_2^1 scheme.

The generic construction is studied in [1], [23], [37], [38], [41], [46]. The scheme in [46] is a general construction for OT_n^1 based on a public-key encryption scheme with some specific properties. The receiver's choice of the scheme is computationally secure. The scheme takes $O(\sqrt{\log_2 n})$ rounds if better efficiency for exchanged messages is desired.

Recently, an efficient two-round OT_n^1 in amortized analysis was proposed in [37]. The sender uses one modular exponentiation on average for each invocation. For comparison, the scheme is indeed more efficient than ours (which needs three modular exponentiations for the sender) in computation when the scheme is invoked many times. But, the size of the system parameter of the scheme is $O(n)$, while ours is a constant, independent of n. Furthermore, our schemes can be extended to threshold oblivious transfer easily and used to transfer any PIR protocol into an SPIR protocol by slightly increasing the communication complexity. In [1], a general methodology, based on conditional opening and the homomorphic property of a public-key encryption scheme, is proposed to construct two-round OT_n^1 schemes. For the schemes, each receiver needs a pair of public and private keys. Therefore, the parameters are not universally usable.

Distributed oblivious transfer has been studied in various contexts under variant models, such as function evaluation [3] and private information retrieval [28]. In the threshold OT_2^1 scheme in [36], the receiver and involved servers need not do public-key operations, such as modular exponentiations. For comparison, in our distributed version, the receiver and each server need one invocation of our OT_n^1 scheme.

In some sense, our schemes fall in the category of noninteractive oblivious transfer [4], [44] in which the receiver selects a public key and the sender performs noninteractive oblivious transfer using the receiver's public key. The schemes in [44] are based on the quadratic residuosity assumption. Each receiver R uses a specific Blum integer N that is reusable by the R only. The receiver's choice is computationally secure and the privacy of the unchosen secrets is unconditionally secure. The bit OT_2^1 scheme is extended to the bit OT_n^1 scheme in which the size of the receiver's public key is $O(n)$.

Transformation from PIR to SPIR has been studied in [19], [34]. The reduction in [34] makes a call to the basic PIR scheme and $\log_2 n$ calls to an OT_2^1 scheme. The reduction in [19] uses communication complexity $poly(t) \cdot c(n)$, where $c(n)$ is the communication cost of the basic PIR scheme and t is the security parameter. For comparison, our reduction uses communication cost $c(n) + O(t)$.

2 Preliminaries

Involved parties. The involved parties are the sender S and the receiver R, which are both polynomial-time probabilistic Turing machines (PPTM). An involved party is semihonest (passive or curious) if it follows the protocol step by step, but may try to compute extra information from received messages. An involved party is malicious (or active) if it deviates from the protocol in an arbitrary way in order to get extra information. For example, a malicious party can send a message that is not of the form defined in the protocol. We consider the semihonest sender S and the semihonest/malicious receiver R.

Security model. Let $m_1, m_2, \ldots, m_n$ be the secrets of S. Since S is semihonest, it won't send secrets that are different from the claimed ones, either in content or in order. The security definition is based on computational indistinguishability. Two probability ensembles $\{X_n\}$ and $\{Y_n\}$ are computationally indistinguishable if, for every PPTM distinguisher D, every polynomial $p(n)$, and sufficiently large n,

$$|\Pr[D(X_n) = 1] - \Pr[D(Y_n) = 1]| < 1/p(n).$$

Since X_n and Y_n look the same for D, if D cannot compute information from X_n, it cannot compute information from Y_n either and vice versa.

An OT_n^1 scheme should meet the following requirements [34]:

1. Correctness: The protocol achieves its goal if both R and S behave properly. That is, if both R and S follow the protocol step by step, R gets m_α after executing the protocol with S, where α is R's choice.
2. Receiver's privacy—indistinguishability: The transcripts corresponding to R's different choices α and α', $\alpha \neq \alpha'$, are computationally indistinguishable to S. If the transcripts are identically distributed, the choice of R is unconditionally secure.
3. Sender's privacy—compared with Ideal Model: In the Ideal Model, a trusted third party (TTP) T acts as an intermediary agent who receives S's secrets $m_1, m_2, \ldots, m_n$ and R's choice α and gives m_α to R. Since R has no way of getting information other than m_α, this model is considered the most secure way to implement oblivious transfer. Therefore, we say that the sender's privacy is guaranteed if, for every possible malicious R which interacts with S, there is a simulator R' (a PPTM) which interacts with T such that the output of R' is computationally indistinguishable from the output of R.

Efficiency. We consider computation and communication efficiency. For computation efficiency, we count the most expensive modular exponentiation of computing $a^b \bmod n$. The other operations, such as hashing, single multiplication, and division, are considered much cheaper. For communication efficiency, we consider both the round efficiency and message efficiency.

Proof of knowledge systems. A zero-knowledge proof of knowledge (ZKPK) system is an interactive proof system between a prover P and a verifier V such that, on a common input y, P convinces V that it owns some secret knowledge (witness) corresponding to y without revealing any information about the secret [30]. A noninteractive ZKPK (NIZKPK) system is a ZKPK system such that P sends a message (string) β to V and V verifies β to determine whether to accept P's assertion. In NIZKPK, P and V need to share a common random string, which may be publicly broadcast [45], [43] or given by a trusted third party.

Assume that each common input y corresponds to two or more secrets. Then, a proof of knowledge system for these inputs is witness-indistinguishable (WIPK) if P, which owns a secret of the input y, convinces V of this fact and the interaction transcript is computationally indistinguishable from that if P owns another secret [25]. A WIPK is perfect (also called witness-independent PK) if the interaction transcripts corresponding to two different secrets are identically distributed.

