A More Efficient Untraceable E-Cash System with Partially Blind Signatures Based on the Discrete Logarithm Problem

Shingo MIYAZAKI and Kouichi SAKURAI*

Dept. of Computer Science, Kyushu Univ.
Hakozaki, Higashi-ku, Fukuoka, 812-8581, JAPAN.
{shingo,sakurai}@csce.kyushu-u.ac.jp

Abstract. We propose a new untraceable electronic money system based on the discrete logarithm problem. Our system improves the efficiency of Yacobi's E-money system by making the applied blind signature *partial*. We compare our system to the previous e-money systems which use the ElGamal-type scheme in their tracing a double-spender.
We also remark a double-registration problem on a digital cash system, recently presented in [Nguyen-Mu-Varadharajan, in Information Security Workshop'97], based on the blind Nyberg-Rueppel signature.

1 Introduction

Electronic cash systems, which use the conventional blind signature schemes [Cha83, CPS94], require many public keys corresponding to the face-values of coins [Bra94, Bra95a, CFN88, Cha83, Scho95, Yac94]. Partially blind signatures introduced in [AF96] makes such systems efficient by decreasing the number of public keys.

The original paper [AF96] preented a RSA-based partially blind signature, and suggested how to apply it to an electronic cash system. Then, discrete-log. based partially blind signature protocols were developed in [AC97, MAS97]. However, no exact application of these partial blind protocols to electronic cash system is investigated. So, this paper further explores the power of partially blind signatures in electronic money systems.

We first discuss an applicability of the partially blind to the Yacobi's untraceable E-money system [Yac94], which uses the ElGamal signature scheme, for making efficient. However, a direct adaptation of the partially blind signatures is shown to require additional cryptographic assumptions on zero-knowledge protocols, which play the central role in the Yacobi's original system.

Then, instead of using zero-knowledge protocols, we utilize the idea of the secret key certificate by Brands [Bra95a] for revising the Yacobi's system [Yac94], then apply a partially blind signature. Thus, our proposed e-money system checks the key certificates at the stage of payment, whereas the Yacobi's system [Yac94] does this in withdrawing.

* A part of this work is done while visiting in Columbia University, Computer Science Dept.

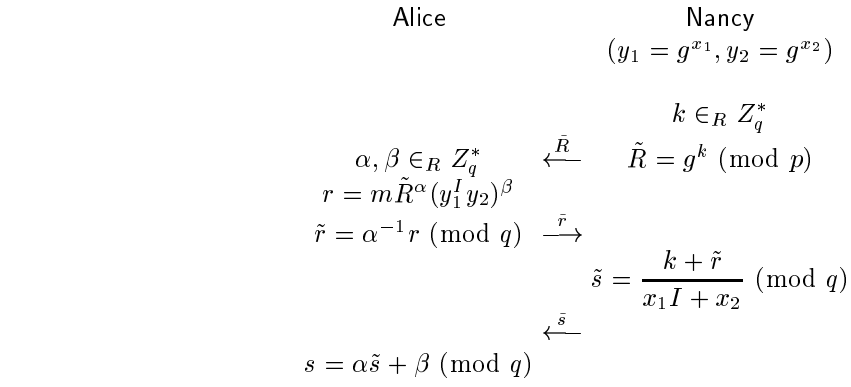

Fig. 1. The partially blind signature scheme [MAS97]

We compare our proposed system to the previous e-money systems [Yac94, Bra94, Scho95, NMV97] that use the discrete-logarithm based scheme in their tracing a double-spender.

Another contribution of this paper is to remark a problem on digital cash system based on the blind Nyberg-Rueppel signature [NMV97]. Nguyen et al. [NMV97] considered the security of their proposed scheme (anonymity, untraceability, double spending detection), However, we show that the double registration by a user's (or by multiple users' conspiring) is possible in this scheme [NMV97] (Appendix).

2 The Partially Blind Signature Scheme

The partially blind signature [AF96] is a signature on message (m, I), where m is blinded part for the signer in the protocols and I is the common information between the message sender and the signer. Unlike the conventional blind signature scheme, in partial blind signature protocols, the signer signs the blinded part m under the common information I.

