A Provably Secure Oblivious Transfer Protocol

Richard Berger (1)

René Peralta (2)

Tom Tedrick (2)

Computer Science Division

University of California

Berkeley, California.

ABSTRACT

The idea of the Oblivious Transfer, developed by Rabin, has been shown to have important applications in cryptography. M. Fischer pointed out that Rabin's original implementation of the Oblivious Transfer was not shown to be secure. Since then it has been an open problem to find a provably secure implementation. We present an implementation which we believe will simplify the development of secure cryptographic protocols. Our protocol is provably secure under the assumptions that factoring is hard and that the message is chosen at random from a large message space.

(1) Research sponsored in part by DARPA grant N00039-C-0235-9-83 and GTE fellowship

(2) Research sponsored in part by NSF grant MCS-82-04506

T. Beth, N. Cot, and I. Ingemarsson (Eds.): Advances in Cryptology - EUROCRYPT '84, LNCS 209, pp. 379-386, 1985.

1. Introduction

An Oblivious Transfer protocol (O.T.) is defined as a transfer of information from one party (Alice) to another (Bob) with the following properties:

1. Bob has a chance of $\frac{1}{2}$ of obtaining a message M.
2. The probability that Alice correctly guesses whether or not Bob obtained M is $\frac{1}{2}$.

The following implementation of O.T., based on the assumption that factoring is hard, was proposed by Rabin.[1] The message M is composed of two large primes p and q.

Rabin's Oblivious Transfer Protocol

step 1: Alice sends Bob $N = pq$.

step 2: Bob chooses a random number $x \in Z_N$ and sends $x^2 \ mod \ N$ to Alice.

step 3: Alice sends u to Bob where u is a square root of $x^2 \ mod \ N$.

A quadratic residue x^2 mod N has exactly four square roots. Distinct roots x, y such that $x \neq -y \ (mod \ N)$ are called *twin roots* of x^2. Given twin roots of x^2, it is possible to efficiently factor N (since GCD($x + y, N \neq 1$). If Bob and Alice follow the protocol, Bob has a chance of $\frac{1}{2}$ of obtaining twin roots of x^2, thus factoring N (obtaining p and q).

The following problem with Rabin's protocol has not been solved:[2]

> *It is conceivable that Bob has a routine P which chooses a quadratic residue r mod N such that given any root of r Bob can factor N.*

If Bob has P then he will always be able to factor N.

We present an O.T. protocol which is provably secure. In addition, our protocol can be used to send many messages under the same modulus N without compromising N's factorization. In applications of the O.T. it is important for Alice and Bob to obtain receipts so that a third party (i.e. a judge) can tell from these receipts whether or not Bob obtained M. The following problem arises:

> Once Bob has obtained the message, how can we prevent him from lying about the information that he originally sent to Alice? For example, if Bob obtains the factorization of N, he can lie about which root of $x^2 modN$ he originally had.

The only solution we know of for this problem in Rabin's protocol is as follows:

At step 2 Bob sends $x^2 = f(d)^2 \ mod \ N$, where f is a one-way function and d is randomly chosen from the domain of f. Then, after the protocol, Bob can prove to a judge that he knew $x = f(d)$ by displaying $d = f^{-1}(x)$.

Using one-way functions is clearly undesirable since the protocol cannot then be proven secure. In our protocol, the factorization of the modulus is never revealed. This makes it possible to solve the problem above without using one-way functions.

2. Terminology and Axioms.

Definition: A number $N = pq$, where $p \equiv q \equiv 3 \ mod \ 4$ are distinct primes and $|log(\frac{p}{q})| < 2$ is called a *Blum integer*.

Assumption 1 (about the model of computation) We assume Alice and Bob have computational power equivalent to a poly-time probabilistic Turing Machine (PTM).

Assumption 2 (Factoring Blum integers is hard): Let M be a poly-time PTM. Let ρ_n be the probability that M factors a random n-bit Blum integer. Then $\rho_n \to 0$ *as* $n \to \infty$.

Assumption 3 (about the message space): Every positive integer $< N$ is a valid message. However, Bob knows that the message M is drawn with a uniform probability distribution from a space of possible messages, MS, of size $\geq \alpha N$ for a fixed constant $0 < \alpha < 1$. MS is the set of integers in Z_N which have a non-zero probability of being chosen by Alice.

Definition: The *length* of a protocol is the total number of bits transferred between the parties in the protocol.

Definition: Whenever the set of possible messages is finite, it is very hard to guarantee that Bob will obtain the message with probability exactly $\frac{1}{2}$. This is true even if we assume that both parties follow the protocol, since Bob has a positive probability of simply guessing the message. Instead, we achieve probabilities which deviate by an arbitrarily small ϵ from $\frac{1}{2}$. We call this ϵ the *bias* of the implementation.