Random oracle model. Some of our schemes use cryptographically strong hash function H. It is a common practice in security analysis of cryptography to assume that H is a truly random function, called the *random oracle model* [5]. The answer for each query is random, but consistent with previous queries, that is, the same queries are answered with the same hash value. Furthermore, one cannot compute the hash value except by querying the hash oracle. In practicality, H is implemented with, for example, the SHA-1 function.

Though a provably secure protocol based on the random oracle model is more efficient, the random oracle model is not realistic. It has been shown that some protocol proven secure under the random oracle model is not necessarily secure in the real situation [15]. Nevertheless, the counterexample in [15] is artificial. The random oracle model is widely used in security analysis of cryptography.

Diffie-Hellman assumptions. Let G_q be a group of order q and g be a generator of G_q, where q is prime. Any element in $G_q \setminus \{1\}$ is a generator of G_q. Hereafter, all operations are over G_q whenever clear. Typically, G_q is the set of quadratic residues of Z_p^*, where $p = 2q + 1$ is also prime. In this case, the exponentiation $g^x \bmod p$ is denoted as g^x, $x \in Z_q$. Let $x \in_R X$ denote that x is chosen uniformly and independently from the set X. The Decisional Diffie-Hellman (DDH) assumption is that the following two distribution ensembles, indexed on G_q, are computationally indistinguishable:

- $Y_1 = \{(g, g^a, g^b, g^{ab})\}_{G_q}$, where $g \in_R G_q \setminus \{1\}$ and $a, b \in_R Z_q$;
- $Y_2 = \{(g, g^a, g^b, g^c)\}_{G_q}$, where $g \in_R G_q \setminus \{1\}$ and $a, b, c \in_R Z_q$.

Note that the description of G_q (in most cases, (p, q)) is given to the algorithm implicitly. We also omit the security parameter $t = size(q)$ hereafter for simplicity. The Computational Diffie-Hellman (CDH) assumption states that no PPTM can compute g^{ab} from given g, g^a, and g^b with nonnegligible probability, which decreases faster than the reciprocal of any polynomial. The DDH assumption is stronger than the CDH assumption. Also, the DDH assumption is stronger than the discrete logarithm (DL) assumption, which states that no PPTM can compute $x = \log_g y$ from given g and $y \in G_q$ with nonnegligible probability.

3 Oblivious Transfer against Semihonest Receiver

We first present a basic oblivious transfer scheme with security against the semihonest receiver, which follows the protocol step by step, but tries to compute information about the unchosen secrets.

Assume an order-q group G_q with a short description, where q is a large prime. Let g and h be two generators of G_q such that the discrete logarithm $\log_g h$ is unknown to all. As long as $\log_g h$ is not revealed, g and h can be used repeatedly. The system-wide parameters (g, h, G_q) are used by all possible senders and receivers.

Our OT_n^1 scheme with security against the semihonest receiver is as shown in Fig. 1. Without loss of generality, we assume that all secrets m_is are in G_q.

Correctness. Since $c_\alpha = (a, b) = (g^{k_\alpha}, m_\alpha (y/h^\alpha)^{k_\alpha})$, we have

$$b/a^r = m_\alpha (y/h^\alpha)^{k_\alpha}/(g^{k_\alpha})^r = m_\alpha (g^r h^\alpha / h^\alpha)^{k_\alpha}/(g^{k_\alpha})^r = m_\alpha.$$

Efficiency. The scheme takes only two rounds. This is optimal since at least R has to choose α and let S know and S has to respond to R's request. R sends one message y to S and S sends n messages c_i, $1 \le i \le n$, to R. This is also optimal (within a constant factor of 2) by the argument for the lower bound $\Omega(n)$ of communication cost of the single-database PIR when R's choice is unconditionally secure [17].

For computation, R needs two modular exponentiations for y and a^r. Straightforwardly, S needs $2n$ modular exponentiations for c_i, $1 \le i \le n$. We can reduce the computation by using the fast exponentiation methods. For example, S precomputes g^{2^j} and $h^{-2^j}, 1 \le j \le l$, where $l = \lfloor \log_2 q \rfloor$. When receiving y, S computes $y^{2^j}, 1 \le j \le l$. Then, S chooses k_i, $1 \le i \le n$, and computes c_i by multiplying appropriate g^{2^j}, h^{-2^j}, and y^{2^j}, $1 \le j \le l$.

Security. The above OT_n^1 scheme has the properties that the choice α of R is unconditionally secure and R gets no information about any other m_i, $i \ne \alpha$, if the DDH problem is hard.

Theorem 3.1. *For scheme OT_n^1-I, R's choice α is unconditionally secure.*

Proof. For any α', there is r' that satisfies $y = g^{r'} h^{\alpha'}$. Therefore, S cannot get any information about R's α even if it has unlimited computing power. □

Theorem 3.2. *For scheme OT_n^1-I, if R is semihonest, it gets no information about m_i, $1 \le i \ne \alpha \le n$, assuming the hardness of the DDH problem. That is, for all $i \ne \alpha$, $e_i = (g, h, c_i)$ are computationally indistinguishable from $x = (g, h, a, b)$, $g, h \in_R G_q \setminus \{1\}$, $a, b \in_R G_q$, even if R knows (r, α) in $y = g^r h^\alpha$.*

Scheme $\mathbf{OT_n^1}$-I:

- System parameters: (g, h, G_q);
- S's input: $m_1, m_2, \ldots, m_n \in G_q$;
- R's choice: α, $1 \le \alpha \le n$;

1. R sends $y = g^r h^\alpha$, $r \in_R Z_q$;
2. S sends $c_i = (g^{k_i}, m_i(y/h^i)^{k_i})$, $k_i \in_R Z_q$, $1 \le i \le n$;
3. By $c_\alpha = (a, b)$, R computes $m_\alpha = b/a^r$.

