可應用於電子貨幣上之低計算量局部盲目簽章技術
Low-Computation Partially Blind Signatures for Electronic Cash

雷欽隆
Chin-Laung Lei

台灣大學電機工程系
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan, R.O.C.
Email: lei@cc.ee.ntu.edu.tw

范俊逸
Chun-I Fan

台灣大學電機工程系
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan, R.O.C.
Email: fan@crypto.ee.ntu.edu.tw

摘要

在局部盲目簽章系統中，簽章者可確保所有簽章都包含著其所需要的特定資訊。局部盲目簽章技術可應用於電子貨幣系統上抑制電子銀行中貨幣資料庫無限制的成長，而該資料庫是用來儲存所有使用過貨幣的必要設備。本論文提出一個具低計算量且可應用於電子貨幣上之局部盲目簽章技術。在本系統中，使用者只需進行數個模加法與模乘法，即可獲得其所要求的簽章。由於不須高指數與反元素的計算，本系統相當適合於行動通訊環境中的使用者，且與現有的系統比較，本系統降低了使用者所需的計算量幾乎達 95%。

關鍵詞：局部盲目簽章，電子貨幣，資訊安全

Abstract

In a secure partially blind signature scheme, the signer assures that the blind signatures issued by him contains the information he desires. The techniques make it possible to minimize the unlimited growth of the bank's database which storing all spent electronic cash when we use the techniques to construct anonymous electronic cash systems. In this paper, we propose an efficient partially blind signature scheme for electronic cash. In our scheme, only several modular additions and modular multiplications are required for a signature requester to obtain and verify a signature. It turns out that our scheme is suitable for mobile users or requesters because no exponentiation and inverse computations are required. Comparing with the existing blind signature schemes proposed in the literatures, our method reduces the amount of computations for signature requesters by almost 97%.

Keywords: Partially Blind Signatures, Electronic Cash, Information Security.

1. Introduction

Blind signatures are important techniques of modern cryptography since the techniques make it possible to protect the privacy of users and to prevent digital signatures from being forged [1, 4, 5, 14, 16, 23]. Typically, a blind signature scheme consists of two types of participants, a signer and a group of signature requesters. A signature requester requests signatures from the signer, and the signer computes and issues blind signatures to the requesters. There are two sets of messages known to the signer: (1) the signing results actually computed by the signer and (2) the signatures shown by the requesters for verification. The key point is that the actual correspondence between these two sets of messages is unknown to the signer. This property is usually referred to as the *unlinkability* property.

Owing to the unlinkability property, blind signature techniques have been widely used in an advanced communication service proposed in the literatures. That is the anonymous electronic cash system [6, 15, 16, 20]. In an electronic cash system, payers can pay electronic cash (e-cash) to payees through electronic communication networks. In this communication service, blind signatures are the key techniques to avoid the forgery of e-cash and to protect the privacy of payers [2, 3, 6, 12, 15, 16, 20].

Due to the characteristics of electronics, e-cash can be easily duplicated. Hence, to prevent a payer from double-spending his e-cash, the bank has to keep a database which stores all spent e-cash to check whether an e-cash has been double-spent or not by searching this database. This operation is referred to as the *freshness checking* (or the *double-spending checking*) of e-cash. Clearly, the database kept by the bank may grow unlimitedly [1, 6, 15, 16, 20].

The techniques of partially blind signatures make it possible to prevent the bank's database from growing unlimitedly. In a partially blind signature scheme, the bank (or signer, respectively) assures that the e-cash (or signatures, respectively) issued

by him contains the information he desires, such as the date information [1]. This property is referred to as the *partial blindness* property. Armed with the partial blindness property, we can deal with the unlimited growth problem of the bank's database, which storing all spent e-cash, in an electronic cash system. Hence, to guarantee the quality of this ever-growing popular communication service, an efficient and secure partially blind signature scheme is urgently desired.