Definition: (In an O.T. implementation with bias ϵ) *Alice cheats Bob* if, when Bob follows the protocol, Alice, by deviating from the protocol, is able to:

i) determine with probability $> \frac{1}{2} + \epsilon$ whether or not Bob obtained M; or
ii) diminish Bob's chances of obtaining M to less than $\frac{1}{2} - \epsilon$.

Definition: (In an O.T. implementation with bias ϵ) *Bob cheats Alice* if, when Alice follows the protocol, Bob, by deviating from the protocol, is able to obtain M with probability $> \frac{1}{2} + \epsilon$.

Definition: An implementation of O.T. in which it is not possible for either Bob or Alice to cheat is called *secure*.

Given this terminology our goal is to describe an implementation of the O.T. with arbitrarily small bias.

3. A Provably Secure Oblivious Transfer Protocol.

Step 1: Alice sends a random n-bit Blum integer, N, to Bob.
Alice knows the factorization of N, but Bob does not.

Step 2: Alice convinces Bob that N is a Blum integer except for the fact that p and q might be raised to odd powers.
(See proof of theorem 4)

Step 3: Bob chooses a random integer $x \in Z_N$ and sends $x^2 \ mod \ N$ to Alice.

Step 4: Alice sends $M^2 \ mod \ N$, where M is her private message;
b = Jacobi symbol $\left(\frac{M}{N}\right)$; and a random root w of $M^2x^2 \ mod \ N$ to Bob.
{At this point the message is defined to be the unique root of $M^2 \ mod \ N$ less than $\frac{N}{2}$ and with Jacobi symbol b.}

Step 5: To insure that w is not junk, Bob verifies that $\frac{w^2}{x^2} \equiv M^2$ (mod N).
Then, if Jacobi symbol $\left(\frac{w/x}{N}\right)$ = b, Bob has the message.

Using well known number theoretical algorithms all computations required by the protocol can be done in polynomial time in n.

4. The protocol works when both parties follow the protocol.

First we show that, after step 4, Bob cannot factor N. For simplicity we ignore the Jacobi symbol $\left(\frac{M}{N}\right)$ since it is clear that it does not help Bob factor N.

We think of Bob as a poly-time PTM B with oracle A (Alice). Oracle A takes as input a pair (N,x^2) where N is an n-bit Blum integer and x^2 is a quadratic residue in Z_N and returns a random root of M^2x^2 where M is a random element in MS. The input to B is an n-bit Blum integer N. B contains a routine P(N) which returns a pair (x,x^2) where $x \in Z_N$. B is allowed to make *one* call A(N,x^2) to A *provided* x^2 was generated by P, i.e. provided Bob knows a root of x^2.

Theorem 1:

Let ψ_n be the probability that B factors N given that N is a random n-bit Blum integer. Then $\psi_n \rightarrow 0$ as $n \rightarrow \infty$.

Proof: Construct a PTM B^{smart} as follows:

INPUT: an n-bit Blum integer N.

B^{smart}: simulate B on input N until B makes the call A(N,x^2);
generate a random element M in Z_N;
assume A(N,x^2) returns Mx;
continue simulating B.

By assumption 3, the probability that M is in MS is α. Given that M is in MS the probability that $\pm$ Mx gets chosen as a root of M^2x^2 is $\frac{1}{2}$. Thus the probability ρ_n that B^{smart} factors N is $\geq \frac{1}{2}\alpha\psi_n$. But B^{smart} is a polytime PTM and so, by assumption 2, $\rho_n \rightarrow 0$. This implies $\psi_n \rightarrow 0$.∇

Theorem 2:

Assume both parties follow the protocol. Let ρ_n be the probability that Bob obtains M. Then $\rho_n \to \frac{1}{2}$ as $n \to \infty$.

Proof: The roots of $x^2M^2 \bmod N$ are $\pm$xM, and $\pm$xL where L, M are twin roots of $M^2 \bmod N$. The probability that Alice sends $\pm$xM is $\frac{1}{2}$. Thus $\rho_n \geq \frac{1}{2}$.

