Computers & Security (2005) **24**, 44–49

ELSEVIER

Computers & Security

www.elsevier.com/locate/cose

A randomized RSA-based partially blind signature scheme for electronic cash

Tianjie Cao[a,b,*], Dongdai Lin[a], Rui Xue[a]

[a]*State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences, Beijing 100080, PR China*
[b]*School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221008, PR China*

Received 24 February 2004; revised 17 May 2004; accepted 26 May 2004
Available online 28 January 2005

KEYWORDS
Partially blind signatures;
Electronic cash;
Privacy;
Randomization;
RSA

Abstract Blind signature schemes can yield a signature and message pair whose information does not leak to the signer. However, when blind signatures are used to design e-cash schemes, there are two problems. One is the unlimited growth of the bank's database which keeps all spent e-cashes for preventing double spending. Another problem is that the signer must assure himself that the message contains accurate information such as the face value of the e-cash without seeing it. Partially blind signatures can cope with these problems. In partially blind signatures, the signer can explicitly include some agreed common information such as the expiration date and the face value in the blind signature. Randomized signature schemes can withstand one-more-forgery under the chosen plaintext attack. Based on RSA cryptosystem Fan–Chen–Yeh proposed a randomized blind signature scheme and Chien–Jan–Tseng also proposed a randomized partially blind signature scheme. But, the attacker can remove the randomizing factor from the messages to be signed in these two schemes. The attacker can also change the common information of Chien–Jan–Tseng's partially blind signature. In this paper, we propose a secure randomized RSA-based partially blind signature scheme, and show that the proposed scheme satisfies the blindness and unforgeability properties. We also analyse the computation cost of the proposed scheme.

* Corresponding author. Tel.: +86 106 265 404 65; fax: +86 106 252 0469.
E-mail addresses: tjcao@cumt.edu.cn (T. Cao), ddlin@is.iscas.ac.cn (D. Lin), rxue@is.iscas.ac.cn (R. Xue).

0167-4048/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cose.2004.05.008

Introduction

The blind signature technique was first introduced by Chaum (1983) to protect an individual's privacy. A secure blind signature scheme should satisfy the blindness and unforgeability properties.

Blindness: it allows a user to acquire a signature on a message without revealing anything about the message to the signer. Blindness property ensures that no one can derive a link between a *view* and a valid blind signature except the signature requester. A *view* of the signer is defined to be the set of all messages that the signer has received and generated when issuing the signature. Owing to the blindness property, blind signatures have been widely used in untraceable electronic cash systems.

Unforgeability: it means that only the signer can generate the valid signatures.

In an electronic cash system, the bank (or the signer) issues electronic cash, i.e., e-cash, and a customer (or a user) can withdraw e-cash from his account and deposit e-cash into his account in the bank. When we use blind signature to design e-cash schemes, there are two shortcomings. (1) To prevent a customer from double-spending his e-cash, the bank has to keep a spent database which stores all spent e-cash to check whether a specified e-cash has been spent or not by searching this database. Certainly, the spent database kept by the bank may grow unlimitedly. (2) To believe the face value of e-cash in the withdraw phase, the signer must assure himself that the message contains accurate information without seeing it. The cut-and-choose algorithm is widely used to solve this problem. But this is very inefficient. To get a low enough probability of cheating, the cut-and-choose must consist of many terms, and the vast amount of data terms spoil its computation and communication efficiency.

Partially blind signatures introduced by Abe and Fujisaki (1996) can eliminate the above two shortcomings. *Partial blindness* property allows the bank to explicitly include some agreed common information such as the expiration date and the face value in the blind signature. In a secure partially blind signature scheme, a user cannot replace the common information with another one. Using the partially blind signatures the bank can prevent the bank's spent database from growing unlimitedly. By embedding an expiration date into each e-cash issued by the bank, all expired e-cash recorded in the bank's database can be removed. This removal of the expired e-cash limits the size of the bank's spent database. The user can also renew his e-cash when the old e-cash is close to the expiration date. After verifying that the old e-cash, making sure the old e-cash is not expired and not in the spent database, the bank issues a new e-cash to the user using the partially blind signature scheme and records the old e-cash in the spent database. In addition, the bank also cannot build a relationship between the old e-cash and the new e-cash. Using partially blind signatures the bank can believe the face value of e-cash to be signed. By embedding the face value in each e-cash, the bank can clearly know the value on the blindly issued e-cash.