We review the partially blind signature protocols for Nyberg-Rueppel signature scheme proposed in [MAS97]. The technique used in this scheme is similar to a reduced restrictive blind signature scheme as used in [Bra95a, Scho95]. Using this technique based on the discrete logarithm problem, Abe and Camenisch [AC97] presented the partially blind signature scheme based on Schnorr signature. Miyazaki, Abe and Sakurai [MAS97] proposed the partially blind signature schemes based on DSS, in addition to the scheme for Message Recovery Signature described here.

A signature on a message (m, I) with respect to the public key (y_1, y_2), where $y_1 = g^{x_1} \pmod p$ and $y_2 = g^{x_2} \pmod p$, is a pair (r, s) satisfying

$$m = r g^r (y_1^I y_2)^{-s} \pmod p.$$

Here, the first part m can be any string while the second part I must be $I \in Z_q^*$. Such a signature can be generated by the signer, who knows both secret key x_1 and x_2 corresponding to the public key y_1 and y_2 respectively, and computed by choosing a random integer $k \in Z_q^*$ and computing

$$r = mg^k \pmod{p} \quad \text{and} \quad s = \frac{k + r}{x_1 I + x_2} \pmod{q}.$$

It can easily be verified that the resulting pair (r, s) is actually a valid signature on (m, I). The below reviews the protocols of partially blind signature scheme proposed in [MAS97].

Step 1 The signer *Nancy* selects randomly an integer $k \in Z_q^*$, and computes $\tilde{R} = g^k \pmod{p}$. Then, *Nancy* sends $\tilde{R}$ to *Alice*.

Step 2 *Alice* generates respectively random integers $\alpha, \beta \in Z_q^*$, and computes $r = m\tilde{R}^{\alpha}(y_1^I y_2)^{\beta} \pmod{p}$. Then, *Alice* computes $\tilde{r} = \alpha^{-1} r \pmod{q}$, and sends $\tilde{r}$ to *Nancy*.

Step 3 *Nancy* computes $\tilde{s} = (k + \tilde{r})/(x_1 I + x_2)^{-1} \pmod{q}$ with her secret key (x_1, x_2), sends $\tilde{s}$ to *Alice*.

Step 4 *Alice* verifies that $1 \leq r \leq (p - 1)$; if not, then rejects the signature. *Alice* computes $s = \alpha\tilde{s} + \beta \pmod{q}$ to obtain *Nancy's* signature (r, s). *Alice* verifies the validity of her signature by computing the verification formula satisfying $rg^r(y_1^I y_2)^{-s} = m$.

The equations below shows that the protocol is sound;

$$rg^r(y_1^I y_2)^{-s} = mg^{r-\alpha\tilde{r}} \pmod{p} = m \pmod{p}.$$

Throughout this paper, let $Sig_A^{(I)}[m]$ be the A's signature on the message (m, I) obtained as the result of the partially blind protocol, where m is the message that should be blinded and I is the common information between the message sender and the signer.

3 The Yacobi's e-money and discussed problems

3.1 Reviewing Yacobi's System

The Yacobi's system [Yac94] uses a protocol for tracing double spenders like the technique used in [Bra94]; if two different messages are signed by the same user using the ElGamal signature scheme, with high probability the secret key of the signer can be efficiently computed. So, the double spender is traced, because the secret key includes the user's ID, We review basic protocols in the system as follows;

At first, User i generates randomly $R_i \in \{0,1\}^l$ (in practice $l = 200$ is currently sufficient), and creates the secret key $S_i = (I_i, R_i)$, where I_i is the user's name(ID). Then, she computes her public key $P_i \equiv g^{S_i} \pmod{p}$. Here, (e_C, N_C) is the CA's RSA public key and (e_B, N_B) is the Bank's one. Let γ be integer, $30 < \gamma < 50$, and 0^{γ} denotes a run of γ 0's.