In this paper, we propose a low-computation partially blind signature scheme based on quadratic residues [18, 30, 29]. Since only several modular additions and multiplications are performed by a signature requester in the proposed scheme, it is especially suitable for mobile clients and smart-card users. In the existing blind signature schemes of [1, 4, 5, 16, 23], it is necessary for a signature requester to perform modular exponentiation computations and inverse computations to obtain and verify a signature. Since these computations are time-consuming [8, 11, 19, 29], these schemes are impractical for the situations where computation capacities are limited such as smart cards and mobile units. Comparing with the blind signature schemes proposed in the literatures [1, 4, 5, 16, 23], our scheme reduces the amount of computations for signature requesters by almost 97% under a 1024-bit modulus.

Our proposed scheme is based on the theories of quadratic residues. Under a modulus n, x is a quadratic residue (QR) in Z_n^* if and only if there exists an integer y in Z_n^* such that $y^2 \equiv_n x$ where Z_n^* is the set of all positive integers less than and relatively prime to n [18, 29, 30]. Given n and x, it is computationally infeasible to compute the square root y of x in Z_n^* if n contains large prime factors and the factorization of n is unknown [26]. The security of our scheme depends on the difficulty of computing a square root of an integer in Z_n^* without the factorization of n, which has been proven to be as intractable as factorization [26].

The rest of the paper is organized as follows. In section 2, we present the proposed partially blind signature scheme. The performance of the scheme is examined in the section 3. Finally, a concluding remark is given in section 4.

2. The protocol

There are two kinds of participants, a signer and a group of requesters, in a partially blind signature scheme. A requester requests signatures from the signer, and the signer computes and issues partially blind signatures to the requesters. In addition to the unlinkability property, the signer has to ensure that any signature issued by him contains the common constant negotiated and agreed by the requesters and him in advance, and the requesters cannot change the common constant embedded in the signatures. That is the partial blindness property.

Our proposed partially blind signature scheme consists of four phases: (1) initialization, (2) requesting, (3) signing, and (4) extraction. In the initialization phase, the signer and requesters negotiate and agree on a common constant, and the signer publishes the necessary information. To obtain the signature of a message, a requester submits an encrypted version of the message to the signer in the requesting phase. In the signing phase, the signer computes the partially blind signature of the message, and then sends the result back to the requester. Finally, the requester extracts the signature from the result he receives in the extraction phase. The details of our partially blind signature scheme are described as follows.

(1) Initialization. The signer randomly selects two distinct large primes p_1 and p_2 where $p_1 \equiv_4 p_2 \equiv_4 3$. The signer computes $n = p_1 \cdot p_2$ and publishes n. Since $p_1 \equiv_4 p_2 \equiv_4 3$, given a QR in Z_n^*, there are four different square roots (or 2nd roots) of the QR in Z_n^*, and one of these roots is a QR in Z_n^*, too [29]. Hence, in addition to the 2nd roots of a QR in Z_n^*, we can derive the 4th roots, 8th roots, and (2^i)th roots of the QR in Z_n^* where i is an integer greater than 1. Such a special form of primes p_1 and p_2 does not affect the difficulty of factoring n [33]. In addition, let H be a public one-way hash function.

Let A with appropriate redundancy be the constant negotiated and agreed by requesters and the signer in advance, so that they can produce A independently, such as the date.

(2) Requesting. To request a signature of a plaintext m, a requester randomly chooses two integers u and v such that $\alpha = (H(m) \cdot (u^2 + Av^2) \bmod n)$ is in Z_n^*. Then the requester submits the integer α to the signer.

After receiving α, the signer randomly selects x such that $(\alpha \cdot (x^2 + A) \bmod n)$ is a QR in Z_n^*, and then sends the integer x to the requester.

After receiving x, the requester chooses an integer b in Z_n^* at random, and then computes $\delta = (b^2 \bmod n)$ and $\beta = (\delta \cdot (u - vx) \bmod n)$. The requester submits the integer β to the signer.