The Chaum (1983) RSA-based blind signature scheme and the Abe and Fujisaki (1996) RSA-based partially blind signature scheme are vulnerable to the one-more-forgery under the chosen plaintext attack (Coron et al. 1999, Desmedt and Odlyzko, 1994). Using the homomorphic property, the attacker can forge a new signature. To be immune to the chosen plaintext attack, Ferguson (1994) suggested that the signer had better inject one or more randomizing factors into the blinded message such that the attackers cannot predict the exact content of the message the signer signs. This is referred to as the *randomization* property. In a secure randomized signature scheme, a user cannot remove the signer's randomizing factor. Based on RSA cryptosystem Fan et al. (2000) proposed a randomized blind signature scheme and Chien et al. (2001) also proposed a randomized partially blind signature scheme. But, the attacker can remove the randomizing factor from the messages to be signed (Kwon and Cho, 2003), thus, Fan—Chen—Yeh's blind signature scheme and Chien—Jan—Tseng's partially blind signature scheme are vulnerable to the chosen plaintext attack. In addition, the attacker can also change the common information of Chien—Jan—Tseng's partially blind signature (Kwon and Cho, 2003).

Our main goal is to design a secure randomized partially blind signature scheme based on RSA. In this paper, we first describe our randomized partially blind signature scheme, and then examine the correctness, blindness and unforgeability properties. Finally we analyse the computation costs of the proposed scheme. Our proposed scheme can be used to design e-cash systems.

The proposed randomized partially blind signature scheme based on RSA

We begin by describing the RSA function. Let $n = pq$ be the product of two large primes of the

same size. A typical size for n is 1024 bits. Each of the factors is 512 bits. Let e, d be two integers satisfying $ed \equiv 1 (\mathrm{mod}\ \phi(n))$ where $\phi(n) = (p-1)(q-1)$. We define the RSA function as $\mathrm{RSA}_{N,e}(x) = x^e (\mathrm{mod}\ n)$. If d is given, the RSA function can be easily inverted using the equality $x^{ed} \equiv x (\mathrm{mod}\ n)$. We refer to d as a trapdoor enabling one to invert the RSA function. We assume that the RSA function is one-way because the RSA function can be easily computed, but cannot be efficiently inverted without the trapdoor d.

Let a be a common information containing an expiration date and the face value of the e-cash. We define the formatting function of common information as

$$\tau(a) = 2^k + H(a)$$

where H is a public one-way hash function whose length is k bits. The formatting function $\tau(a)$ is designed to keep its domain in $2^k < \tau(a) < 2^{k+1}$ so that $\tau(a)$ does not divide $\tau(a')$ where $a \neq a'$.

Our randomized RSA-based partially blind signature scheme consists of five stages: *Initialization*, *Blinding*, *Signing*, *Unblinding*, and *Verifying*, described as follows.

1. *Initialization*. Initially, the singer randomly selects two distinct large primes p and q, and then computes $n = pq$ and $\phi(n) = (p-1)(q-1)$. The signer chooses two large integers e and d at random such that $ed \equiv 1 (\mathrm{mod}\ \phi(n))$. Then, the signer publishes (e, n) and a one-way hash function H. Here the public and private keys of the signer are (e, n) and (p, q, d), respectively. The integer e satisfies $2^{k+1} < e$.
2. *Blinding*. A user chooses a plaintext message m to be signed "blindly", and then prepares a string a negotiated and agreed by the user and the signer. The user submits a to the signer.

   After verifying that the string a is containing a proper expiration date and a face value, the signer randomly selects his randomizing factor $x \in Z_n^*$, where the integer x will be used to perturb the blinded message received from the user and Z_n^* is the set of all positive integers less than and relatively prime to n. The signer sends the integer $y = x^e (\mathrm{mod}\ n)$ to the user, where the integer y is the commitment to the signer's randomizing factor x and will be opened in the *Signing* stage.

