JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 18, 47-58 (2002)

Short Paper

A User Efficient Fair Blind Signature Scheme for Untraceable Electronic Cash

CHUN-I FAN AND CHIN-LAUNG LEI[*]
Telecommunication Laboratories
Chunghwa Telecom Co., Ltd.
Taoyuan, 326 Taiwan
E-mail: chunifan@ms35.hinet.net
[*]*Department of Electrical Engineering*
National Taiwan University
Taipei, 107 Taiwan
E-mail: lei@cc.ee.ntu.edu.tw

Blind signatures have been widely adopted to construct untraceable electronic cash systems since they are both unlinkable and unforgeable. Although unlinkability protects the privacy of customers and users, it may be abused by criminals for such purposes as to launder money or to safely get a ransom. The techniques of fair blind signatures are developed to deal with the abuse of unlinkability. In this paper we propose a user efficient fair blind signature scheme which makes it possible for a government or a judge to recover the link between a signature and the instance of the signing protocol which produces that signature when the unlinkability property is abused. Only two integers are required to form a signature in the proposed fair blind signature scheme. Furthermore, it only takes several modular multiplications for a user to obtain and verify a signature. It turns out that the scheme is suitable for situations where computation capability of users or customers is limited, such as smart cards and mobile units. Compared with existing blind signature schemes proposed in the literatures, our method reduces the computation required of users by more than 99%.

Keywords: blind signatures, electronic cash, cryptology, privacy, information security

1. INTRODUCTION

Due to the fast pace of computer and network technologies, many advanced communication services have been proposed to take advantage of ever-growing networking capabilities. Among these services, electronic cash is popular since the technique makes it possible for a customer to electronically transmit cash (e-cash) through communication

Received July 7, 1999; revised December 6, 1999; accepted March 22, 2000.
Communicated by Chi Sung Laih.

networks. Owing to their ability to protect the privacy of customers, blind signature techniques are usually taken as the underlying foundations.

A typical blind signature scheme consists of two kinds of participants, a signer and a group of users. A user requests signatures from the signer, and the signer issues blind signatures to the users. There are two sets of messages known to the signer: the messages received from users for signatures, and the signatures shown by the users for later verification. The key point is that the actual correspondence between these two sets of messages is unknown to the signer. This is usually referred to as the *unlinkability* property. Due to the unlinkability property, blind signatures have been widely used in untraceable electronic cash systems [1-7] and in anonymous electronic voting systems [8-11].

Since blind signatures provide perfect unlinkability, it is computationally infeasible for anyone but the user himself to link a signature to the instance of the signing protocol which produces that signature. In electronic cash systems, the unlinkability property might be abused by criminals, e.g., to launder money or to safely get a ransom [12, 13]. Hence, robust blind signatures should possess the following properties to withstand the possible abuse of unlinkability:

(1) If users or customers are engaged in legal commercial transactions or payments, the unlinkability is preserved against the signer or the bank; on the other hand, if they abuse the unlinkability property, then a government or a judge will have enough information to link the signatures shown by the users or customers for verification to the instances of the signing protocols which produce those signatures. This property is the *fairness* property.

(2) The addition of the fairness property to blind signatures cannot significantly increase the computation load on users or customers since their computation capability is limited, such as smart-card users and mobile clients.

In this paper, we propose a user-efficient fair blind signature scheme for untraceable electronic cash to meet the above two requirements. In our scheme, with the help from the judge, the government, or a trusted party, it is possible to link a signature to its corresponding instance of the signing protocol when the unlinkability is abused. The proposed scheme only takes several modular multiplications for a user to obtain and verify a signature. Compared with existing schemes, our method greatly reduces users' computational load.

2. PRELIMINARY

Typically, a blind signature protocol consists of four phases: initializing, blinding, signing, and unblinding [6, 14-16]. In the initializing phase, the signer publishes some necessary information. To request the signer's signature on a message, a user blinds the message via an encryption-like process, and then submits the blinded message to the signer in the blinding phase. In the signing phase, the signer computes the signature on the blinded message, and then sends the signing result called the blind signature to the user. Finally, in the unblinding phase, the user unblinds the blind signature to obtain the exact signature on the message he had chosen. In a secure blind signature scheme, it is computationally infeasible for the signer to derive the link between the blind signature and the exact signature on the same message. This is the unlinkability property.