Fig. 1. Scheme OT_n^1-I.

0. S chooses (g, h, G_q) and sends them to R, where $g, h \in_R G_q \backslash \{1\}$.

Fig. 2. Added step to scheme $OT_n^1 - 1$.

Proof. Since the DDH assumption is stronger than the DL assumption, R cannot compute two different pairs of (r, α) and (r', α') that satisfy $y = g^r h^\alpha = g^{r'} h^{\alpha'}$. Otherwise, R computes $\log_g h = (r' - r)/(\alpha - \alpha') \bmod q$. Thus, R cannot get two secrets.

We show that, for each $i \neq \alpha$, $e_i = (g, h, c_i)$ looks random assuming hardness of the DDH problem. Formally, we define the random variable

$$E_i = (g, h, g^{k_i}, m_i(g^r h^{\alpha - i})^{k_i}),$$

where $k_i \in_R Z_q$, $g, h \in_R G_q \setminus \{1\}$. Note that we treat g and h as random variables in E_i. Let $X = (r_1, r_2, r_3, r_4)$, where $r_1, r_2 \in_R G_q \setminus \{1\}$ and $r_3, r_4 \in_R G_q$. We show that, if E_i and X are distinguishable by a PPTM distinguisher D, Y_1 and Y_2 of the DDH problem are distinguishable by the following PPTM distinguisher D', which uses D as a subroutine:

- Input: (g, u, v, w) (which is either from Y_1 or Y_2);
  1. If $u = 1$, then output 1;
  2. Randomly select $r \in Z_q$;
  3. If $D(g, u, v, m_i v^r w^{\alpha - i}) = 1$, then output 1, else output 0.

We can see that if $(g, u, v, w) = (g, g^a, g^b, g^{ab})$ is from Y_1 and $a \neq 0$,

$$(g, u, v, m_i v^r w^{\alpha - i}) = (g, h, g^b, m_i(g^r h^{\alpha - i})^b)$$

has the right form for E_i, where $h = u$. If $(g, u, v, w) = (g, g^a, g^b, g^c)$ is from Y_2 and $a \neq 0$,

$$(g, u, v, m_i v^r w^{\alpha - i}) = (g, h, g^b, m_i g^{br + c(\alpha - i)})$$

is uniformly distributed over

$$G_q \setminus \{1\} \times G_q \setminus \{1\} \times G_q \times G_q,$$

which is X. Therefore, if D distinguishes E_i and X with a nonnegligible advantage ϵ, D' distinguishes Y_1 and Y_2 with an advantage $\epsilon \cdot (1 - 1/q) + 1/q$, where $1/q$ is the offset probability in Step 1. □

3.1 Without System-Wide Parameters

We can remove the requirement of using system-wide parameters (g, h, G_q). Now, S first chooses g, h and G_q and sends them to R, that is, the step shown in Fig. 2 is added to the scheme.

When R receives (g, h, G_q), it needs to check that q is prime, $g \neq 1$, and $h \neq 1$. Otherwise, if S chooses a nonprime q and g and h of small orders, it can get information about R's choice. Note that, even if S knows $\log_g h$, R's choice α is still unconditionally secure.

4 Oblivious Transfer against Malicious Receiver

In the scheme OT_n^1-I, a malicious R may not follow Step 1 to compute y. Instead, R computes y of some special form such that it is possible to compute combined information of the secrets, such as $m_i \oplus m_j$, $i \neq j$. We don't know whether such y exists. To prevent this attack, we require R to know (r, α) that satisfies $y = g^r h^\alpha$. Two solutions are presented. One is based on the witness-indistinguishable proof of knowledge (WIPK) system and the other is based on the random oracle model.

4.1 Based on WIPK

The witness set about (y, g, h) contains all $(r, \alpha) \in Z_q \times Z_q \setminus \{0\}$ that satisfy $y = g^r h^\alpha$. The WIPK-based OT_n^1 scheme with security against the malicious receiver is as shown in Fig. 3.

Since $yy'^c = g^r h^\alpha (g^{r'} h^{\alpha'})^c = g^{r + r'c} h^{\alpha + \alpha' c} = g^{z_1} h^{z_2}$, the correctness of the scheme follows easily. For computation, R needs three modular exponentiations for y, y', and a^r. S needs 2n + 3 modular exponentiations for checking $yy'^c \neq$

Scheme $\mathbf{OT_n^1}$-II:

- System parameters: (g, h, G_q);
- S's input: $m_1, m_2, \ldots, m_n \in G_q$;
- R's choice: α, $1 \leq \alpha \leq n$;

1. R sends $y = g^r h^\alpha$ and $y' = g^{r'} h^{\alpha'}$, where $r, r', \alpha' \in_R Z_q$;
2. S sends $c \in_R Z_q$;
3. R sends $z_1 = (r + r'c) \bmod q$ and $z_2 = (\alpha + \alpha' c) \bmod q$;
4. If $yy'^c \neq g^{z_1} h^{z_2}$ then S aborts, else S sends $c_i = (g^{k_i}, m_i(y/h^i)^{k_i})$, $k_i \in_R Z_q$, $1 \leq i \leq n$;
5. By $c_\alpha = (a, b)$, R computes $m_\alpha = b/a^r$.

Fig. 3. Scheme OT_n^1-II.

$g^{z_1} h^{z_2}$ and computing c_i, $1 \leq i \leq n$. We can speed up S's computation by precomputation, as discussed in Section 3. The security is shown as follows.