(3) Signing. After receiving β, the signer computes $\lambda = (\beta^{-1} \bmod n)$ and derives an integer t in Z_n^* such that

$$t^4 \equiv_n \alpha \cdot (x^2 + A) \cdot \lambda^2$$

since the signer knows the primes p_1 and p_2 [21, 26]. Hence, t is one of the 4th roots of $(\alpha \cdot (x^2 + A) \cdot \lambda^2 \bmod n)$ in Z_n^*. The signer sends (t, λ) to the requester.

(4) **Extraction.** After receiving the tuple (t, λ), the requester computes

$$\begin{cases} s = b \cdot t \mod n \\ c = \delta \cdot \lambda \cdot (ux + Av) \mod n. \end{cases}$$

The tuple (s, c) is a signature of m. To verify the signature (s, c) of m, one can examine if

$$s^4 \equiv_n H(m) \cdot (c^2 + A).$$

Theorem 1 ensures that a signature (s, c) of a plaintext m produced by the proposed blind signature scheme with the common constant A satisfies that $s^4 \equiv_n H(m) \cdot (c^2 + A)$.

Theorem 1 *If (s, c) is a signature of a plaintext m produced by the blind signature scheme of section 2 with the common constant A, then*

$$s^4 \equiv_n H(m) \cdot (c^2 + A).$$

Proof. By the Chinese remainder theorem [29], every integer w in Z_n^* can be represented by $< w_1, w_2 >$ where $w_1 = (w \bmod p_1)$ and $w_2 = (w \bmod p_2)$. For convenience, $< w_1, w_2 >$ is denoted by $< w >$ sometimes. For every $< k > = < k_1, k_2 >$ and $< w > = < w_1, w_2 >$ in Z_n^*, $< kw \bmod n > = < k_1 w_1 \bmod p_1, k_2 w_2 \bmod p_2 >$, and $< k^{-1} \bmod n > = < k_1^{-1} \bmod p_1, k_2^{-1} \bmod p_2 >$. In addition, for every $< k_1, k_2 >$ and $< w_1, w_2 >$ in Z_n^*, $< k_1, k_2 > = < w_1, w_2 >$ if and only if $k_1 \equiv_{p_1} w_1$ and $k_2 \equiv_{p_2} w_2$.

Let $\left[\frac{g}{h}\right]$ denote the Legendre symbol g over h where h is a prime [29]. Since both $(\alpha \cdot (x^2 + A) \bmod n)$ and $(\lambda^2 \bmod n)$ are QR's in Z_n^*,

$$\left[\frac{\alpha \cdot (x^2 + A) \cdot \lambda^2}{p_1}\right] = \left[\frac{\alpha \cdot (x^2 + A)}{p_1}\right] \left[\frac{\lambda^2}{p_1}\right] = 1 \cdot 1 = 1$$

and

$$\left[\frac{\alpha \cdot (x^2 + A) \cdot \lambda^2}{p_2}\right] = \left[\frac{\alpha \cdot (x^2 + A)}{p_2}\right] \left[\frac{\lambda^2}{p_2}\right] = 1 \cdot 1 = 1.$$

Therefore, we have that

$$\begin{aligned}
&\alpha \cdot (x^2 + A) \cdot \lambda^2 \bmod n \\
&\equiv_n \alpha \cdot (x^2 + A) \cdot \beta^{-2} \bmod n \\
&\equiv_n H(m) \cdot (u^2 + Av^2) \cdot (x^2 + A) \cdot (b^2(u - vx))^{-2} \\
&\equiv_n b^{-4} \cdot H(m) \cdot (u^2 + Av^2) \cdot (x^2 + A) \cdot (u - vx)^{-2} \\
&\equiv_n b^{-4} \cdot H(m) \cdot ((ux + Av)^2 + A(u - vx)^2) \cdot (u - vx)^{-2} \\
&\equiv_n b^{-4} \cdot H(m) \cdot ((ux + Av)^2 (u - vx)^{-2} + A) \\
&\equiv_n b^{-4} \cdot H(m) \cdot (((ux + Av)(u - vx)^{-1})^2 + A) \\
&\equiv_n b^{-4} \cdot H(m) \cdot ((b^2 b^{-2} (u - vx)^{-1} (ux + Av))^2 + A) \\
&\equiv_n b^{-4} \cdot H(m) \cdot ((\delta \cdot \lambda \cdot (ux + Av))^2 + A) \\
&\equiv_n b^{-4} \cdot H(m) \cdot (c^2 + A)
\end{aligned}$$