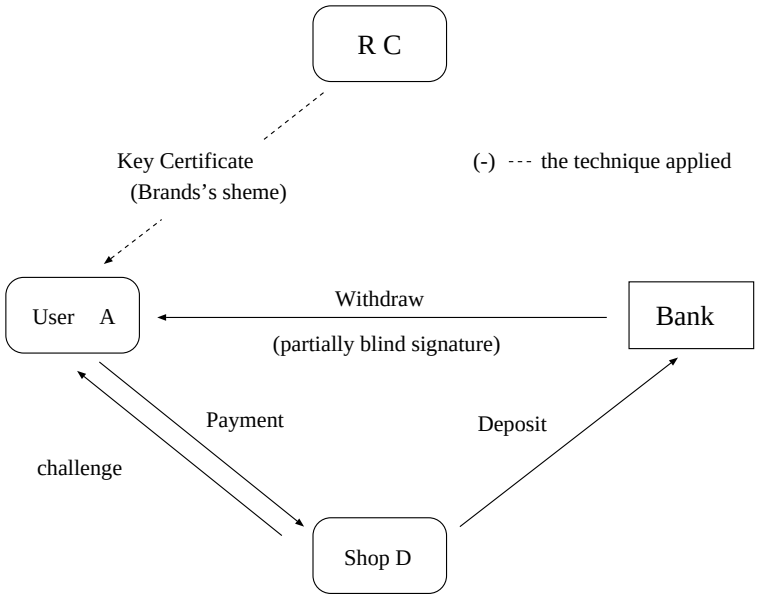

Fig. 2. The structure of our system and the technique applied

Initial-certificate: This operation takes place between a user and CA. User i selects randomly a integer $x \in Z^*_{N_C}$, and computes $z \equiv x^{e_C} f(P_i, 0^\gamma) \pmod{N_C}$ and sends z to CA with own ID, I_i. Next, User i proves to CA in Zero-Knowledge that S_i includes I_i. Precisely, the following predicate is proven to CA in zero-knowledge.

Predicate :

Given: z, g, p, N_C, e_C, I_i,

$(\exists x, P_i, R_i, S_i)[\ z \equiv_{N_C} x^{e_C} f(P_i, 0^\gamma);\ P_i \equiv_{N_C} g^{S_i};\ S_i = (I_i, R_i)]$

If CA verifies the proof of this predicate positively, CA computes $z^{d_C} \pmod{N_C}$ and sends it to User i. User i multiplies it by $x^{-1} \pmod{p}$ to obtain the unblinded certificate $Cert(i) \equiv (f(P_i, 0^\gamma))^{d_C} \pmod{N_C}$.

Withdrawal: User i computes $u \equiv g^r \pmod{p}$, where $gcd(r, p-1) = 1$. User i sends a candidate blinded coin $w \equiv x^{e_B} f(P_i, u, 0^\gamma) \pmod{N_B}$ to the Bank. Then, User i proves to the Bank in Zero-Knowledge that P_i is properly constructed. More precisely, the following predicate is proven to Bank.

Predicate :

Given: w, g, p, N_B, e_B, I_i,

$(\exists x, P_i, R_i, S_i, u)[\ w \equiv_{N_B} x^{e_B} f(P_i, u, 0^\gamma);\ P_i \equiv_{N_B} g^{S_i};\ S_i = (I_i, R_i)]$

After the Bank verifies the zero-knowledge proof on this predicate, the Bank computes $w^{d_B} \equiv x(f(P_i, u, 0^\gamma))^{d_B} \pmod{N_B}$ with the secret exponent *cor-*

responding to the required face-value, and sends w to User i. User i unblinds it to obtain the coin $c \equiv (f(P_i), u, 0^\gamma)^{d_B} \pmod{N_B}$.

Payment: User i sends a coin (P_i, u, c) to the payee. If the Bank's signature is correct, the payee generates the random challenge m and sends it to User i. User i signs on m with own secret key corresponding to the public key in the coin. The payee verifies the signature using the (P_i, u) embedded in the coin, rejects the coin if the signature is invalid.

Deposit: The payee sends the E-money (m, c, s) to the Bank, where $s = (u, v)$ is the payer User i's ElGamal signature on the challenge m. The Bank compares the (m, c, s) to the list of already deposited coins on Bank's database. If there is not a collision, the Bank increases the amount of the payee's account by the amount of the coin's face-value after verifying the validity of the deposited coin.

Now, in the Deposit protocol, the double-spending is detected if a collision is found. In this case, the double-spender could be traced as follows;

Tracing a double-spender: Now, User i double-spends the coin (P_i, u) for two distinct payments with the same secret key S_i. Then, the Bank obtains two different i's signatures (u, v_1) and (u, v_2) for the same coin, where $S_i u + r v_1 \equiv m_1 \pmod{(p-1)}$ and $S_i u + r v_2 \equiv m_2 \pmod{(p-1)}$. Hence, the relation $r(v_1 - v_2) \equiv (m_1 - m_2) \pmod{(p-1)}$ holds. So, if $gcd(v_1 - v_2, p-1) = 1$, then the Bank knowing the challenges m_1, m_2 can find r. Or, if $gcd(v, p-1) = 1$ then S_i can be computed. Finally, the Bank traces the double spender i by the I_i embedded in the S_i.