Theorem 4.1. *The scheme OT_n^1-II meets the requirements of Receiver's privacy and Sender's privacy assuming hardness of the DDH problem.*

Proof. The value y is treated as the common input. The first three steps, with messages $y', c, (z_1, z_2)$, constitute a typical 3-round perfect WIPK system for the witnesses of y. Since $y = g^r h^\alpha$, $r \in_R Z_q$, is uniformly distributed over G_q and the perfect WIPK system leaks no information about (r, α) unconditionally, R's choice is unconditionally secure.

For each malicious R in the real run, we construct a simulator R' in the Ideal Model such that the outputs of R and R' are computationally indistinguishable as follows: First, R' simulates R to the point of producing $(\bar{y}, \bar{y}')$. Then, R' randomly selects $\bar{c} \in Z_q$ as S's challenge and continues the simulation to get $(\bar{z}_1, \bar{z}_2)$. If R produces valid $(\bar{y}, \bar{y}', \bar{z}_1, \bar{z}_2)$ with nonnegligible probability (taken over $\bar{c}$), by the soundness property of the WIPK system, R' can use R as a subroutine in a resettable way to compute α with overwhelming probability. If the simulation fails to produce α, TTP T outputs $\perp$ (*abort*). The probability that TTP T outputs $\perp$ is almost equal to the probability that S aborts the protocol in Step 4. After obtaining α, R' sends α to TTP T and gets m_α. R' sets $\bar{c}_\alpha = (g^k, m_\alpha(y/h^\alpha)^k)$, $k \in_R Z_q$, and $\bar{c}_i = (a_i, b_i)$ for $1 \leq i \neq \alpha \leq n$, $a_i, b_i \in_R G_q$. Finally, R' outputs $(\bar{y}, \bar{y}', \bar{c}, \bar{z}_1, \bar{z}_2, \bar{c}_1, \ldots, \bar{c}_n)$ as the simulation result.

We now show that if there is a PPTM D that distinguishes R's view $(y, y', c, z_1, z_2, c_1, c_2, \ldots, c_n)$ from the simulation result $(\bar{y}, \bar{y}' \bar{c}, \bar{z}_1, \bar{z}_2, \bar{c}_1, \ldots, \bar{c}_n)$ of R' with nonnegligible probability ϵ, then there is another PPTM D' that distinguishes Y_1 from Y_2 of the DDH problem with probability ϵ/n. The distributions of (y, y', c, z_1, z_2) and $(\bar{y}, \bar{y}', \bar{c}, \bar{z}_1, \bar{z}_2)$ are identical due to direct simulation of R. Also, c_α and $\bar{c}_\alpha$ are identically distributed since they both are encryptions of m_α. By the triangular inequality, there is an $i \neq \alpha$ such that the distributions

$$X_1 = (y, y', c, z_1, z_2, c_1, \ldots, c_{i-1}, c_i, \bar{c}_{i+1}, \ldots, \bar{c}_n)$$

and

$$X_2 = (y, y', c, z_1, z_2, c_1, \ldots, c_{i-1}, \bar{c}_i, \bar{c}_{i+1}, \ldots, \bar{c}_n)$$

are distinguishable by D with probability ϵ/n at least. Then, D' takes as input (g, u, v, w), sets $h = u$, and computes

$$\begin{aligned} X_3 = (y, y', c, z_1, z_2, &(g^{k_1}, m_1(y/h)^{k_1}), \ldots, \\ &(g^{k_{i-1}}, m_{i-1}(y/h^{i-1})^{k_{i-1}}), \\ &(g^{k_i}, m_i v^r w^{\alpha - i}), (a_{i+1}, b_{i+1}), \ldots, (a_n, b_n)), \end{aligned}$$

where $k_j \in_R Z_q, 1 \leq j \leq i$, $r \in Z_q$, and

$$a_j, b_j \in_R G_q, i + 1 \leq j \leq n.$$

By the same argument as Theorem 3.2, if (g, u, v, w) is from Y_1, then X_3 is equal to X_1; if (g, u, v, w) is from Y_2, X_3 is equal to X_2. Thus, D', using D as a subroutine, distinguishes Y_1 from Y_2 with nonnegligible probability ϵ/n at least. This is a contradiction. Therefore, R's view and the simulation result of R' are computationally indistinguishable. The scheme meets the requirement of Sender's privacy. □

The scheme OT_n^1-II takes four rounds due to the interaction nature of WIPK. We can use noninteractive ZKPK to reduce the number of rounds to two. In this case, R sends a string β to prove its knowledge of (r, α)

Scheme $\mathbf{OT_n^1}$-III:

- System parameters: (g, h, G_q);
- S's input: $m_1, m_2, \ldots, m_n \in G_q$;
- R's choice: α, $1 \le \alpha \le n$;

1. R sends $y = g^r h^\alpha$, $r \in_R Z_q$;
2. S sends $a = g^k$, and $c_i = m_i \oplus H((y/h^i)^k, i)$, $k \in_R Z_q$, $1 \le i \le n$;
3. By a and c_α, R computes $m_\alpha = c_\alpha \oplus H(a^r, \alpha)$.

Fig. 4. Scheme OT_n^1-III.

in $y = g^r h^\alpha$ in Step 1 of the scheme OT_n^1-I. The scheme based on NIZKPK has two different points. The first is that R and S have to share a random string, which may be publicly broadcast. The second is that β only conceals α computationally. R's choice is only computationally secure.

4.2 Based on Random Oracle Model

We apply the technique in [37] to achieve security against the malicious receiver, assuming the random oracle model and hardness of the CDH problem. Let H be a cryptographically strong hash function. The scheme is as shown in Fig. 4.