   After receiving the commitment value y, the user randomly selects his randomizing factor u and the blinding factor r, where the user's randomizing factor u is used to perturb the message m, the user's blinding factor r is used to generate the blinded message and will be removed from the blinded signature in the *Unblinding* stage. The user computes the blinded message $\alpha = r^e u H(m \| u^e y) (\mathrm{mod}\ n)$, and then submits α to the signer where $\|$ is the string concatenation operator.
3. *Signing*. After receiving α, the signer injects his randomizing factor x into the blinded message α and computes the blinded signature $t = ((\alpha x)^{d\tau(a)})^{-1} (\mathrm{mod}\ n)$ using his private key d, where the message (αx) to be signed is determined by both the user and the signer. Then the signer sends the blinded signature t and his randomizing factor x to the user.
4. *Unblinding*. Using the signer's randomizing factor x, the user computes the randomizing factor $c = ux (\mathrm{mod}\ n)$, where u is the user's contribution and x is the signer's contribution. The randomizing factor c is determined by both the user and the signer. The user computes $s = r^{\tau(a)} t (\mathrm{mod}\ n)$ to remove the blinding factor r from the blinded signature t. The tuple (s, c, a) is the signer's signature on plaintext message m.
5. *Verifying*. To verify the signature (s, c, a) of m, one can examine if $s^e (H(m \| c^e) c)^{\tau(a)} \equiv 1 (\mathrm{mod}\ n)$.

In the following section, we will examine the correctness, blindness, unforgeability and performance of the proposed partially blind signature scheme.

Analysis

Correctness

In the *Blinding* stage of the proposed partially blind signature, the user computes the blinded message $\alpha = r^e u H(m \| u^e y) (\mathrm{mod}\ n)$ and sends α to the signer. If one of the integers u, r or $H(m \| u^e y)$ is not in Z_n^*, the signer cannot compute $((\alpha x)^{d\tau(a)})^{-1} (\mathrm{mod}\ n)$ in the *Signing* stage. However, the probability of that u, r or $H(m \| u^e y)$ is not in Z_n^* is negligible and nearly $2^{-|p|}$ or $2^{-|q|}$ where $|p|$, $|q|$ denote the bit lengths of p, q, and $|p|$, $|q| = 512$ in a practical implementation. This negligible situation also exists in the Chaum (1983) blind signature scheme, Ferguson (1994) randomized blind signature scheme and Abe and Fujisaki (1996) partially blind signature scheme. In the following discussion, we shall assume that u, r and $H(m \| u^e y)$ are all

in Z_n^*. Obviously, the integers α, t, c, s and $H(m\|c^e)$ are also in Z_n^* due to $u \in Z_n^*$, $r \in Z_n^*$, $H(m\|u^e y) \in Z_n^*$ and $x \in Z_n^*$.

Now we show that if 3-tuple (s,c,a) is a signature of a message m produced by the proposed partially blind signature scheme then $s^e(H(m\|c^e)c)^{\tau(a)} \equiv 1 \pmod n$. The validity can easily be established as follows.

$$
\begin{aligned}
&s^e(H(m\|c^e)c)^{\tau(a)} \\
&\equiv \left(r^{\tau(a)}t\right)^e (H(m\|c^e)c)^{\tau(a)} \\
&\equiv \left(r^{\tau(a)}((\alpha x)^{d\tau(a)})^{-1}\right)^e (H(m\|c^e)c)^{\tau(a)} \\
&\equiv \left(r^{e\tau(a)}(\alpha x)^{\tau(a)}\right)^{-1} (H(m\|c^e)c)^{\tau(a)} \\
&\equiv (r^{e\tau(a)}((r^e u H(m\|u^e y)x)^{\tau(a)})^{-1} (H(m\|c^e)c)^{\tau(a)} \\
&\equiv ((uH(m\|u^e x^e)x)^{\tau(a)})^{-1} (H(m\|c^e)c)^{\tau(a)} \\
&\equiv 1 \pmod n
\end{aligned}
$$

Blindness

For each instance numbered i of the proposed scheme, the signer can record α_i received from the user who communicated with the signer during the instance i of the scheme. The tuple (α_i, x_i, t_i) is referred to as the *view* of the signer to the instance i of the scheme. The blindness property means that the signer cannot derive a link between a *view* and a valid blind signature after the blind signature has been revealed to the public. Theorem 1 ensures the blindness property of the scheme.

Theorem 1. *Given a 4-tuple (s, m, c, a) produced by the proposed partially blind signature scheme, the signer can derive u and r for each view (α_i, x_i, t_i) such that*

$$c \equiv ux_i \pmod n,$$

$$\alpha_i \equiv r^e u H\left(m\|u^e x_i^e\right) \pmod n \text{ and}$$

$$s = r^{\tau(a)} t_i \pmod n$$

are satisfied where (α_i, x_i, t_i) is regarded as (α, x, t).