To cope with the possible abuse of the unlinkability property, Stadler, Piveteau, and Camenisch proposed three fair blind signature schemes [13]. The first scheme of [13] is based on Chaum’s blind signatures and the cut-and-choose method [4, 15]. The second one is based on a variation of the Fiat-Shamir signature scheme and the concept of one-out-of-two oblivious transfers [22, 23]. The main idea of the third scheme in [13] is that the user has two pseudonyms registered with the judge. One of the pseudonyms is used during the signing protocol, while the other one is part of the signature. Thus, the judge, who knows the two corresponding pseudonyms, can link a view of the signing protocol and the corresponding signature.

In the fair blind signature scheme using the cut-and-choose method of [13], a large amount of data is exchanged during the signing protocol, and the resulting signature is quite large. Although the resulting signature produced by the fair blind signature scheme using oblivious transfer in [13] is short, it is necessary for a user to perform a large amount of modular computations. Considering the fair blind signature scheme with registration of [13], a large amount of computation is still required of users.

3. A USER EFFICIENT FAIR BLIND SIGNATURE SCHEME

In this section we propose a user efficient fair blind signature scheme for untraceable electronic cash. The proposed scheme is based on the theory of quadratic residues [24-27]. Under a modulus n, an integer x is a quadratic residue (QR) in Z_n^* if and only if there exists an integer y in Z_n^* such that $y^2 \equiv x \pmod{n}$ where Z_n^* is the set of all positive integers which are less than and relatively prime to n. Given x and n, it is computationally infeasible to derive y if n contains large prime factors and the factorization of n is unknown [26]. The details of the scheme are described as follows.

3.1 Initializing

The signer randomly selects two distinct large primes p_1 and p_2 such that $p_1 \equiv p_2 \equiv 3 \pmod{4}$. The signer computes $n = p_1p_2$ and then publishes n. Since $p_1 \equiv p_2 \equiv 3 \pmod{4}$, 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 square roots is a QR in Z_n^*, too [27]. 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 [28]. Also, let F and H be two public one-way hash functions where the range (or image) of F contains all of the positive integers less than n [27, 29, 30].

The judge randomly chooses two distinct large primes p_3 and p_4 such that $p_3 \equiv p_4 \equiv 3 \pmod{4}$ and $p_3p_4 > n$, and then computes $\hat{n} = p_3 p_4$. The judge publishes $\hat{n}$ and a string ϖ selected by itself.

3.2 Blinding

A user randomly chooses three integers y_1, y_2, and y_3 such that for every i with $1 \le i \le 3$,

$$\begin{cases} y_i \in Z_{\hat{n}}^*, \\ y_i \bmod n \in Z_n^*, \\ n < y_i < \hat{n} < y_i^2, \ \text{and} \\ \varpi \ \text{is a prefix of } y_i. \end{cases}$$

Then, the user computes $q_i = (y_i^2 \bmod \hat{n})$ and submits q_i to the judge for $i = 1, 2$, and 3.

After receiving all (q_i)'s, the judge derives the square roots of q_i in $Z_{\hat{n}}^*$ for $i = 1, 2$, and 3 [25, 26]. For $i = 1, 2$, and 3, there exists one square root with the prefix ϖ of q_i in $Z_{\hat{n}}^*$. This enables the judge to obtain y_1, y_2, and y_3, respectively, by finding the square roots with the prefix ϖ of q_1, q_2, and q_3, respectively, in $Z_{\hat{n}}^*$.

The judge randomly selects two integers β and γ, and forms $u = F(\beta)$ and $v = F(\gamma)$ such that $((u^2 + v^2) \bmod n)$ is in Z_n^*. Let z be an integer to uniquely identify this instance of the protocol, where z is chosen by the judge such that $F(z)$ is a QR in $Z_{\hat{n}}^*$.[1] The integer z is referred to as the *identifier* of this instance of the protocol. The judge derives a square root $\hat{z}$ of $F(z)$ in $Z_{\hat{n}}^*$ such that $(\hat{z})^2 \equiv F(z) \pmod{\hat{n}}$. The judge randomly selects an integer b in Z_n^*, and then computes

$$\begin{cases} \hat{b} = y_1^{-1} b \mod n \\ \hat{u} = y_2^{-1} u \mod n \\ \hat{v} = y_3^{-1} v \mod n. \end{cases} \tag{1}$$

The judge sends the 5-tuple $(\hat{b}, \hat{u}, \hat{v}, \hat{z}, z)$ to the user, and stores the 4-tuple (β, γ, b, z) in its database.