The correctness of the scheme follows easily. As for computation, R needs two modular exponentiations for y and a^r and S needs three modular exponentiations for a, y^k, and h^k, where a and h^k can be precomputed. The security is shown as follows.

Theorem 4.2. *The scheme OT_n^1-III meets the requirements of Receiver's privacy and Sender's privacy assuming the random oracle model and hardness of the CDH problem.*

Proof. We can see that R's choice α is unconditionally secure.

In the random oracle model, the malicious R has to know the whole information $(y/h^i)^k$ in order to query the hash oracle to get $H((y/h^i)^k, i)$. If R can compute two values $t_1 = (y/h^i)^k$ and $t_2 = (y/h^j)^k$, $i \neq j$, it can compute $h^k = (t_1/t_2)^{1/(j-i)}$. This implies the following method of solving the CDH problem: For given g, $g^{a'}$ and $g^{b'}$, let $h = g^{a'}$ and $a = g^{b'}$, and compute $h^k = g^{a'b'}$. Therefore, R cannot compute both $(y/h^i)^k$ and $(y/h^j)^k$ for $i \neq j$ with nonnegligible probability.

The following simulator R' for the Ideal Model outputs an indistinguishable distribution:

1. Simulate R for generating $\bar{y}$;
2. Randomly select $\bar{c}_i$, $1 \le i \le n$;
3. Simulate S on input $\bar{y}$ (externally without knowing m_is) to obtain $\bar{k}$ and compute $\bar{a} = g^{\bar{k}}$;
4. Simulate R on input $\bar{a}$ and $\bar{c}_i$s while monitoring its queries to the hash oracle closely. If R queries (z, j) and $z = (\bar{y}/h^j)^{\bar{k}}$ (this j is α), R' sends j to TTP T to obtain m_j and returns $h = \bar{c}_j \oplus m_j$ as the hash value $H(z, j)$; otherwise, R' returns a random hash value conditioned on the consistency of previous hash values;
5. Output $(\bar{y}, \bar{a}, \bar{c}_1, \bar{c}_2, \ldots, \bar{c}_n)$.

When R queries (z, j) with $z = (\bar{y}/h^j)^{\bar{k}}$, it must know $\bar{r}$ in $g^{\bar{r}} = \bar{y}/h^j$. Otherwise, R can compute $g^{\bar{r}\bar{k}}$ from $g^{\bar{r}}$ and $g^{\bar{k}}$ without knowing either $\bar{r}$ or $\bar{k}$, which contradicts with the hardness assumption of the CDH problem. Thus, j is the choice of R. By the above argument, no other $((\bar{y}/h^i)^{\bar{k}}, i)$, $i \neq j$, can be queried to the hash oracle with nonnegligible probability. All other $\bar{c}_i$, $i \neq j$, are distributed correctly. Therefore, the output of R' is computationally indistinguishable from the view of R.□

5 Threshold Oblivious Transfer

For a threshold t-out-of-p OT_n^1 (or (t, p)-OT_n^1) scheme, there are three types of parties: one sender S, p servers $S_1, S_2, \ldots, S_p$, and one receiver R. S has n secrets $m_1, m_2, \ldots, m_n$. It computes shares $m_{i,j}$, $1 \le j \le p$, of m_i, $1 \le i \le n$, and distributes shares $m_{i,j}$, $1 \le i \le n$, to server S_j, $1 \le j \le p$. Then, R chooses α, $1 \le \alpha \le n$, and contacts any t or more servers to get information about the shares. We assume a mechanism, such as the broadcast channel, for ensuring that R contacts servers with the same request. Otherwise, R can contact a set of t servers for m_α and another set of t servers for $m_{\alpha'}$. It is also possible to restrict R to contacting t servers at most. By the received information, R should be able to compute m_α and no others.

A (t, p)-OT_n^1 scheme should meet the following requirements [36]:

1. Correctness: If R and servers follow the protocol and R receives information from t or more servers, R can compute one m_α, where α is its choice.
2. Sender's privacy: Even if R receives information from t or more servers, it gains no information about any other m_i, $1 \le i \neq \alpha \le n$. Furthermore, if R

Scheme $(\mathbf{t},\mathbf{p})$-$\mathbf{OT_n^1}$:

- System parameters: (g, h, G_q);
- S_j's input: $m_{1,j}, m_{2,j}, \ldots, m_{n,j}$;
- R's choice: α, $1 \le \alpha \le n$;

1. R sends $y = g^r h^\alpha$ to t different servers $S_{j_1}, S_{j_2}, \ldots, S_{j_t}$, $r \in Z_q$;
2. Each S_{j_l}, $1 \le l \le t$, sends $c_{i,j_l} = (g^{k_{i,j_l}}, m_{i,j_l}(y/h^i)^{k_{i,j_l}})$, $1 \le i \le n$, to R;
3. By $c_{\alpha,j_l} = (a_{j_l}, b_{j_l})$, R computes shares $m_{\alpha,j_l} = b_{j_l}/a_{j_l}^r$, $1 \le l \le t$. Then, R interpolates these t shares to get

$$m_\alpha = \sum_{l=1}^{t} m_{\alpha,j_l} \Big(\prod_{1 \le d \ne l \le t} \frac{j_d}{j_d - j_l} \Big).$$

Fig. 5. Scheme (t,p)-OT_n^1.

receives information from less than t servers, it gains no information about any m_i, $1 \le i \le n$.