Proof. Since (s, m, c, a) is produced by the proposed partially blind signature scheme and (α_i, x_i, t_i) is the view of the signer, we have s, c, $H(m\|c^e)$, α_i, x_i, $t_i \in Z_n^*$. Since $x_i \in Z_n^*$, the signer can drive the unique integer $u = cx_i^{-1} \pmod n$ from equation $c \equiv ux_i \pmod n$. Since $c \in Z_n^*$, we have $u \in Z_n^*$. Since $c \in Z_n^*$ and $H(m\|c^e) \in Z_n^*$, from equation $\alpha_i \equiv r^e u H(m\|u^e x_i^e) \pmod n$, the signer can obtain the unique solution $r = (\alpha_i (c^{-1}) x_i H^{-1}(m\|c^e))^d \pmod n$.

We now show that $s \equiv r^{\tau(a)} t_i \pmod n$ is satisfied for determined integers $u = cx_i^{-1} \pmod n$ and $r = (\alpha_i (c^{-1}) x_i H^{-1}(m\|c^e))^d \pmod n$.

Since the 4-tuple (s, m, c, a) is produced by the proposed partially blind signature scheme, then we have equality $s^e(H(m\|c^e)c)^{\tau(a)} \equiv 1 \pmod n$. Now we may obtain the equivalence form $s \equiv (c^{-1}H^{-1}(m\|c^e))^{d\tau(a)} \pmod n$.

Since $r = (\alpha_i (c^{-1}) x_i H^{-1}(m\|c^e))^d \pmod n$ and $t_i \equiv ((\alpha_i x_i)^{d\tau(a)})^{-1} \pmod n$, we have

$$
\begin{aligned}
&r^{\tau(a)} t_i \\
&\equiv (\alpha_i (c^{-1}) x_i H^{-1}(m\|c^e))^{d\tau(a)} ((\alpha_i x_i)^{d\tau(a)})^{-1} \\
&\equiv (c^{-1}H^{-1}(m\|c^e))^{d\tau(a)} \\
&\equiv s \pmod n
\end{aligned}
$$

Therefore, given a 4-tuple (s, m, c, a) produced by the scheme, the signer can always derive u and r for each view (α_i, x_i, t_i) such that $c \equiv ux_i \pmod n$, $\alpha_i \equiv r^e u H(m\|u^e x_i^e) \pmod n$ and $s = r^{\tau(a)} t_i \pmod n$ are satisfied. It turns out that all the 4-tuples (s, m, c, a) are indistinguishable from the signer's point of view. This is the blindness property of blind signature scheme. □

Unforgeability

Four types of forgery against the partial blind signature are considered here.

(1) Without the help from the signer, a user generates a valid partial blind signature.
(2) Given a valid partial blind signature, a user extracts another valid signature.
(3) Given a larger number of valid partial blind signatures, a user extracts a new valid signature. This is a kind of chosen plaintext attack.
(4) A user replaces the common information a with a' where $a \neq a'$ when he requires a signature on a plaintext message.

First, type (1) is discussed.Theorem 2 ensures that the forgery of type (1) is computationally infeasible.

Theorem 2. *It is computationally infeasible for a user to obtain a 4-tuple (s, m, c, a) such that*

$$s^e(H(m\|c^e)c)^{\tau(a)} \equiv 1 \pmod n$$

is satisfied without the help from the signer.

Proof. The signature verification formula of $s^e(H(m\|c^e)c)^{\tau(a)} \equiv 1 (\mathrm{mod}\ n)$ is equivalent to

$$s^e(gc)^f \equiv 1 \left(\mathrm{mod}\ n\right),$$

$$g \equiv H(m\|c^e) (\mathrm{mod}\ n) \text{ and}$$

$$f = 2^k + H\left(a\right)$$

Given n, if one can derive a 6-tuple (s, m, c, a, g, f) such that $s^e(gc)^f \equiv 1 (\mathrm{mod}\ n)$, $g \equiv H(m\|c^e) (\mathrm{mod}\ n)$ and $f = 2^k + H(a)$ are satisfied without the help from the signer, then the scheme is insecure. In the following, we discuss the difficulty of the above derivation. First, to find (m, c, g) such that $g \equiv H(m\|c^e) (\mathrm{mod}\ n)$ is true, the attackers have to decide the values of m and c in advance because both of them are the parameters of H. If they do not do so, it is computationally infeasible to derive (m, c) because H is one-way. Similarly, the attackers must determine the value of a previously for solving $f = 2^k + H(a)$ except they can invert the one-way hash function H. Let the attackers decide the values of m, c and a in advance. Thus, the values of g and f are then decided. To solve s from $s^e(gc)^f \equiv 1 (\mathrm{mod}\ n)$, the attackers have to compute s from $s^e \equiv (g^{-1}c^{-1})^f (\mathrm{mod}\ n)$ where $f < e$, that is intractable without known private key d.