The user can obtain b, u, and v by computing

$$\begin{cases} b = (y_1 \hat{b} \mod n) \\ u = (y_2 \hat{u} \mod n) \\ v = (y_3 \hat{v} \mod n). \end{cases} \tag{2}$$

The user chooses a message m such that $H(m)$ is in Z_n^*, and computes $\alpha = (H(m)(u^2 + v^2) \bmod n)$. Then, he submits the triple $(\alpha, z, \hat{z})$ to the signer.

After verifying that $(\hat{z})^2 \equiv F(z) \pmod{\hat{n}}$, the signer randomly selects an integer δ and computes $x = F(\delta)$ such that $(\alpha(x^2+1) \bmod n)$ is a QR in Z_n^*, and then sends $(x, z, \hat{z})$ to the judge.

[1] Since $\hat{n} = p_3 p_4$ and $p_3 \equiv p_4 \equiv 3 \pmod 4$, $(p_3 - 1)(p_4 - 1)/4$ elements are QR's in $Z_{\hat{n}}^*$ [27]. The probability that a randomly-chosen integer is a QR in $Z_{\hat{n}}^*$ is about 1/4.

After verifying that $(\hat{z})^2 \equiv F(z)$ (mod $\hat{n}$) and that z has not been received from the signer in previous instances of the protocol, the judge retrieves the stored (β, γ, b, z) through the identifier z. The judge computes c = (($ux + v$)($u - vx$)$^{-1}$ mod n), where u = $F(\beta)$ and v = $F(\gamma)$, and checks if the integer c is different from all the other c's which are recorded by the judge in the previous instances of the protocol. If true, the judge computes λ = (b^2($u - vx$) mod n) and sends λ to the signer.[2] Then, the judge records the 5-tuple (β, γ, b, c, z).

3.3 Signing

After receiving λ, the signer computes e = (λ^{-1} mod n) and derives an integer t in Z_n^* [25, 26] such that

$$t^4 \equiv \alpha(x^2 + 1)e^2 \pmod{n} \qquad (3)$$

The signer sends the triple (e, t, x) to the user, and stores (δ, z).

3. 4 Unblinding

After receiving the triple (e, t, x), the user computes

$$\begin{cases} s = bt \mod n \\ c = b^2 e(ux + v) \mod n. \end{cases} \qquad (4)$$

Thus, (c, s) is a valid signature on the message m. To verify the signature (c, s) on m, one can check that

$$s^4 \equiv H(m)(c^2 + 1) \pmod{n}. \qquad (5)$$

4. DISCUSSIONS

In this section we examine the security and performance of the fair blind signature protocol proposed in section 3.

4.1 Correctness

The following theorem ensures that a signature (c, s) on m produced by the proposed blind signature scheme satisfies formula (5).

[2] If (($u - vx$)$^{-1}$ mod n) does not exist in Z_n^* or the integer c is not unique among all the recorded c's, the judge requests the signer to choose another integer x and repeat this protocol. However, the modulus n is about 1024 or more bits in a practical implementation and the integers u, v, x are randomly chosen by the judge and the signer, so that the probability that (($u - vx$)$^{-1}$ mod n) does not exist in Z_n^* or the integer c is not unique is usually slight.

Theorem 1 If (c, s) is a signature on m produced by the fair blind signature scheme in section 3 under the modulus n, then formula (5) is true.

Proof. The proof of theorem 1 is shown in the appendix.

4.2 Unlinkability

In the blinding phase of the protocol, the signer receives two integers α and λ from the user and the judge for requesting a signature on a message m. Then in the unblinding phase of the protocol, the user obtains a signature (c, s) on the message m.

For every instance, i, of the signing protocol in the proposed scheme of section 3, the signer can record the parameters (α_i, λ_i) received from the user. The triple (α_i, λ_i, x_i) is said to be the *view* of the signer to the instance i of the signing protocol. Thus, we have the following theorem.