3.2 The drawbacks to be improved

The Yacobi's system has two problems: one is the complexity of the withdrawing protocol based on zero knowledge proofs and the other is the potential drawback of the e-money systems by using the conventional blind signature in the withdrawing protocol. The latter problem is to be necessary to prepare so many public keys corresponding to the face-values of coins in such systems.

Here, we discuss the direct adaptation of the partially blind signatures to the Yacobi's system. If this strategy is successful, then the resulted system would be more efficient by reducing the number of the pairs of the public and secret keys.

Now, User i performs the process of the initial-certificate same as the original Yacobi's system does. More precisely, the User must prove the existence of the parameters (a, P_i, u, S_i, R_i) satisfying below, giving (g, p, N_b, e_b, I_i, z) to the Bank via zero knowledge.

$$\begin{cases} z = a^{e_b} f(P_i, u, 0^\gamma) \pmod{N_b} \\ P_i = g^{S_i} \pmod{p} \\ S_i = (I_i, R_i) \end{cases}$$

Subsequently, the E-money is withdrawn through the protocol with the partially blind signature scheme on the agreed information I. Here, we use the scheme for

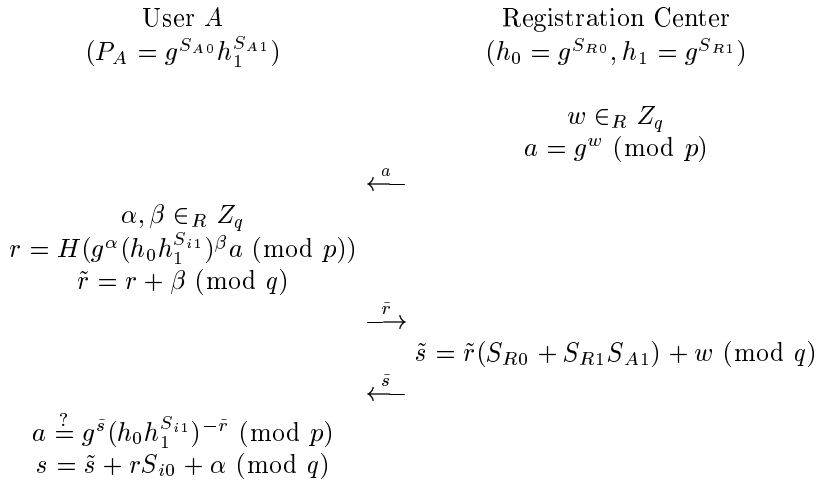

Fig. 3. Key Certificate issuing protocol

the Nyberg-Rueppel signature described in Section 2. In this case, User i must prove the Bank the following predicate via zero knowledge, where α, β, I, y_1, y_2 and $\tilde{R}$ correspond to those in Figure 1, respectively.

Predicate :

Given: $I_i, \tilde{R}, \tilde{r}, p, q, y_1, y_2, I$,
$(\exists r, \alpha, \beta, P_i, R_i, S_i, u)$

$$[\ r \equiv_p f(P_i, u, 0^\gamma) \tilde{R}^\alpha (y_1^I y_2)^\beta;$$
$$\tilde{r} \equiv_q \alpha^{-1} r; P_i \equiv_p g^{S_i}; S_i = (I_i, R_i)]$$

Here, this predicate is based on the (partially) blind signature for the Nyberg-Rueppel signature. It differs from the Yacobi's predicate based on the blind signature for RSA (the former is actually more complicated than the latter). So, the former requires additional assumption that the users could prove the predicate based on the discrete logarithm problem in zero-knowledge. A practical problem is that we do not have any practical solution to such a zero-knowledge proof than the theoretical method [GMW86] of reducing the predicate to NP-complete languages.

4 Our Proposed System

We describe our proposed system.

4.1 Registration

Brands presented the scheme issuing the secret key certificate [Bra95b] and proposed the E-money system [Bra95a] using this technique In our system, User obtains the certificate of the own key through the Brands's scheme. The system parameter consist of a large prime p, a prime factor q, and a generator g in Z_p^* of order q. Here, let (S_{A0}, S_{A1}) be the User A's secret key and (S_{R0}, S_{R1}) be the Registration Center(RC)'s secret key. Also, the part S_{A1} of User A's secret key (S_{A0}, S_{A1}) denotes the ID of User A and the common information between User A and Registration Center. Blinding own public key $P_A = g^{S_{A0}} h_1^{S_{A1}} \pmod p$ and the certificate (r, s), User A obtains the key certificate issued from Registration Center. Now, the public keys of Registration Center are $h_0 = g^{S_{R0}}$ and $h_1 = g^{S_{R1}} \pmod p$.