3. Receiver's privacy: There is a threshold t', $t' \ge 1$, such that no coalition of less than t' servers can gain any information about the choice α of R. The threshold t' should be as large as possible.
4. Security against receiver-server collusion: After R gets m_α, there is a threshold t'', $1 \le t'' \le t$, such that no coalition of less than t'' servers and R can gain any information about any other m_i, $1 \le i \ne \alpha \le n$. The threshold t'' should be as close to t as possible.

Our (t,p)-OT_n^1 scheme makes use of any threshold secret sharing scheme. It achieves $t' = \infty$ and $t'' = t$. Both are optimal. Let m_i be shared by the servers via polynomial $f_i(x)$ of degree t-1 such that $f_i(0) = m_i$, $1 \le i \le n$. Each server S_j, $1 \le j \le p$, holds the shares $m_{i,j} = f_i(j)$, $1 \le i \le n$. By contacting t servers, R can compute t shares of $m_{\alpha,j}$s and construct m_α. Our (t,p)-OT_n^1 scheme is as shown in Fig. 5.

The scheme in Fig. 5 is based on OT_n^1-I. We can construct similar schemes based on OT_n^1-II and OT_n^1-III, respectively.

Efficiency. The scheme takes only two rounds. R sends one message y to t servers and each contacted server S_j responds with n messages $c_{i,j}$, $1 \le i \le n$. For computation, R needs $t+1$ modular exponentiations for y and t shares m_{α,j_l}, $1 \le l \le t$, and one Lagrange interpolation for m_α. Each contacted server S_j needs $2n$ modular exponentiations for $c_{i,j}$, $1 \le i \le n$.

Correctness. If R contacts t or more servers, it can compute t shares m_{α,j_l} of m_α, $1 \le l \le t$. Therefore, it can compute m_α as shown in the scheme.

Security. Our (t,p)-OT_n^1 scheme has the following security properties:

1. Sender's privacy: If R contacts t or more servers, the privacy of m_i, $1 \le i \ne \alpha \le n$, is at least as strong as hardness of the DDH problem. (The proof is similar to that of Theorem 3.2.) Furthermore, if R gets information from less than t servers, R cannot compute information about any m_i, $1 \le i \le n$. This is guaranteed by the polynomial secret sharing scheme we use.
2. Receiver's privacy is unconditionally secure. Since, for any α', there is r' that satisfies $y = g^{r'} h^{\alpha'}$. Even if the servers have unlimited computing power, they cannot compute R's choice α.
3. It is secure against collusion of R and t-1 servers $S_{r_1}, S_{r_2}, \ldots, S_{r_{t-1}}$, assuming hardness of the DDH problem. Since, for R and S_{r_l}, $1 \le l \le t-1$, the privacy of shares $m_{i,j}$, $i \ne \alpha$, $j \ne r_1, r_2, \ldots, r_{t-1}$, is at least as strong as the hardness of the DDH problem, R and these t-1 servers cannot compute any information about other secrets m_i, $1 \le i \ne \alpha \le n$.

5.1 Access-Structure Oblivious Transfer

Let $\Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_z\}$ be a monotonic access structure over p servers $S_1, S_2, \ldots, S_p$. Each $\gamma_i = \{S_{i_1}, S_{i_2}, \ldots, S_{i_l}\}$ is an authorized set of servers such that all servers in γ_i together can construct the shared secret. Assume that n messages $m_1, m_2, \ldots, m_n$ are shared according to Γ by some secret sharing scheme $\mathcal{S}$ such that $S(\gamma) = (m_1, m_2, \ldots, m_n)$ if and only if $\gamma \in \Gamma$, where $\mathcal{S}(\gamma)$ means that $\mathcal{S}$ computes shared secrets from shares of the servers in γ.

We define Γ-OT_n^1 such that R can get the secret m_α from the servers in an authorized set $\gamma \in \Gamma$, where α is R's choice.

The requirements for a satisfactory Γ-OT_n^1 are the same as those for the threshold OT_n^1 schemes in Section 5.

We can combine our OT_n^1-I scheme and a general secret sharing scheme $\mathcal{S}$ to form a Γ-OT_n^1-I scheme as follows:

1. Let S_j obtain a share $m_{i,j}$ of m_i by the secret sharing scheme S, $1 \le i \le n$.
2. Let γ be an authorized set that R contacts its servers to obtain m_α. When R contacts $S_j \in \gamma$ with $y = g^r h^\alpha$, S_j responds with $c_{i,j} = (g^{k_{i,j}}, m_{i,j}(y/h^i)^{k_{i,j}})$, $1 \le i \le n$.
3. R computes $m_{\alpha,j}$ for each $S_j \in \gamma$ and applies $S(\gamma)$ to compute m_α.

6 Transformation of PIR to SPIR

One primary application of our techniques is a reduction from (single-database) private information retrieval (PIR) to symmetric PIR (SPIR). In PIR, a user U queries one data block from a database, but U does not want the database manager (DBM) to know which data block he is interested in [16]. PIR does not restrict U to obtain only one data block from the database. In SPIR, the DBM just releases the data block which U requests [28]. SPIR is equivalent to OT_n^1 with security against the malicious receiver.

Assume that the database has n data blocks m_i, $1 \le i \le m$, each is in G_q. The following, based on the technique of OT_n^1-III, transforms any PIR scheme into an SPIR scheme with security under the random oracle model:

1. U sends $y = g^r h^\alpha$ to DBM.
2. DBM computes $a = g^k$ and $c_i = m_i \oplus H((y/h^i)^k, i)$, $1 \le i \le n$, and treats c_is as its data blocks.
3. DBM and U perform a regular PIR protocol so that U obtains (a, c_α).
4. U computes $m_\alpha = c_\alpha \oplus H(a^r, \alpha)$.