Theorem 2 Given a triple (c, m, s) produced by the protocol in section 3, the signer can derive $b_i^{'}, u_i^{'}$, and $v_i^{'}$ for every view (α_i, λ_i, x_i) such that

$$\begin{cases} c \equiv (u_i^{'} x_i + v_i^{'})(u_i^{'} - v_i^{'} x_i)^{-1} \pmod{n} \\ \alpha_i \equiv H(m)((u_i^{'})^2 + (v_i^{'})^2) \pmod{n} \\ \lambda_i \equiv (b_i^{'})^2 (u_i^{'} - v_i^{'} x_i) \pmod{n}. \end{cases} \tag{6}$$

Proof. The proof of theorem 2 is shown in the appendix.

Hence, given a triple (c, m, s) produced by the protocol in section 3, the signer can always derive three blinding factors $b_i^{'}, u_i^{'}$, and $v_i^{'}$ for every recorded (α_i, λ_i, x_i). It turns out that all of the signatures produced by the proposed protocol are indistinguishable from the signer's point of view. Therefore, it is computationally infeasible for the signer to derive the link between an instance i of the signing protocol and the signature produced by that instance of the protocol. This is the unlinkability property.

In our scheme, the signer does not perform any signing operation without receiving a valid tuple $(z, \hat{z})$. To request a valid signature on a message m, it is necessary for a user to submit to the signer a tuple $(z, \hat{z})$ obtained from the judge with $(\hat{z})^2 \equiv F(z) \pmod{\hat{n}}$ The tuple $(z, \hat{z})$ cannot be used twice by users or the signer. If $(z, \hat{z})$ is reused, it can be detected by the judge in the blinding phase of the protocol.

4.3 Linkage Recovery

Consider the linkage recovery in the proposed scheme. Given a signature $(\tilde{c}, \tilde{s})$ on a message $\tilde{m}$ produced by some instance of the protocol, the judge can retrieve the unique 5-tuple (β, γ, b, c, z) with $c = \tilde{c}$ from its database. Hence, the signature $(\tilde{c}, \tilde{s})$ on $\tilde{m}$ is produced by the instance with identifier z of the protocol. If the judge reveals the 4-tuple (β, γ, c, z) to the signer, then the signer can retrieve the tuple (δ, z) through z from his database, where $c \equiv (F(\beta)F(\delta) + F(\gamma))(F(\beta) - F(\gamma)F(\delta))^{-1}$ (mod n). Thus, the signer can verify that the instance with identifier *z* of the protocol produces the signature $(\tilde{c}, \tilde{s})$ on $\tilde{m}$. Therefore, if the judge reveals appropriate information to the signer, the link between an instance of the signing protocol and the corresponding signature can be established by the signer. On the other hand, given the signature $(\tilde{c}, \tilde{s})$ on $\tilde{m}$, the signer cannot find the instance of the protocol which produces that signature without the help from the judge. This is the fairness property.

4.4 Unforgeability

In the proposed scheme, the signer perturbs the message received from every user before he signs it by using a random integer *x*. This is usually referred to as the *randomization* property [6]. A randomized blind signature scheme can withstand chosen-text attacks [32]. Our scheme and the blind signature schemes of [6, 13, 14, 16, 21] possess the randomization property, while the blind signature scheme of [15] does not.

The underlying foundation of the proposed protocol in section 3 is Fan-Lei's blind signature scheme which was first presented in [9] and improved in [31]. Given a tuple (*c*, *m*), the difficulty of deriving an integer *s* in Z_n^* such that formula (5) is true depends on the security of [31].

The comparisons of the properties between our scheme and the existing schemes of [6, 13-16, 21] are summarized in Table 1. The mathematical foundation of our scheme and [21] is QR [26]. The security of the schemes of [6, 13, 15, 16] depends on the RSA assumption [17], while the schemes of [13, 14, 16] are based on discrete logarithms (DL).

Table 1. Property comparisons.