is a QR in Z_n^*. Since $\left[\frac{b^{-4}}{p_1}\right] = \left[\frac{b^{-4}}{p_2}\right] = 1$, the integer $(H(m) \cdot (c^2 + A) \bmod n)$ is also a QR in Z_n^*. Let $< d_1, d_2 >$ be one of the 4th roots of the integer $(H(m) \cdot (c^2 + A) \bmod n)$ in Z_n^*. Then the four 4th roots of that integer in Z_n^* are $< \pm d_1, \pm d_2 >$. Thus, the four 4th roots of $(b^{-4} \cdot H(m) \cdot (c^2 + A) \bmod n)$ in Z_n^* are $< \pm b_1^{-1} d_1, \pm b_2^{-1} d_2 >$. As $t^4 \equiv_n b^{-4} \cdot H(m) \cdot (c^2 + A)$, t belongs to $\{< \pm b_1^{-1} d_1, \pm b_2^{-1} d_2 >\}$. Since $s = (b \cdot t \bmod n)$, s is an element in $\{< \pm b_1 b_1^{-1} d_1, \pm b_2 b_2^{-1} d_2 > \} = \{< \pm d_1, \pm d_2 >\}$. It follows that s is a 4th root of the integer $(H(m) \cdot (c^2 + A) \bmod n)$ in Z_n^*. Hence, $s^4 \equiv_n H(m) \cdot (c^2 + A)$.

Q.E.D.

Based on the proposed protocol, an on-line electronic cash system can be constructed through the methods introduced in [6], where the signer of the blind signature protocol is regarded as the bank of the electronic cash scheme. An e-cash issued by the bank is of the form (s, m, c, A) where (s, c, A) is a signature of a plaintext m produced by our blind signature protocol with a common information A. Let (s, m, c, A) be an e-cash withdrawn by a payer from the bank by performing an electronic cash scheme based on our proposed protocol. To pay a payee the e-cash, the payer gives him (s, m, c, A). The payee verifies the correctness of the e-cash by examining if $s^4 \equiv_n H(m) \cdot (c^2 + A)$, and then he immediately calls the bank to verify if the e-cash is fresh. An e-cash is fresh if and only if the e-cash has not been deposited into the bank, i.e., the e-cash has not been spent. If the e-cash has not been spent previously, the payee accepts this payment, and deposits the e-cash into the bank. Then the bank stores the e-cash in its database. In other words, the bank has to record all the used e-cash in its database to check whether a specified e-cash has been spent or not. Hence, the bank's database may grow unlimitedly. With the help of the partial blindness techniques, the size of the bank's database can be controlled. Let the common information A contain an expiration date of e-cash in an electronic cash scheme based on our proposed partially blind signature protocol. If an e-cash (s, m, c, A) is with an expired A, then it cannot be used in any transaction. Hence, every e-cash (s, m, c, A) with an expired A recorded in the bank's database can be removed. Certainly, any fresh e-cash (s, m, c, A) with an unexpired A can be exchanged for another fresh e-cash (s', m', c', A') with a newer A' by performing another run of our proposed protocol.

In our scheme, the signer perturbs the message received from a requester before he signs it by using a random integer x. This is usually referred to as the *randomization* property [16]. A randomized blind signature scheme can withstand the chosen-text attacks [28]. Our scheme and the blind signature schemes of [4, 16, 23] possess the randomiza-

Table 1: Property Comparisons.