Our certificate issuing protocol between User A and Registration Center is the followings.

Step.1 Registration Center selects a random integer $w \in Z_q$ and computes $a = g^w \pmod p$. Registration Center sends a to User A.

Step.2 User A chooses random integers $\alpha, \beta \in Z_q$ and computes $r = H(g^\alpha (h_0 h_1^{S_{A1}})^\beta a \pmod p)$ and $\tilde{r} = r + \beta \pmod q$. User A sends $\tilde{r}$ to Registration Center.

Step.3 Registration Center computes $\tilde{s} = \tilde{r}(S_{R0} + S_{R1} S_{A1}) + w \pmod q$ with own secret key (S_{R0}, S_{R1}) and sends $\tilde{s}$ to User A.

Step.4 After User A verifies the signature satisfying $a = g^{\tilde{s}} (h_0 h_1^{S_{A1}})^{-\tilde{r}} \pmod p)$, computes $s = \tilde{s} + r S_{A0} + \alpha$.

The verification formula for the certificate of User A's key is $H(g^s (h_0 P_A)^{-r}) = r$. We prove the soundness of the verification formula below.

$$\begin{aligned} g^s (h_0 P_A)^{-r} &= g^{\beta(S_{R0} + S_{R1} S_{A1}) + w + \alpha} \pmod p \\ &= g^\alpha (h_0 h_1^{S_{A1}})^\beta a \pmod p \end{aligned}$$

Remark: Bank itself may play the role of Registration Center.

4.2 Withdrawal

User A withdraws the E-money from his account as follows. Bank reserves own secret key (x_1, x_2) and the corresponding to public keys; $y_1 = g^{x_1}$ and $y_2 = g^{x_2} \pmod p$.

Step 1 User A generates random integers $k_0, k_1 \in Z_q^*$ and computes $t = g^{k_0} h_1^{k_1} \pmod p$. Then, User A requests the Bank's signature on the message (m, I) through the partially blind signature protocols, where $m = (P_A \| t)$ is the blind part for Bank and I is the common information between User A and Bank including the amount of the withdrawing money and date.

Step 2 Bank, after deducting the amount of the money withdrawn from User A's account, sends the signature on (m, I) through the partially blind signature protocols.

Step 3 *User A* verifies the Bank's signature $Sig_B^{(I)}[m]$ on the money.

Remark: In step 1, User A should check if there is no candidate in the agreed information I, which destroys the anonymity of E-money.

4.3 Payment

User A makes a payment to *Shop D* as follows.

Step 1 User A sends $(Sig_B^{(I)}[m], m)$ and the certificate (r, s) of own key to *Shop D*.

Step 2 *Shop D* verifies the Bank's signature on the money and the certificate (r, s) of User A's key. If the validness of them are accepted, *Shop D* generates the challenge M and sends it to User A.

Step 3 User A signs on the challenge M with own secret key (S_{A0}, S_{A1}) and sends the signature (t, u, v) satisfying the following equations;

$$u = h(M)k_0 + S_{A0}t \pmod{q}$$
$$v = h(M)k_1 + S_{A1}t \pmod{q}.$$

Step 4 *Shop D* verifies User A's signature on challenge M with the following formula;

$$g^u h_1^v = t^{h(M)} P_A^t \pmod{p}.$$

4.4 Deposit

In Deposit protocol, *Shop D* sends the E-money $((Sig_B^{(I)}[m], m), (r, s), (t, u, v, M))$ to Bank. At the begin, Bank verifies the signature $(Sig_B^{(I)}[m], m)$ on the money, then, compares $(Sig_B^{(I)}[m], m)$ to the list of already deposited money on the database. If the deposited money $(Sig_B^{(I)}[m], m)$ is the first visit to Bank's database, Bank adds $(Sig_B^{(I)}[m], m)$ to the list linking the money to *Shop D* after increasing the amount of *Shop D*'s account by the amount indicated in I.