	Ours	[6]	[13][1]	[14][2]	[15]	[16][2]	[21][2]
Foundation	QR	RSA	RSA/DL/DL	DL/DL	RSA	RSA/DL	QR/QR
Randomization	Yes	Yes	No/Yes/Yes	Yes/Yes	No	Yes/Yes	Yes/Yes
Unlinkability	Yes	Yes	Yes/Yes/Yes	Yes/Yes	Yes	Yes/Yes	Yes/Yes
Fairness	Yes	No	Yes/Yes/Yes	No/No	No	No/No	No/No

4.5 Performance

In the typical square-multiply algorithm [27] under a modulus *n*, the computation time for a modular exponentiation operation is about 1.5|*n*| times that of a modular multiplication, where |*n*| denotes the bit length of *n*. In addition, an inverse computation takes about the same amount of time as that of a modular exponentiation computation under a common modulus [27]. Compared with the fair blind signature schemes of [13], if we take a 1024-bit modulus *n*, our scheme reduces the computations for users by more than 99%. Table 2 summarizes the numbers of modular computations performed by a user of our scheme and of the fair blind signature schemes of [13]. The comparisons of the storage of signatures between our scheme and the schemes of [13] are also summarized in Table 2.

In the proposed scheme, the signer is required to record (δ, z) for every instance of the signing protocol and to perform one inverse, four QR tests, and one 4th root computation to produce a signature. The signer performs complicated computations in the proposed scheme and in each of the schemes in [13]. However, the signer usually possesses much more computation and storage capability than the user in most of the applications that are based on blind signatures, such as electronic cash and voting protocols. Hence, to guarantee the quality of these ever-growing popular communication services, it is more urgent to reduce the load for users than that for the signer.

Table 2. Performance comparisons for users.

	Our scheme	[13][1]	[13][2]	[13][3]
Modular exponentiation computations	0	40	240	10
Inverse computations	0	1	0	1
Hashing computations	2	60	2	2
Modular multiplication computations	18	60	160	6
Computations reduced by:		99%	99%	99%
Number of integers in a signature	2	40	2	6
Signature size reduced by:		95%	0%	67%

[1] The first scheme of [13]. [2] The second scheme of [13]. [3] The third scheme of [13].

5. CONCLUSIONS

In this paper we have proposed a user-efficient blind signature scheme. Our scheme not only possesses the fairness property, but also minimizes the computation required of users. The proposed scheme is suitable for situations where hardware and computation capability of users or customers is limited. With the help from a judge or a government, it is possible to recover the link between a signature and the instance of the signing protocol which produces that signature when the unlinkability property is abused.

Nevertheless, the proposed scheme requires an on-line judge which may cause a bottleneck, and the judge must maintain a database for storing related information. Therefore, in the future it also is necessary to design a user-efficient fair blind signature protocol with an efficient judge.

ACKNOWLEDGEMENTS

We would like to thank the anonymous referees of this paper for their valuable comments.

REFERENCES

1. S. Brands, "Untraceable off-line cash in wallets with observers," *Advances in Cryptology-CRYPTO* '93, LNCS 773, Springer-Verlag, 1993, pp. 302-318.
2. J. L. Camenisch, J. M. Piveteau, and M. A. Stadler, "An efficient payment system protecting privacy," in *Proceedings of European Symposium on Research in Computer Security* '94, LNCS 875, Springer-Verlag, 1994, pp. 207-215.