Let E_1 be the event that Bob factors N. Let $prob(E_1) = \psi_n$. Assume for simplicity that, given twin roots of $M^2 \bmod N$, Bob can factor N in 0 steps. Let E_2 be the event that Bob obtains M. Then

$$\begin{aligned} \rho_n &= prob(E_2) \\ &= prob(E_2 | E_1) * prob(E_1) + prob(E_2 | \neg E_1) * prob(\neg E_1) \\ &\leq prob(E_1) + prob(E_2 \mid \neg E_1) \\ &\leq \psi_n + prob(E_2 \mid \neg E_1). \end{aligned}$$

Now, given $\neg E_1$, Bob can obtain at most one root of M^2 less than $\frac{N}{2}$. Thus he will obtain M if and only if Alice sends $\pm$xM. The probability of this event is $\frac{1}{2}$. Thus $prob(E_2 \mid \neg E_1) = \frac{1}{2}$, which implies $\rho_n = prob(E_2) \leq \frac{1}{2} + \psi_n \to \frac{1}{2}$ by theorem 1 .$_\nabla$

Theorem 3:

Assume both parties follow the protocol. Let N be an n-bit Blum integer. Let ρ_n be the probability that Alice correctly guesses whether or not Bob obtained M. Then $\rho_n \to \frac{1}{2}$ as $n \to \infty$.

Proof: Let ψ_n be the probability that Bob factors N. If Alice guesses that Bob obtained M, then she is right if either Bob was able to factor N or Bob received $\pm$xM (probability $= \frac{1}{2}$). Thus she is right with probability p, where $\frac{1}{2} \leq p \leq \frac{1}{2} + \psi_n$. If Alice guesses that Bob did not obtain M, then she is right with probability $1 - p$, where $\frac{1}{2} - \psi_n \leq 1 - p \leq \frac{1}{2}$. Thus $\rho_n \in [\frac{1}{2} - \psi_n, \frac{1}{2} + \psi_n]$. By theorem 1, $\rho_n \to \frac{1}{2}$ as $n \to \infty$.$_\nabla$

Result

Theorems 1,2 and 3 prove that our protocol works for honest parties. Now we must show it is secure.

5. The protocol is secure

We will first assume Bob knows a root of $x^2 \bmod N$. Later we will drop this assumption.

Theorem 4:

Assume that at step 3 Bob knows a root of x^2. Then Alice can not cheat Bob, nor can Bob cheat Alice.

Proof: We look at possible deviations from the protocol and show that they are not useful or cannot be hidden.

Assume Alice follows the protocol. At step 2 Bob must send a quadratic residue because Alice has the factorization of N and can decide quadratic residuosity. Theorem 1 shows Bob obtains at most one root of M^2 independently of how he chose x. Thus not choosing x at random does not constitute cheating. This exhausts the possibilities of

Bob cheating.

Now assume Bob follows the protocol. We do not know of an efficient protocol by which Alice can prove to Bob that N is a Blum integer. However, the remainder of the proof relies only on the fact that N is the product of two distinct primes congruent to 3 mod 4, each raised to an odd power.

N is the product of two distinct primes congruent to 3 mod 4, each raised to an odd power if and only if the following 3 conditions are met:

a) The Jacobi symbol $\left(\frac{-1}{N}\right) = 1$.
b) N has exactly 2 distinct prime factors.
c) quadratic residues have roots with distinct Jacobi symbols.

The first condition is efficiently verifiable by Bob. Goldwasser and Micali[3] have shown that Alice can convince Bob (efficiently, securely and with exponentially small probability of error) that (b) holds. Blum[4] has shown Alice can convince Bob (efficiently, securely and with exponentially small probability of error) that (c) holds.

Now Bob knows that $M^2 \; mod \; N$ has exactly 2 roots less than $\frac{N}{2}$ and that these roots have opposite Jacobi symbols. At step 4 Alice defines the message to be the (unique) square root of $M^2 \; mod \; N$ which has Jacobi symbol b and is less than $\frac{N}{2}$. She cannot avoid sending a root of $M^2 \; mod \; N$, and she has no way of knowing which root she is actually sending .$\triangledown$

Theorem 4 assumes that Bob knows a root of $x^2 \; mod \; N$. The next theorem says Bob cannot cheat Alice at step 3 by sending a quadratic residue without knowing one of its roots.

Theorem 5:

Assume Alice follows the protocol. If, at step 3, Bob does not know a square root of x^2, yet he has probability $\geq \frac{1}{2}$ of obtaining M, then there exists an efficient probabilistic procedure to compute a root of $x^2 \; mod \; N$ with exponentially small probability of failure.

Proof: We think of Bob as a dishonest PTM $B^{dishonest}$ with oracle A. Recall that oracle A takes as input a pair (N,x^2) where N is an n-bit Blum integer and x^2 is a quadratic residue in Z_N and returns a random root of M^2x^2 where M is a random element in MS.

The input to $B^{dishonest}$ is an n-bit integer N. Since Bob is dishonest we must drop the requirement that the routine P(N) returns a root of the quadratic residue x^2. Thus P(N) will return only the quadratic residue x^2. $B^{dishonest}$ is allowed to make one call A(N,P(N)) to A.