We now consider the forgery of type (2). Given a 4-tuple (s, m, c, a) produced by the proposed partially blind signature scheme, we have

$$s^e(H(m\|c^e)c)^{\tau(a)} \equiv 1 (\mathrm{mod}\ n).$$

For an integer π, the equation $s^e(H(m\|c^e)c)^{\tau(a)} \equiv 1 (\mathrm{mod}\ n)$ can be rewritten in form

$$(s^\pi)^e(H(m\|c^e)c)^{\pi\tau(a)} \equiv 1 \left(\mathrm{mod}\ n\right)$$

If the user can select integers a' and π such that $\tau(a') \equiv \pi\tau(a) (\mathrm{mod}\ \phi(n))$, the user can extract a new partial blind signature (s^π, m, c, a') from the signature (s, m, c, a).

Here we show the user selects a' and π such that $\tau(a') \equiv \pi\tau(a) (\mathrm{mod}\ \phi(n))$ is intractable. Because the user could not know $\phi(n)$ without the factorization of n, we consider the equation $\tau(a') = \pi\tau(a)$. According to the definition of the formatting function of common information, $\tau(a)$ is designed to keep its domain in $2^k < \tau(a) < 2^{k+1}$so that $\tau(a)$ does not divide $\tau(a')$, that prevents the forgery attack.

Then we discuss the type (3) threat. A secure randomized signature scheme can withstand the chosen text attacks. Using the randomized scheme, even if the user has a larger number of valid signatures, it is difficult for him to find new ones. In the *Blinding* stage of the proposed scheme, a user chooses an integer α and submits α to the signer. If the user tries to choose α such that $((\alpha x)^{d\tau(a)})^{-1} \equiv 1 (\mathrm{mod}\ n)$, then he has to compute α such that $\alpha \equiv (y^{-1})^d (\mathrm{mod}\ n)$ where $y = x^e$ $(\mathrm{mod}\ n)$. Since the integer x is unknown to the user in the *Blinding* stage and the integer d is the signer's private key, given $y = x^e (\mathrm{mod}\ n)$ and n, it is infeasible for the user to compute α such that $\alpha \equiv (y^{-1})^d (\mathrm{mod}\ n)$. Hence, in the proposed scheme, the user cannot remove the signer's randomizing factor x from the corresponding signature (s, c, a) of m.

Finally we discuss the type (4) cheating. The *partial blindness* property can guarantee that all signatures issued by the signer contain a valid information a agreed by the users and the signer, and the users cannot change the string a embedded in their signatures.

Observing the proposed scheme, in *Signing* stage, the signer derives an integer t such that

$$t^e \equiv ((\alpha x)^{\tau(a)})^{-1} (\mathrm{mod}\ n)$$

The above equation can be rewritten in form

$$(t^\pi)^e \equiv ((\alpha x)^{\pi\tau(a)})^{-1} \left(\mathrm{mod}\ n\right) \text{ for any integer } \pi$$

If the user tries to change the string a with $a' \neq a$, then he has to select a' and π such that $\tau(a') \equiv \pi\tau(a) (\mathrm{mod}\ \phi(n))$. In this case, if ($s$, c, a) is a signature of the message m produced by the proposed partially blind signature scheme, then (s^π, c, a') is a signature of the message m. According to the discussion of type (2), the user selects a' and π such that $\tau(a') \equiv \pi\tau(a) (\mathrm{mod}\ \phi(n))$ is intractable. □

Performance

The computation time for an inverse computation in Z_n^* is about $O(|n|^3)$ where $|n|$ denotes the bit length of n (Stinson, 2002). The modulus n is usually taken about 1024 bits in a practical implementation. A modular exponentiation computation in Z_n^* takes about the same time as that of an inverse computation in Z_n^*, and a hashing computation does not take longer time than that of a modular multiplication computation. Typically, under a modulus n, the computation time for a modular exponentiation is about $O(|n|)$ times that of a modular multiplication (Stinson, 2002).