If U's choice α of the basic PIR scheme in Step 3 is computationally secure, the transformed SPIR scheme's user privacy is computationally secure. On the other hand, if U's choice α is unconditionally secure, U's choice of the transformed SPIR is unconditionally secure.

The transformed SPIR scheme uses at most one more round than that of the basic PIR scheme because Step 1 may be combined with the first step of the basic PIR. Overall, if there exists a PIR scheme with computation complexity $t(n)$, message complexity $m(n)$, and round complexity $r(n)$, there exists an SPIR scheme with computation complexity $t(n) + n + 3$ modular exponentiations, message complexity $m(n) + 2$ (one for y and the other for a), and round complexity $r(n)$ or $r(n) + 1$, but with the additional assumptions of hardness of the CDH problem and the random oracle model.

We can use the technique of OT_n^1-II in the reduction so that the security is under the assumption of hardness of the DDH problem. But, it takes more time and exchanges more messages.

7 Conclusion

We have presented efficient string 1-out-of-n oblivious transfer schemes and extended them to threshold and access-structure oblivious transfer schemes for any $n \ge 2$. We have also presented its application on private information retrieval. It is interesting to find more applications of this construction.

For the schemes with security against the malicious receiver, three approaches are mentioned. One is based on WIPK, another is based on NIZKPK, and the other is based on the random oracle model. The one based on WIPK needs more rounds. The one based on NIZKPK needs a shared random string between the sender and the receiver. The one based on the random oracle model, though efficient and adopted in security analysis of cryptography widely, is not technically sound. It may be possible to replace the cryptographically strong hash function with a universal family of hash functions such that the random oracle model assumption is removed and the round efficiency is maintained.

Acknowledgments

Research supported in part by National Science Council grant 90-2213-E-009-152 and MOE Program of Promoting Academic Excellence of Universities under grant number 90-E-FA04-1-4, Taiwan, Republic of China.

References

[1] B. Aiello, Y. Ishai, and O. Reingold, "Priced Oblivious Transfer: How to Sell Digital Goods," *Proc. Advances in Cryptology (Eurocrypt '01)*, pp. 119-135, 2001.

[2] D. Beaver, "How to Break a 'Secure' Oblivious Transfer Protocols," *Proc. Advances in Cryptology (Eurocrypt '92)*, pp. 285-196, 1993.

[3] D. Beaver, J. Feigenbaum, J. Kilian, and P. Rogaway, "Locally Random Reductions: Improvements and Applications," *J. Cryptology*, vol. 10, no. 1, pp. 17-36, 1997.

[4] M. Bellare and S. Micali, "Non-Interactive Oblivious Transfer," *Proc. Advances in Cryptology (Crypto '89)*, pp. 547-557, 1990.

[5] M. Bellare and P. Rogaway, "Random Oracles Are Practical: A Paradigm for Designing Efficient Protocols," *Proc. First ACM Conf. Computer and Comm. Security*, pp. 62-73, 1993.

[6] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation," *Proc. 20th ACM Symp. Theory of Computing*, pp. 1-10, 1988.

[7] B. den Boer, "Oblivious Transfer ProtectingSecrecy," *Proc. Advances in Cryptology (Eurocrypt '90)*, pp. 31-45, 1991.

[8] G. Brassard and C. Crépeau, "Oblivious Transfers and Privacy Amplification," *Proc. Advances in Cryptology (Eurocrypt '97)*, pp. 334-346, 1997.

[9] G. Brassard, C. Crépeau, and J.-M. Robert, "Information Theoretic Reduction among Disclosure Problems," *Proc. 27th IEEE Symp. Foundations of Computer Science*, pp. 168-173, 1986.

[10] G. Brassard, C. Crépeau, and J.-M. Robert, "All-or-Nothing Disclosure of Secrets," *Proc. Advances in Cryptology (Crypto '86)*, pp. 234-238. 1987.

[11] G. Brassard, C. Crépeau, and M. Santha, "Oblivious Transfer and Intersecting Codes," *IEEE Trans. Information Theory*, vol. 42, no. 6, pp. 1769-1780, 1996.

[12] C. Cachin, "On the Foundations of Oblivious Transfer," *Proc. Advances in Cryptology (Eurocrypt '98)*, pp. 361-374, 1998.

[13] C. Cachin, C. Crépeau, and J. Marcil, "Oblivious Transfer with a Memory-Bounded Receiver," *Proc. 39th IEEE Symp. Foundations of Computer Science*, pp. 493-502, 1998.

[14] C. Cachin, S. Micali, and M. Stadler, "Computationally Private Informational Retrieval with Polylogarithmic Communication," *Proc. Advances in Cryptology (Eurocrypt '99)*, pp. 402-414, 1999.

[15] R. Canetti, O. Goldreich, and S. Halevi, "The Random Oracle Methodology, Revisited," *Proc. 30th ACM Symp. Theory of Computing*, pp. 209-218, 1998.

[16] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, "Private Information Retrieval," *J. ACM*, vol. 45, no. 6, pp. 965-982, 1998.

