Simple Zero-Knowledge Proof of Knowing $\sqrt{X_1}$ or $\sqrt{X_2}$ mod N

Sheng Zhong

Abstract

Tompa and Woll constructed a zero-knowledge proof of knowing a square root of X mod N, where N is the product of two large, secret primes. In this paper, we construct a zero-knowledge proof of knowing a square root of X_1 or X_2 mod N. Compared with the existing solution to this problem, ours is significantly simpler.

1 Introduction

Suppose that $N = PQ$, where P and Q are two large, secret prime numbers. Then the *Quadratic Residue Assumption* [3] states that there is no probabilistic polynomial-time algorithm for computing square roots with respect to modulus N. (For simplicity, in this paper we denote by $\sqrt{X}$ a square root of X mod N.) Now suppose that there is a public value $X \in Z_N^*$ such that Alice knows $\sqrt{X}$. Alice wants to convince Bob about this fact. Clearly, this could be a trivial task if she were willing to show $\sqrt{X}$ to Bob. However, for some reason, Alice is not willing to do so. So she needs to present a *zero-knowledge proof* [4] that she knows $\sqrt{X}$. Tompa and Woll [5] showed how to construct such a zero-knowledge proof. The proof has the following four steps: In the first step, Alice picks $r \in Z_N^*$ uniformly at random and computes $R = r^2 \bmod N$; she sends R to Bob. In the second step, Bob picks a random number $\beta \in \{0, 1\}$ uniformly at random and sends β to Alice. In the third step, Alice computes $z = \sqrt{X}^{\beta} r$ and sends z to Bob. In the fourth step, Bob verifies that $z^2 \equiv X^{\beta} R \pmod{N}$.

Received by the editors December 2007.

Communicated by A. Hoogewijs.

2000 *Mathematics Subject Classification :* Primary 11T71, 94A60.

Key words and phrases : Cryptography, interactive proof, zero-knowledge proof.

In one execution of the above zero-knowledge proof, with probability at most $\frac{1}{2}$ a dishonest Alice (who does not know $\sqrt{X}$) can cheat Bob. If we repeat the zero-knowledge proof for t times, the probability is reduced to at most $\frac{1}{2^t}$.

2 Our Problem

In this paper, we consider a more challenging problem. Suppose that there are public values $X_1, X_2 \in Z_N^*$, and that, for some reason, Alice knows the square root of one of them (with respect to modulus N). Assume that she is not even willing to reveal to Bob which of the two has the square root that she knows. Is it possible for her to prove this fact to Bob? In other words, is it possible for Alice to prove, in zero knowledge, that she knows $\sqrt{X_1}$ OR $\sqrt{X_2}$? In principle, this can be done using the general technique of proving first-order logic formulae [1]. Nevertheless, in this paper we give an alternative zero-knowledge proof, which is significantly simpler.

3 Our Solution

Now we present a four-step interactive proof system for the above problem. Just as the Tompa-Woll proof for knowing square root, our interactive proof can also be repeated to reduce the probability that a dishonest Alice can cheat Bob. In Section 4 we give a rigorous proof that this interactive proof system is zero-knowledge. In the first step, if Alice knows $\sqrt{X_1}$, then she picks $r_1, \overline{r}_2 \in Z_N^*$ uniformly and independently, and computes $R_1 = r_1^2 \bmod N$, $\overline{R}_2 = \overline{r}_2^2 \bmod N$, $R_2 = \overline{R}_2 / X_2 \bmod N$. If Alice knows $\sqrt{X_2}$, then she picks $\overline{r}_1, r_2 \in Z_N^*$ uniformly and independently, and computes $\overline{R}_1 = \overline{r}_1^2 \bmod N$, $R_1 = \overline{R}_1 / X_1 \bmod N$, $R_2 = r_2^2 \bmod N$. In both cases, Alice sends R_1, R_2 to Bob, *in a random order*. (Note that Alice knows $\sqrt{R_1 X_1}$ and $\sqrt{R_2 X_2}$ in both cases; in addition, she knows $\sqrt{R_1}$ if she knows $\sqrt{X_1}$, and she knows $\sqrt{R_2}$ if she knows $\sqrt{X_2}$.) In the second step, Bob picks a random number $\beta \in \{0, 1\}$ and sends β to Alice. In the third step, if $\beta = 0$, then Alice tells Bob the order in which she sent R_1 and R_2, computes $\overline{r}_1$ $(= \sqrt{R_1 X_1})$ and $\overline{r}_2$ $(= \sqrt{R_2 X_2})$ as follows and sends $\overline{r}_1, \overline{r}_2$ to Bob:

- If Alice knows $\sqrt{X_1}$, then she already knows $\overline{r}_2$; she only needs to compute $\overline{r}_1 = r_1 \sqrt{X_1}$.
- If Alice knows $\sqrt{X_2}$, then she already knows $\overline{r}_1$; she only needs to compute $\overline{r}_2 = r_2 \sqrt{X_2}$.