	Our Scheme	[1]	[4]*	[5]	[16]	[23]*
Mathematical Foundation	QR	RSA	DL/DL	RSA	RSA	DL/RSA
Unlinkability	Yes	Yes	Yes/Yes	Yes	Yes	Yes/Yes
Randomization	Yes	No	Yes/Yes	No	Yes	Yes/Yes
Partial Blindness	Yes	Yes	No/No	No	No	No/No

*Two blind signature schemes are proposed in [4] and [23].

tion property, while the blind signature schemes of [1, 5] do not have this property. In the requesting phase of our scheme, a requester chooses and submits an integer α to the signer, and then the requester receives the integer x from the signer. Let β' be an integer such that $\alpha \cdot (x^2 + A) \cdot \beta'^{-2} \equiv_n \alpha$. Then, $\beta'^2 \equiv_n (x^2 + A)$. Hence, β' is a square root of $((x^2 + A) \bmod n)$ in Z_n^*. Since the integer x is randomly chosen by the signer and n contains large prime factors, it is infeasible for the requester to compute a square root of $((x^2 + A) \bmod n)$ in Z_n^* during the requesting phase of the scheme without the factorization of n [26].

Given an integer c and a plaintext m, let s be an integer such that $s^4 \equiv_n H(m) \cdot (c^2 + A)$. Thus, s is a 4th root of the integer $(H(m) \cdot (c^2 + A) \bmod n)$ in Z_n^*. Since n contains large prime factors and these factors are unknown to the requesters, computing a 4th root of an integer in Z_n^* is computationally infeasible [26].

Given A and B in Z_n^*, one can efficiently compute g and h without the factorization of n such that $g^2 + Ah^2 \equiv B \pmod n$ through the lattice-based attack methods shown in [24]. However, given two integers A and B, it is still computationally infeasible to compute g without the factorization of n such that $g^2 + A \equiv B \pmod n$ because g is a square root of $(B - A \bmod n)$ in Z_n^*.

In the requesting stage of the scheme, the signer receives two integers α and β submitted by a signature requester for requesting a signature of a plaintext m, where

$$\begin{cases} \alpha = H(m) \cdot (u^2 + Av^2) \bmod n \\ \beta = b^2 \cdot (u - vx) \bmod n \end{cases}$$

Then in the extraction stage, the requester obtains a signature (s, c) of m, where

$$\begin{cases} s = b \cdot t \bmod n \\ c = (ux + Av) \cdot (u - vx)^{-1} \bmod n \end{cases}$$

and $t^4 \equiv_n \alpha \cdot (x^2 + A) \cdot \beta^{-2}$. The signer cannot link the tuple (α, β) to the signature (s, c) of m because the integers (u, v, b) are randomly selected and kept secret by the requester in the scheme.

In our scheme, by theorem 1, a correct signature (s, c) of a plaintext m with the constant A has to satisfy that $s^4 \equiv_n H(m) \cdot (c^2 + A)$. Since it is computationally infeasible to derive a square root of an integer in Z_n^* without the factorization of n [26], the requesters cannot change the common constant A embedded in their signatures. In our scheme, the signer ensures that all signatures issued by him contain the common constant A.

The comparisons of the properties between our scheme and the schemes of [1, 4, 5, 16, 23] are summarized in table 1. The mathematical foundation of our scheme is QR [26]. The security of the schemes of [1, 5, 16] depends on the RSA assumption [27], while the schemes of [4, 23] are based on the discrete logarithms (DL).

3. Performance

Typically, under a modulus n, the computation time for a modular exponentiation operation is about $O(|n|)$ times that of a modular multiplication where $|n|$ denotes the bit length of n [29]. The modulus n is usually taken from 512 bits to 1024 bits in a practical implementation [29]. In [8, 11, 19], some fast exponentiation algorithms are proposed. In [11], it requires $0.3381|n|$ modular multiplications and large amount of storage, e.g. 83370 stored values for a 512-bit modulus, to compute a modular exponentiation computation. An enhanced version of [11] is introduced in [8]. However, it still requires $0.3246|n|$ modular multiplications and large amount of storage, e.g. 36027 stored values for a 512-bit modulus, to compute a modular exponentiation computation [8]. The algorithm of [19] needs $(1.164|e| + 3)$ modular multiplications to compute $(x^e \bmod n)$ where $|e|$ is the bit length of e and $|e|$ has to be large enough (say 128 bits) in the RSA-type blind signature schemes of [1, 5, 16] to resist possible low-exponent attacks [9, 17, 31, 32].