3. D. Chaum, B. den Boer, E. van Heyst, S. Mjolsnes, and A. Steenbeek, “Efficient off-line electronic checks,” *Advances in Cryptology-EUROCRYPT* ’89, LNCS 434, Springer-Verlag, 1989, pp. 294-301.
4. D. Chaum, A. Fiat, and M. Naor, “Untraceable electronic cash,” *Advances in Cryptology-CRYPTO* ’88, LNCS 403, Springer-Verlag, 1990, pp. 319-327.
5. C. I. Fan and C. L. Lei, “Secure rewarding schemes,” in *Proceedings of the Thirtieth Annual Hawaii International Conference on System Sciences*, Vol. 3, 1997, pp. 571-580.
6. N. Ferguson, “Single term off-line coins,” *Advances in Cryptology-EUROCRYPT* ’93, LNCS 765, Springer-Verlag, 1994, pp. 318-328.
7. T. Okamoto and K. Ohta, “Universal electronic cash,” *Advances in Cryptology-CRYPTO* ’91, LNCS 576, Springer-Verlag, 1992, pp. 324-337.
8. C. A. Boyd, “A new multiple key ciphers and an improved voting scheme,” *Advances in Cryptology-EUROCRYPT* ’89, LNCS 434, Springer-Verlag, 1990, pp. 617-625.
9. 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.
10. A. Fujioka, T. Okamoto, and K. Ohta, “A practical secret voting scheme for large scale elections,” *Advances in Cryptology-AUSCRYPT* ’92, LNCS 718, Springer-Verlag, 1992, pp. 244-251.
11. H. Nurmi, A. Salomaa, and L. Santean, “Secret ballot elections in computer networks,” *Computers & Security*, Vol. 10, 1991, pp. 553-560.
12. S. V. Solms and D. Naccache, “On blind signatures and perfect crime,” *Computer and Security*, Vol. 11, 1992, pp. 581-583.
13. M. Stadler, J. M. Piveteau, and J. Camenisch, “Fair blind signatures,” *Advances in Cryptology-EUROCRYPT* ’95, LNCS 921, Springer-Verlag, 1995, pp. 209-219.
14. J. L. Camenisch, J. M. Piveteau, and M. A. Stadler, “Blind signatures based on the discrete logarithm problem,” *Advances in Cryptology-EUROCRYPT* ’94, LNCS 950, Springer-Verlag, 1995, pp. 428-432.
15. D. Chaum, “Blind signatures systems,” *Advances in Cryptology-CRYPTO* ’83, Plenum, 1983, p. 153.
16. D. Pointcheval and J. Stern, “Provably secure blind signature schemes,” *Advances in Cryptology-ASIACRYPT* ’96, LNCS 1163, Springer-Verlag, 1996, pp. 252-265.
17. R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public key cryptosystems,” *Communications of the ACM*, Vol. 21, 1978, pp. 120-126.
18. T. Okamoto, “Provably secure and practical identification schemes and corresponding signature schemes,” *Advances in Cryptology-CRYPTO* ’92, LNCS 740, Springer-Verlag, 1992, pp. 31-53.
19. C. P. Schnorr, “Efficient identification and signatures for smart cards,” *Advances in Cryptology-CRYPTO* ’89, LNCS 435, Springer-Verlag, 1990, pp. 235-251.
20. L. C. Guillou and J. J. Quisquater, “A practical zero-knowledge protocol fitted to security microprocessor minimizing both transmission and memory,” *Advances in Cryptology-EUROCRYPT* ’88, LNCS 330, Springer-Verlag, 1988, pp. 123-128.
21. D. Pointcheval and J. Stern, “New blind signatures equivalent to factorization,” in

Proceedings of the 4th ACM Conference on Computer and Communication Security, 1997, pp. 92-99.

22. S. Even, O. Goldreich, and A. Lempel, "A randomized protocol for signing contracts," *Communications of the ACM*, Vol. 28, 1985, pp. 637-647.
23. A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identification and signature problems," *Advances in Cryptology-CRYPTO* '86, LNCS 263, Springer-Verlag, 1986, pp. 186-194.
24. W. J. LeVeque, *Fundamentals of Number Theory*, Addison-Wesley, Reading, Mass., 1977.
25. R. C. Peralta, "A simple and fast probabilistic algorithm for computing square roots modulo a prime number," *IEEE Transactions on Information Theory*, Vol. 32, 1986, pp. 846-847.
26. M. O. Rabin, "Digitalized signatures and public-key functions as intractable as factorization," Technical Report, MIT/LCS/TR212, MIT Lab., Computer Science, 1979.
27. A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
28. H. C. Williams, "A modification of the RSA public-key encryption procedure," *IEEE Transactions on Information Theory*, Vol. 26, 1980, pp. 726-729.
29. A. Evans, W. J. Kantrowitz, and E. Weiss, "A user authentication scheme not requiring secrecy in the computer," *Communications of the ACM*, Vol. 17, 1974, pp. 437-442.
30. G. P. Purdy, "A high security log-in procedure," *Communications of the ACM*," Vol. 17, 1974, pp. 442-445.
31. C. I. Fan and C. L. Lei, "User efficient blind signatures," *Electronics Letters*, Vol. 34, 1998, pp. 544-546.
32. A. Shamir and C. P. Schnorr, "Cryptanalysis of certain variants of Rabin's signature scheme," *Information Processing Letters*, Vol. 19, 1984, pp. 113-115.