4.5 Tracing a Double-spender

In Deposit, if Bank discovers the corresponding money with deposited *Coin* on the database, Bank performs the tracing the double-spender as follows, original idea of which is proposed by Franklin and Yung [FY93]. Like Yacobi's way, from the part of the double-spent money

$$v_1 = h(M_1)k_1 + S_{A1}t \pmod{q}$$
$$v_2 = h(M_2)k_1 + S_{A1}t \pmod{q},$$

Bank computes $v_1 - v_2 = (h(M_1) - h(M_2))k_1 \pmod{q}$ to obtain k_1. Finally, Bank computes the secret key S_{A1} corresponding to the User's ID, and detects the double-spender.

References	Public key	Secret key	Key Certificates	Tools
Yacobi [Yac94]	Open	User (only)	In Withdraw	ZK protocol & RSA
Brands [Bra95a]	Blind	User (Bank partially)	Nowhere	Tamper-resistant device Schnorr signature
Schoenmakers [Scho95]	Blind	Bank/User	Nowhere	Schnorr signature
Nguyen et al. [NMV97]	Blind	User(only)	Nowhere	Nyberg-Rueppel signature
Our scheme	Open	User (Bank partially)	In Payment	Schnorr signature Partially blind signature

Fig. 4. Comparison to other systems

5 Comparison to the previous e-money systems

We depicts the comparison to related works [Yac94, Bra95a, Scho95, NMV97] in Figure 5, in which Bank uses the similar technique for tracing a double-spender.

5.1 Anonymity from public-keys

In Yacobi's [Yac94] and our system, the payer must show his own public key to payee in the payment protocol. Namely, in our system (also in Yacobi's), all e-money spent with the same user's public key P_A is linkable.

On the other hand, in [Bra95a, Scho95], the payer's public key is blinded by a random number at the payment. Then, the deposited money does not include any payer's footstep.

Thus, the systems presented in [Bra95a, Scho95, NMV97] achieves higher *anonymity* than ours (and Yacobi's).

5.2 Security vs. Efficiency

In [Yac94, NMV97] the user preserves only his own secret key including his ID. On the other hand, in [Scho95] Bank must conserve the user's secret key in addition to own secret key issuing much money. In terms of the dispersion of secret data, the former schemes are superior to the latter for their *security against the attack to user's secret keys*. However, the former have the lack of efficiency, because the user must prove the validity of the secret key, in which ID is embedded, by using a zero knowledge protocol.

In [Bra95a] and ours, Registration Center (maybe Bank) preserves the user's ID, which is only a part of the user's secret key, and embeds the ID in the issuing money or the user's key certificate. Hence, in such systems, the user can prove the validity of his own key without zero knowledge protocols (*efficiency*).

Even if a user's partial secret key stored in Bank is stolen, the thief cannot withdraw e-money from the user's account, because the thief cannot know the user's other secret key. For achieving this property, Brands [Bra95a] assumes

the use of a tamper-resister device, while in our system the verification of the key certificate in the payment protocol performs the function. Note that, if the Brands's system [Bra95a] is implemented without the tamper-resister device, the secret data would be centralized to Bank as Schoenmaker's system [Scho95].

Therefore, such systems achieve higher *security against the compromise of the user's secret key* than one proposed in [Scho95] because of the distributions of secret data. We should remark that, in Brands's scheme [Bra95a], the withdrawn E-money is the certificate not of the user's fixed public key but of the blinded temporary public key. device.

5.3 Attacks

The scheme proposed by Nguyen at al. [NMV97] appears to withstand parallel attacks remarked by Schoenmakers [Scho95]. However, another attack proposed by Chan at al. [CFMT96] is applicable to the system, if the registration has no process of the verification that the public key of the user should be constructed in the correct manner (See Appendix).

Furthermore, at the first step of the withdrawal protocol, Bank constructs the commitment using the user's public key as a generator. This makes the application of the partially blind signature scheme to the system [NMV97] difficult.