In our partially blind signature scheme of section 2, no exponentiation and inverse computations are performed by signature requesters. Moreover, only several modular additions and multiplications are required for a requester to obtain and verify a signature.

In the blind signature schemes of [1, 4, 5, 16, 23], modular exponentiation computations and inverse

Table 2: Performance Comparisons.

	Our Scheme	[1]	[4]*	[5]	[16]	[23]*
No. of Exponentiation Computations	0	2	4	2	4	6
No. of Inverse Computations	0	1	2	1	1	0
No. of Hash Computations	2	2	0	2	2	2
No. of Multiplications	16	2	6	2	3	5
Computations Reduced:		97%	99%	97%	99%	99%

*The fastest scheme mentioned in the paper is selected for comparison in this table.

computations are needed for the requesters to obtain and verify signatures, while these time-consuming computations are not required in our scheme. The comparisons of the numbers of computations performed by a signature requester between our scheme and the schemes of [1, 4, 5, 16, 23] are summarized in table 2. Comparing with the schemes of [1, 4, 5, 16, 23], our scheme reduces the amount of modular computations for signature requesters by almost 97% under a 1024-bit modulus.

4. Conclusion

In this paper, we have proposed a low-computation partially blind signature scheme based on quadratic residues to minimize the bank's storage in an electronic cash system. Since no exponentiation and inverse computations are performed by signature requesters, our scheme is suitable for the situations where computation capacities are limited such as mobile clients and smart-card users. Comparing with the existing blind signature schemes, the computation amounts are greatly reduced for the requesters to obtain and verify signatures in our partially blind signature scheme.

References

[1] M. Abe and E. Fujisaki, "How to date blind signatures," *Advances in Cryptology-ASIACRYPT'96*, LNCS 1163, Springer-Verlag, 1996, pp. 244-251.

[2] S. Brands, "Untraceable off-line cash in wallets with observers," *Advances in Cryptology-CRYPTO'93*, LNCS 773, Springer-Verlag, pp. 302-318, 1993.

[3] J. Camenisch, J. M. Piveteau, and M. Stadler, "An efficient payment system protecting privacy," *Proceedings of ESORICS'94*, LNCS 875, Springer-Verlag, pp. 207-215, 1994.

[4] J. L. Camenisch, J. M. Pivereau, and M. A. Stadler, "Blind signatures based on the discret logarithm problem," *Advances in Cryptology-EUROCRYPT'94*, LNCS 950, Springer-Verlag, 1995, pp. 428-432.

[5] D. Chaum, "Blind signatures systems," *Advances in Cryptology-CRYPTO'83*, Plenum, 1983, p. 153.

[6] D. Chaum, A. Fiat, and M. Naor, "Untraceable electronic cash," *Advances in Cryptology-CRYPTO'88*, LNCS 403, Springer-Verlag, 1990, pp. 319-327.

[7] D. Chaum and T. Pedersen, "Wallet databases with observers," *Advances in Cryptology-CRYPTO'92*, LNCS 740. Springer-Verlag, pp. 89-105, 1992.

[8] C. Y. Chen, C. C. Chang, and W. P. Yang, "Hybrid method for modular exponentiation with precomputation," *Electronics Letters*, vol. 32, no. 6, 1996, pp. 540-541.

[9] D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter, "Low-exponent RSA with related messages," *Advances in Cryptology-EUROCRYPT'96*, LNCS 1070, Springer-Verlag, 1996, pp. 1-9.

[10] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, 1976, pp. 644-654.

[11] V. Dimitrov and T. Cooklev, "Two algorithms for modular exponentiation using nonstandard arithmetics," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E78-A, no. 1, 1995, pp. 82-87.