Let ρ_n be the probability that $B^{dishonest}$ gets the message. We will use $B^{dishonest}$ to construct a parallel PTM B^{smart} which computes a root of $x^2 \; mod \; N$. The sequential version of B^{smart} runs in polynomial time and computes a root of $x^2 \; mod \; N$ with probability of failure $(1 - \frac{\alpha}{4})^r$ for an arbitrarily large constant r. The construction follows:

```
INPUT: an n-bit integer N.
$B^{smart}$:

simulate $B^{dishonest}$ until call A(N,P(N)) is made;
{Let $x^2 = P(N)$}
For each of r processors do
begin
    generate a random number $y \in Z_n$;   { $\sqrt{\frac{y^2}{x^2}}$ is called the "fake message" }
    Assume A(N,$x^2$) returns y ;

    continue simulating $B^{dishonest}$

    if $B^{dishonest}$ gets the fake message all processors stop;
end.
```

Lemma 1: If any of the r processors gets the fake message then B^{smart} knows a root of x^2.

Proof: The processor that gets the fake message can compute $\sqrt{x^2} = y(\sqrt{\frac{y^2}{x^2}})^{-1}$.▽

Lemma 2: The probability that a particular processor gets the fake message is $\geq \frac{\alpha}{4}$.

Proof: The probability that $z = \frac{y}{x}$ lies in MS is α. Given that z lies in MS, the probability that $\pm y$ gets chosen as a root of z^2x^2 is $\frac{1}{2}$. Given this event the probability that $B^{dishonest}$ obtains the fake message is (by assumption) $\geq \frac{1}{2}$. Thus the total probability that a particular processor gets the fake message $\geq \frac{\alpha}{4}$.▽

Thus the probability that no processor gets the fake message $\leq (1 - \frac{\alpha}{4})^r$. Therefore, by Lemma 1, B^{smart} obtains a root of $x^2 \ mod \ N$ with probability $1 - (1 - \frac{\alpha}{4})^r$.▽

Theorems 4 and 5 establish that our protocol is secure.

6. Generalizations

We state without proof that the following generalizations do not compromise the security of the O.T. protocol:

i) we may replace α by $\frac{1}{p(n)}$ for a fixed polynomial p.

ii) If the protocol is implemented "with receipts", i.e. Bob and Alice send a receipt for each message received, then Bob can prove to a third party whether or not he received M.

iii) Goldreich has proposed a version of the Oblivious Transfer in which Alice transfers to Bob exactly one out of two recognizable messages M_1 , M_2. Our protocol can be easily adapted to perform Goldreich's OT as follows :
Let XOR be the bitwise exclusive-or operator for bit vectors.
Let L be the twin root of M_1. Let $Y = L \ XOR \ M_2$.
(Notice that $M_2 = L \ XOR \ Y$)
At step 4 Alice sends Y along with b and $\sqrt{M_1^2x^2}$.

iv) if we wish to send many independently distributed messages, say q messages for a fixed integer q, we may replace steps 3, 4, 5 of the protocol by the loop:

```
for i:= 1 to q do
begin
    Step 3: Bob chooses a random integer $x_i \in Z_N$ and
            sends $x_i^2$ $mod$ $N$ to Alice.

    Step 4: Alice sends $M_i^2$ $mod$ $N$, where $M_i$ is her private message;
            b = Jacobi symbol $\left(\frac{M_i}{N}\right)$; and a random root $w$ of $M_i^2 x_i^2$ $mod$ $N$ to Bob.
            {At this point the message is defined to be the unique root of $M_i^2$ $mod$ $N$
             less than $\frac{N}{2}$ and with Jacobi symbol b.}

    Step 5: To insure that $w$ is not junk, Bob verifies that $\frac{w^2}{x_i^2} = M_i^2$.
              Then, if Jacobi symbol $\left(\frac{w/x_i}{N}\right)$ = b, Bob has the message.
end
```

7. Conclusions and Suggestions for Further Research

Thus we have developed a provably secure implementation of the Oblivious Transfer protocol. In our implementation it is essentially impossible for either Bob or Alice to successfully cheat. We have also shown that our implementation has certain properties which will make it an important building block for designing secure protocols. Essential to this research is the creation of a formal model of a protocol. Once this has been accomplished, one could prove theorems about the ways that various protocols can be combined so that the security of the implementation is not compromised.

Acknowledgements:

We are indebted to Manuel Blum for his insight and encouragement, and for providing a wonderful environment for research. We are also indebted to Umesh Vazirani for several enlightening discussions.

References

1. M. Rabin, *Private Communication* .

2. M. Fischer, *Private Communication through M. Blum.*

3. S. Goldwasser and S. Micali, *Proofs with Untrusted Oracles,* Department of Computer Science MIT and Department of Computer Science University of Toronto, 1983.

4. M. Blum, "Coin Flipping by Telephone," *Proc. IEEE COMPCON,* pp. 133-137, 1982.