APPENDIX

Proof of theorem 1. By the Chinese remainder theorem [27], every integer w in Z_n^* can be represented by <w_1, w_2> where w_1 = (w mod p_1) and w_2 = (w mod p_2). For convenience, sometimes <w_1, w_2> is denoted by <w>. For every <k> = <k_1, k_2> and <w> = <w_1, w_2> in Z_n^*, <kw mod n> = <k_1w_1 mod p_1, k_2w_2 mod p_2>, and <k^{-1} mod n> $= < k_1^{-1}$ mod p_1, k_2^{-1} mod p_2 >. 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 $k_1 \equiv w_1 \pmod{p_1}$ and $k_2 \equiv w_2 \pmod{p_2}$. In addition, if w and k are QR's in Z_n^*, then the integer (wk mod n) is also a QR in Z_n^* [27].

Since both ($\alpha(x^2 + 1)$ mod n) and (e^2 mod n) are QR's in Z_n^*, we have

$$\begin{aligned}
&\alpha(x^2 + 1)e^2\\
&\equiv \alpha(x^2 + 1)\lambda^{-2}\\
&\equiv H(m)(u^2 + v^2)(x^2 + 1)(b^2(u - vx))^{-2}\\
&\equiv b^{-4}H(m)(u^2 + v^2)(x^2 + 1)(u - vx)^{-2}\\
&\equiv b^{-4}H(m)((ux + v)^2 + (u - vx)^2)(u - vx)^{-2}
\end{aligned}$$

$$
\begin{aligned}
&\equiv b^{-4}H(m)((ux+v)^2(u-vx)^{-2}+1)\\
&\equiv b^{-4}H(m)(((ux+v)(u-vx)^{-1})^2+1)\\
&\equiv b^{-4}H(m)\,((b^2b^{-2}(u-vx)^{-1}(ux+v))^2+1)\\
&\equiv b^{-4}H(m)\,((b^2e(ux+v))^2+1)\\
&\equiv b^{-4}H(m)\,(c^2+1) \pmod{n}
\end{aligned}
$$

is a QR in Z_n^*. Since (b^4 mod n) is a QR in Z_n^*, the integer ($H(m)(c^2 + 1)$ mod n) is also a QR in Z_n^*. Let $<d_1, d_2>$ be one of the 4th roots of the integer ($H(m)(c^2 + 1)$ mod n) in Z_n^*. Then the four 4th roots of ($H(m)(c^2 + 1)$ mod n) in Z_n^* are $< \pm d_1, \pm d_2 >$. Thus, the four 4th roots of ($b^{-4}H(m)(c^2 + 1)$ mod n) in Z_n^* are $< \pm b_1^{-1}d_1, \pm b_2^{-1}d_2 >$. Since $t^4 \equiv b^{-4}H(m)(c^2 + 1) \pmod{n}$, t belongs to $\{< \pm b_1^{-1}d_1, \pm b_2^{-1}d_2 >\}$ and since $s = (bt \bmod n)$, s is an element in $\{< \pm b_1b_1^{-1}d_1, \pm b_2b_2^{-1}d_2 >\} = \{< \pm d_1, \pm d_2 >\}$. It follows that s is a 4th root of the integer ($H(m)(c^2 + 1)$ mod n) in Z_n^*. Hence, formula (5) holds. ❑

Proof of theorem 2. If $c \equiv (u_i'x_i + v_i')(u_i' - v_i'x_i)^{-1} \pmod{n}$ we have $u_i' \equiv v_i'(cx_i + 1)$ $(c - x_i)^{-1} \pmod{n}$. For every quadratic residue r in Z_n^*, we define ($r^{\frac{1}{2}}$ mod n) to be a square root of r in Z_n^*, where ($r^{\frac{1}{2}}$ mod n) has four possible values in Z_n^* because n is the product of two distinct primes [26, 27]. By theorem 1, $s^4 \equiv H(m)(c^2 + 1) \pmod{n}$. If $\alpha_i \equiv H(m)((u_i')^2 + (v_i')^2) \pmod{n}$, then we have the following derivation:

$$
\begin{aligned}
&\alpha_i \equiv H(m)((u_i')^2 + (v_i')^2) \pmod{n}\\
&\alpha_i \equiv H(m)((v_i')^2(cx_i+1)^2(c-x_i)^{-2} + (v_i')^2) \pmod{n}\\
&\alpha_i \equiv H(m)(v_i')^2((cx_i+1)^2(c-x_i)^{-2} + 1) \pmod{n}\\
&\alpha_i \equiv H(m)(v_i')^2((cx_i+1)^2 + (c-x_i)^2)(c-x_i)^{-2} \pmod{n}\\
&\alpha_i \equiv H(m)(v_i')^2(c^2+1)(x_i^2+1)(c-x_i)^{-2} \pmod{n}\\
&\alpha_i \equiv (v_i')^2 s^4 (x_i^2+1)(c-x_i)^{-2} \pmod{n}\\
&(v_i')^2 \equiv s^{-4}\alpha_i(x_i^2+1)^{-1}(c-x_i)^2 \pmod{n}\\
&(v_i')^2 \equiv s^{-4}\alpha_i(x_i^2+1)(x_i^2+1)^{-2}(c-x_i)^2 \pmod{n}\\
&v_i' \equiv s^{-2}(\alpha_i(x_i^2+1))^{\frac{1}{2}}(x_i^2+1)^{-1}(c-x_i) \pmod{n}.
\end{aligned}
$$

The integer $(\alpha_i(x_i^2+1) \bmod n)$ is a quadratic residues in Z_n^*, so that $((\alpha_i(x_i^2+1))^{\frac{1}{2}}$ mod n) exists in Z_n^*, and both u_i' and v_i' have four different values in Z_n^*. Thus, if $\lambda_i \equiv (b_i')^2$ $(u_i' - v_i'x_i) \pmod{n}$, we have

$$
\begin{aligned}
&\lambda_i \equiv (b_i')^2(v_i'(cx_i+1)(c-x_i)^{-1} - v_i'x_i) \pmod{n}\\
&\lambda_i \equiv (b_i')^2 s^{-2}(\alpha_i(x_i^2+1))^{\frac{1}{2}}(x_i^2+1)^{-1}(c-x_i)((cx_i+1)(c-x_i)^{-1} - x_i)\\
&\lambda_i \equiv (b_i')^2 s^{-2}(\alpha_i(x_i^2+1))^{\frac{1}{2}}(x_i^2+1)^{-1}((cx_i+1) - (c-x_i)x_i) \pmod{n}\\
&\lambda_i \equiv (b_i')^2 s^{-2}(\alpha_i(x_i^2+1))^{\frac{1}{2}}(x_i^2+1)^{-1}(x_i^2+1) \pmod{n}\\
&\lambda_i \equiv (b_i')^2 s^{-2}(\alpha_i(x_i^2+1))^{\frac{1}{2}} \pmod{n}\\
&(b_i')^2 \equiv \lambda_i s^2(\alpha_i(x_i^2+1))^{-\frac{1}{2}} \pmod{n}.
\end{aligned}
\tag{7}
$$

Since there must exist exactly one value among the four different values of $((\alpha_i(x_i^2 + 1))^{-\frac{1}{2}} \bmod n)$ such that $(\lambda_i s^2(\alpha_i(x_i^2+1))^{-\frac{1}{2}} \bmod n)$ is a quadratic residue in Z_n^* [27], we can also derive four different values of b_i' in Z_n^* from equation (7). ❑

Chun-I Fan (范俊逸) was born in Tainan, Taiwan on October 15, 1967. He received the M.S. degree in computer science and information engineering from National Chiao Tung University, Taiwan, in 1993, and the Ph.D. degree in electrical engineering from National Taiwan University, Taiwan, in 1998. He joined the Telecommunication Laboratories of Chunghwa Telecom Co., Ltd. in 1999. His current research interests include information security & privacy, cryptographic protocols, and electronic commerce.

Chin-Laung Lei (雷欽隆) was born in Taipei, Taiwan on January 9, 1958. He received the B.S. degree in electrical engineering from National Taiwan University in 1980, and the Ph.D. degree in computer science from the University of Texas at Austin in 1986. From 1986 to 1988, he was an assistant professor in the Computer and Information Science Department at the Ohio State University, Columbus, Ohio, U.S.A. In 1988, he joined the department of electrical engineering, National Taiwan University, where he is now a professor. His current research interests include network and computer security, parallel and distributed processing, operating system design, and formal semantics of concurrent programs. Dr. Lei is a member of the Institute of Electrical and Electronic Engineers and the Association for Computing Machinery.