If $\beta = 1$, then Alice sends r to Bob, where $r = r_1$ when Alice knows $\sqrt{X_1}$ and $r = r_2$ when Alice knows $\sqrt{X_2}$. In the fourth step, if $\beta = 0$, then Bob verifies $\overline{r}_1^2 = R_1 X_1, \overline{r}_2^2 = R_2 X_2$. If $\beta = 1$, then Bob verifies that r^2 is equal to R_1 or R_2. It is important to note that, in the first step, Bob receives R_1 and R_2 in a random order. Consequently, if $\beta = 1$, then Bob does not know which of them is R_1 and which is R_2. Therefore, in the last step, Bob can't know whether r is a square root of R_1 or it is a square root of R_2. Otherwise, Bob would be able to find out whether Alice knows $\sqrt{X_1}$ or $\sqrt{X_2}$.

4 Security Analysis

Formally, we can treat Alice as a polynomial-time interactive machine A and Bob as another polynomial-time interactive machine B. Then (A, B) is an interactive proof system with common input $X = (X_1, X_2) \in (Z_N^*)^2$.

Definition 1. *(Perfect Zero-Knowledge, [2]) Let (A, B) be an interactive proof system. We say that (A, B) is* perfect zero-knowledge *if for every probabilistic polynomial-time interactive machine B^* there exists a probabilistic polynomial-time algorithm M^* such that for every common input X the following two conditions hold:*

1. *With probability at most $\frac{1}{2}$, on input X, machine M^* outputs a special symbol $\perp$ (i.e., $\Pr[M^*(X) = \perp] \leq \frac{1}{2}$).*
2. *Let $m^*(X)$ be a random variable describing the distribution of $M^*(X)$ conditioned on $M^*(X) \neq \perp$ (i.e., $\Pr[m^*(X) = \alpha] = \Pr[M^*(X) = \alpha | M^*(X) \neq \perp]$ for every $\alpha \in \{0, 1\}^*$). Then the following random variables are identically distributed:*
   - *$\langle A, B^* \rangle(X)$ (i.e., the output of the interactive machine B^* after interacting with the interactive machine A on common input X).*
   - *$m^*(X)$ (i.e., the output of algorithm M^* on input X, conditioned on it not being $\perp$).*

Machine M^* is called a perfect simulator *for the interaction of B^* with A.*

Theorem 2. *The interactive proof system presented in Section 3 is perfect zero-knowledge.*

Proof. For any B^*, we can construct a perfect simulator M^* as follows. (Note that M^* incorporates a copy of B^* such that M^* can interact with B^*.) At the very beginning, M^* picks $\beta' \in \{0, 1\}$ uniformly at random. If $\beta' = 0$, then M^* picks $\overline{r}_1, \overline{r}_2 \in Z_N^*$ uniformly and independently, and computes $\overline{R}_1 = \overline{r}_1^2$, $\overline{R}_2 = \overline{r}_2^2$, $R_1 = \overline{R}_1/X_1$, $R_2 = \overline{R}_2/X_2$. If $\beta' = 1$, then M^* picks $r_1, r_2 \in Z_N^*$ uniformly and independently and computes $R_1 = r_1^2 \bmod N$, $R_2 = r_2^2 \bmod N$. In both cases, M^* sends R_1, R_2 to B^*, in a random order. Next, M^* will receive β from B^*. If $\beta \neq \beta'$, M^* outputs $\perp$ and halts. Obviously, the probability of outputting $\perp$ is $\frac{1}{2}$. Otherwise, M^* proceeds as follows. If $\beta = 0$, then M^* tells B^* the order in which it sent R_1 and R_2, and sends $\overline{r}_1, \overline{r}_2$ to Bob. If $\beta = 1$, then M^* sends r to B^*, where $r = r_1$ or r_2, each with probability $\frac{1}{2}$. The output of M^* is defined as the output of the incorporated B^*. Clearly, the distribution of this output, conditioned on it not being $\perp$, is identical to the distribution of $\langle A, B^* \rangle(X)$. ∎

References

[1] R. Cramer, I. Damgaard, and B. Schoenmakers, Proofs of partial knowledge and simplified design of witness hiding protocols, In: Proc. Crypto'94, Lecture Notes in Computer Science, Vol. 839, Springer, Berlin, 1994, pp. 174-187.

[2] O. Goldreich, *Foundations of Cryptography*, Volume 1. Cambridge University Press, Cambridge, 2001.

[3] S. Goldwasser and S. Micali, Probabilistic Encryption, *Journal of Computer System Sciences* **28**(1984) 270–299.

[4] S. Goldwasser, S. Micali, and C. Rackoff, The Knowledge Complexity of Interactive Proof Systems, SIAM Journal on Computing, **18**(1989) 186-208.

[5] M. Tompa and H. Woll. Random Self-Reducibility and Zero Knowledge Interactive Proofs of Possession of Information, In: Proc. FOCS'87, IEEE Press, New York, 1987, pp. 472-482.

Computer Science and Engineering Department,
SUNY Buffalo, Amherst, NY 14260.
Email: szhong@cse.buffalo.edu