Now we discuss how to compute the blinded signature t in the *Signing* stage. To compute the

blinded signature $t = ((\alpha x)^{d\tau(a)})^{-1} (\mathrm{mod}\, n)$, we have

$$\begin{aligned}&((\alpha x)^{d\tau(a)})^{-1}\\ &\equiv (\alpha x)^{-d\tau(a)}\\ &\equiv (\alpha x)^{(\phi(n)-d)\tau(a)} (\mathrm{mod}\, n)\end{aligned}$$

The signer can pre-compute the integer $d' = (\phi(n) - d)$ using his two private integers d and $\phi(n)$ in the *Initialization* stage. When the signer computes $t = ((\alpha x)^{d\tau(a)})^{-1} (\mathrm{mod}\, n)$, the signer can calculate $t = (\alpha x)^{(\phi(n)-d)\tau(a)} (\mathrm{mod}\, n) = (\alpha x)^{d'\tau(a)} (\mathrm{mod}\, n) = (\alpha x)^{(d'\tau(a) \bmod \phi(n))} (\mathrm{mod}\, n)$.

Here, we summarize the computation costs of the proposed scheme. In the proposed scheme, no inverse computations in Z_n^* are performed in a signing procedure. There are six modular exponentiations, six modular multiplications, three hashing operations and twice of random number generation performed by the user to obtain and verify a signature. There are two modular exponentiations, two modular multiplications, one hashing operation and once random number generation performed by the signer to issue a signature.

Conclusion

In this paper, we have presented a secure randomized partially blind signature scheme based on RSA. The blindness and unforgeability properties and the computation costs were also examined. We believe that more efficient randomized RSA-based scheme for electronic cash will be proposed in the future.

Acknowledgments

We thank the support of National Natural Science Foundation of China (NSFC90204016, NSFC60373048) and the National High Technology Development Program of China under Grant (863, No. 2003AA144030). Finally, we thank the anonymous referees for their helpful comments and suggestions.

References

Abe M, Fujisaki E. How to date blind signatures, Advances in cryptology—Asiacrypt'96, LNCS,1163. Springer-Verlag; 1996. p. 244–251.

Chaum D. Blind signatures for untraceable payments, Advances in cryptology—CRYPTO' 82, Plenum; 1983. p. 199–203.

Chien HY, Jan JK, Tseng YM. RSA-based partially blind signature with low computation. In: Proceedings of the eighth international conference on parallel and distributed systems; 2001. p. 385–389.

Coron JS, Naccache D, Stern JP. On the security of RSA padding, Advances in cryptology—CRYPTO'99, LNCS,1666. Springer-Verlag; 1999. p. 1–18.

Desmedt Y, Odlyzko A. A chosen text attack on the RSA cryptosystem and some discrete logarithm schemes, Advances in cryptology—CRYPTO'85, LNCS,218. Springer-Verlag; 1994. p. 318–328.

Fan CI, Chen WK, Yeh YS. Randomization enhanced Chaum's blind signature scheme. Comput Commun, vol. 23, 2000 p. 1677–1680.

Ferguson N. Single term off-line coins, Advances in Cryptology—EUROCRYPT'93, LNCS,765. Springer; 1994. p. 318–328.

Kwon MS, Cho YK. Randomization enhanced blind signature schemes based on RSA. IEICE A Fundam 2003;E86-A(3): 730–3.

Stinson D. Cryptography theory and practice. 2nd ed. CRC Press; 2002.

Tianjie Cao received the M.Sc. degree in Mathematics from Nankai University (P.R. China) in 1993. Currently he is an Associate Professor of Computer Science at China University of Mining and Technology. He is also Ph.D. candidate at the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences. His research interests include electronic cash, cryptographic protocols and network security.

Dongdai Lin received the M.Sc. and Ph.D. degree in Cryptography at Institute of System Sciences, Chinese Academy of Sciences in 1990. Currently he is a Research Professor at the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences.

Rui Xue received the Ph.D. degree in Mathematics from Beijing Normal University (P.R. China) in 1999. He was a post-doctorial fellow at the Laboratory of Computer Science in the Institute of Software (1999–2001), Chinese Academy of Sciences (CAS). Currently he is a Research Professor at the State Key Laboratory of Information Security, Institute of Software, CAS.