[17] C. Crépeau, "Equivalence between Two Flavors of Oblivious Transfers," *Proc. Advances in Cryptology (Crypto '87),* pp. 350-354, 1988.
[18] C. Crépeau, J. van de Graff, and A. Tapp, "Committed Oblivious Transfer and Private Multi-Party Computations," *Proc. Advances in Cryptology (Crypto '95),* pp. 110-123, 1995.
[19] G. Di Crescenzo, T. Malkin, and R. Ostrovsky, "Single Database Private Information Retrieval Implies Oblivious Transfer," *Proc. Advances in Cryptology (Eurocrypt '00),* pp. 122-138, 2000.
[20] Y.Z. Ding, "Oblivious Transfer in the Bounded Storage Model," *Proc. Advances in Cryptology (Crypto '01),* pp. 155-170, 2001.
[21] Y. Dodis and S. Micali, "Lower Bounds for Oblivious Transfer Reductions," *Proc. Advances in Cryptology (Eurocrypt '99),* pp. 42-45, 1999.
[22] T. ElGamal, "A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE Trans. Information Theory,* vol. 31, no. 4, pp. 469-472, 1985.
[23] S. Even, O. Goldreich, and A. Lempel, "A Randomized Protocol for Signing Contracts," *Comm. ACM,* vol. 28, pp. 637-647, 1985.
[24] P. Feldman, "A Practical Scheme for Non-Interactive Verifiable Secret Sharing," *Proc. 28th IEEE Symp. Foundations of Computer Science,* pp. 427-437, 1987.
[25] U. Feige and A. Shamir, "Witness Indistinguishable and Witness Hiding Protocols," *Proc. 22nd ACM Symp. Theory of Computing,* pp. 416-426, 1990.
[26] M.J. Fischer, S. Micali, and C. Rackoff, "A Secure Protocol for Oblivious Transfer (Extended Abstract)," *J. Cryptology,* vol. 9, no. 3, pp. 191-195, 1996.
[27] J.A. Garay and P.D. MacKenzie, "Concurrent Oblivious Transfer," *Proc. 41st IEEE Symp. Foundations of Computer Science,* pp. 314-324, 2000.
[28] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin, "Protecting Data Privacy in Private Data Retrieval Schemes," *Proc. 30th ACM Symp. Theory of Computing,* pp. 151-160, 1998.
[29] O. Goldreich and R. Vainish, "How to Solve Any Protocol Problem: An Efficient Improvement," *Proc. Advances in Cryptology (Crypto '87),* pp. 73-86, 1988.
[30] S. Goldwasser, S. Micali, and C. Rackoff, "The Knowledge Complexity of Interactive Proof Systems," *SIAM J. Computing,* vol. 18, no. 1, pp. 186-208, 1989.
[31] D.M. Gordon, "A Survey of Fast Exponentiation Methods," *J. Algorithms,* vol. 27, no. 1, pp. 129-146, 1998.
[32] J. Kilian, "Founding Cryptography on Oblivious Transfer," *Proc. 20th ACM Symp. Theory of Computing,* pp. 20-31, 1988.
[33] E. Kushilevitz and R. Ostrovsky, "Replication Is Not Needed: Single Database, Computationally-Private Informational Retrieval," *Proc. 38th IEEE Symp. Foundations of Computer Science,* pp. 364-373, 1997.
[34] M. Naor and B. Pinkas, "Oblivious Transfer and Polynomial Evaluation," *Proc. 31st ACM Symp. Theory of Computing,* pp. 145-254, 1999.
[35] M. Naor and B. Pinkas, "Oblivious Transfer with Adaptive Queries," *Proc. Advances in Cryptology (Crypto '99),* pp. 573-590, 1999.
[36] M. Naor and B. Pinkas, "Distributed Oblivious Transfer," *Proc. Advances in Cryptology (Asiacrypt '00),* pp. 205-219, 2000.
[37] M. Naor and B. Pinkas, "Efficient Oblivious Transfer Protocols," *Proc. 12th Ann. Symp. Discrete Algorithms,* pp. 448-457, 2001.
[38] V. Niemi and A. Renvall, "Cryptographic Protocols and Voting," *Result and Trends in Theoretical Computer Science,* pp. 307-316, 1994.
[39] T. P. Pedersen, "Non-Interactive and Information-Theoretical Secure Verifiable Secret Sharing," *Proc. Advances in Cryptology (Crypto '91),* pp. 129-140, 1991.
[40] M. Rabin, "How to Exchange Secrets by Oblivious Transfer," Technical Report TR-81, Aiken Computation Laboratory, Harvard Univ., 1981.
[41] A. Salomaa and L. Santean, "Secret Selling of Secrets with Several Buyers," *42nd EATCS Bulletin,* pp. 178-186, 1990.
[42] A. De Santis, G. Di Crescenzo, and G. Persiano, "Zero-Knowledge Arguments and Public-Key Cryptography," *Information and Computation,* vol. 121, no. 1, pp. 23-40, 1995.
[43] A. De Santis, G. Di Crescenzo, and G. Persiano, "Necessary and Sufficient Assumptions for Non-Interactive Zero-Knowledge Proofs of Knowledge for All NP Relations," *Proc. 27th Int'l Colloquium Automata, Languages, and Programming,* pp. 451-462, 2000.
[44] A. De Santis and G. Persiano, "Public-Randomness in Public-Key Cryptography," *Proc. Advances in Cryptology (Eurocrypt '90),* pp. 46-62, 1991.
[45] A. De Santis and G. Persiano, "Zero-Knowledge Proofs of Knowledge without Interactions," *Proc. 33rd IEEE Symp. Foundations of Computer Science,* pp. 427-436, 1992.
[46] J.P. Stern, "A New and Efficient All-or-Nothing Disclosure of Secrets Protocol," *Proc. Advances in Cryptology (Asiacrypt '98),* pp. 357-371, 1998.

Wen-Guey Tzeng received the BS degree in computer science and information engineering from National Taiwan University, Taiwan, in 1985 and the MS and PhD degrees in computer science from the State University of New York at Stony Brook in 1987 and 1991, respectively. He joined the Department of Computer and Information Science, National Chiao Tung University, Taiwan, in 1991, where he still works. His current research interests include cryptology and network security.