[12] T. Eng and T. Okamoto, "Single-term divisible electronic coins," *Advances in Cryptology-EUROCRYPT'94*, LNCS 950, Springer-Verlag, 1995, pp. 306-319.

[13] A. Evans, W. Jr. Kantrowitz, and E. Weiss, "A user authentication scheme not requiring secrecy in the computer," *Communications of the ACM*, vol. 17, no. 8, 1974, pp. 437-442.

[14] C. I. Fan and C. L. Lei, "A multi-recastable ticket scheme for electronic elections," *Advances in Cryptology-AISACRYPT'96*, LNCS 1163, Springer-Verlag, 1996, pp. 116-124.

[15] C. I. Fan and C. L. Lei, "Secure Rewarding Schemes," *Proceedings of the Thirtieth Annual*

Hawaii International Conference on System Sciences, vol. 3, 1997, pp. 571-580.

[16] N. Ferguson, "Single term off-line coins," *Advances in Cryptology-EUROCRYPT'93*, LNCS 765, Springer-Verlag, 1994, pp. 318-328.

[17] J. Hastad, "On using RSA with low exponent in a public key network," *Advances in Cryptology-CRYPTO'85*, LNCS 218, Springer-Verlag, 1985, pp. 403-408.

[18] W. J. LeVeque, *Fundamentals of Number Theory*, Addison-Wesley, Reading, Mass., 1977.

[19] D. C. Lou and C. C. Chang, "Fast exponentiation method obtained by folding the exponent in half," *Electronics Letters*, vol. 32, no. 11, 1996, pp. 984-985.

[20] T. Okamoto and K. Ohta, "Universal electronic cash," *Advances in Cryptology-CRYPTO'91*, Springer-Verlag, LNCS 576, 1992, pp. 324-337.

[21] R. C. Peralta, "A simple and fast probabilistic algorithm for computing square roots modulo a prime number," *IEEE Transactions on Information Theory*, vol. 32, no. 6, 1986, pp. 846-847.

[22] S. Pohlig and M. E. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance," *IEEE Transactions on Information Theory*, vol. 24, 1978, pp. 106-110.

[23] D. Pointcheval and J. Stern, "Provably secure blind signature schemes," *Advances in Cryptology-ASIACRYPT'96*, LNCS 1163, Springer-Verlag, 1996, pp. 252-265.

[24] J. M. Pollard and C. P. Schnorr, "An efficient solution of the congruence $x^2 + ky^2 = m \pmod{n}$," *IEEE Transactions on Information Theory*, vol. 33, no. 5, 1987, pp. 702-709.

[25] G. P. Purdy, "A high security log-in procedure," *Communications of the ACM*, vol. 17, no. 8, 1974, pp. 442-445.

[26] M. O. Rabin, "Digitalized signatures and public-key functions as intractable as factorization," *Technical Report*, MIT/LCS/TR212, MIT Lab., Computer Science, Cambridge, Mass. Jan. 1979.

[27] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, 1978, pp. 120-126.

[28] A. Shamir and C. P. Schnorr, "Cryptanalysis of certain variants of Rabin's signature scheme," *Information Processing Letters*, vol. 19, 1984, pp. 113-115.

[29] G. J. Simmons, *Contemporary Cryptology: The Science of Information Integrity*, IEEE Press, N.Y., 1992.

[30] I. M. Vinogradov, *An Introduction to the Theory of Numbers*, Pergamon Press, Elmsford, N.Y., 1955.

[31] M. J. Wiener, "Cryptanalysis of short RSA secret exponents," *IEEE Transactions on Information Theory*, vol. 36, 1990, pp. 553-558.

[32] H. C. Williams and B. Schmid, "Some remarks concerning the MIT public-key cryptosystem," *BIT*, vol. 19, 1979, pp. 525-538.

[33] H. C. Williams, "A modification of the RSA public-key encryption procedure," *IEEE Transactions on Information Theory*, vol. 26, no. 6, 1980, pp. 726-729.