5.4 Number of required keys for coins

Due to the partially blind signature scheme, only our system has no need for the keys corresponding to the face-values of coins (*efficiency*) and allows users to make flexible payments by embedding various information (e.g. amount of money, date, valid period and so on) in the agreed information I. Actually, how many keys are cut down in our improved system by applying the partially blind signature scheme? The number of keys required in the system with the conventional blind signature scheme depend on the number of kinds of the agreed information. Suppose that the issuing *Time* and *Amount* of money are set as the agreed information, where *Time* consists of 12 months $\{January, February, ..., December\}$ and *Amount* has 5 kinds of the amount $\{\$1, \$5, \$10\,\$20, \$50\}$. In this case, the system with conventional blind signature scheme needs $12 \times 5 = 60$ key-pairs to represent the validity of the money such as the \$20 issued in a January. So, the more the number of kinds of the agreed information are set, the more the number of keys required are. On the other hand, our system with the partially blind signature scheme can deal with such a case by using only two fixed key-pairs. The more various information are set, the more efficient our system are getting compared with the previous systems.

6 Conclusion

We proposed a discrete-log based untraceable electronic money system with a partially blind signature protocol, discussed the advantage of our system over

the previous systems.

However, it is hard for Yacobi's system [Yac94] and ours to achieve e-money's transferability, because there is a troublesome case that Bank cannot detect a double-spender. So, making our e-money transferable is a challenging open problem.

Acknowledgments

The authors would like to thank to U. Mauer and M. Abe for giving the first author a chance of presenting the primitive version of the proposed scheme in ETH-Zurich. The authors are also grateful to J. Camenish for remarking the connection of the authors' primitive scheme to Brands' work on secret-key certificates. The second author wish to thank Zvi Galil and Moti Yung for their hospitality while his visiting Columbia Univ. Computer Science Dept.

References

[AC97] M. Abe, J. Camenisch, *"Partially Blind Signature Schemes,"* Proceedings of the 1997 Symposium on Cryptography and Information Security, SCIS97-33D, 1997.

[AF96] M. Abe, E. Fujisaki, *"How to Date Blind Signatures,"* Advances in Cryptology – ASACRYPT '96, LNCS 1163, pp. 244-251, 1996.

[Bra94] S. Brands, *"Untraceable off-line cash in wallet with observers,"* In Advances in Cryptology – CRYPTO '93, LNCS 773, pp. 302-318, 1994.

[Bra95a] S. Brands, *"Off-Line Electronic Cash Based on Secret-KeyCertificates,"* Proceedings of the Second International Symposium of Latin American Theoretical Informatics, 1995.
`http://www.cwi.nl/cwi/publications/CS-R9506.ps.Z`

[Bra95b] S. Brands, *"Restrictive Binding of Secret-Key Certificates,"* Advances in Cryptology – EUROCRYPT '95, LNCS 921, pp.231-247, 1995.

[CFMT96] A. Chan, Y. Frankel, P. MacKenzie, Y. Tsiounis, *"Mis-representation of Identities in E-cash Schemes and how to Prevent it,"* Advances in Cryptology – ASIACRYPT '96, LNCS 1163, pp. 276-285, 1996.

[CFN88] D. Chaum, A. Fiat, M. Naor, *"Untraceable Electronic Cash,"* In Advances in Cryptology – CRYPTO '88, pp. 319-327, 1988.

[Cha83] D. Chaum, *"Blind Signature for Untraceable Payments,"* In Advances in Cryptology – CRYPTO '82, pp. 199-203, 1983.

[CPS94] J. Camenisch, J. M. Piveteau, M. StadlerœB!œ(B *"Blind Signatures Based on the Discrete Logarithm Problem,"* Advances in Cryptology – EUROCRYPT '94, LNCS 950, pp. 428-432, 1994.

[ElG84] T. ElGamal, *"A public key cryptosystem and a signature scheme based on discrete logarithms,"* IEEE Transactions on Information Theory, pp. 469-472, 1985.

[FY93] M. K. Franklin, M. Yung *"Secure and Efficient Off-line Digital Money,"* Proceedings of ICALP '93, 1993.

[GMW86] O. Goldreich, S. Micali, A. Wigderson, *"Proofs that yield nothing but their validity and a methodology of cryptographic protocol design,"* Proceedings of IEEE FOCS '86, p.174-187, 1986.

[HMP94a] P. Horster, M. Michels, H. Petersen, *"Meta-ElGamal signature schemes,"* Proceedings of 2nd ACM CCS manuscript, pp. 96-107, 1994.

[HMP94b] P. Horster, M. Michels, H. Petersen, *"Meta Message Recovery and Meta Blind signature schemes based on the discrete logarithm problem and their applications,"* Advances in Cryptology – ASIACRYPT '94, LNCS 917, pp. 224-237, 1994.

[MAS97] S. Miyazaki, M. Abe, K. Sakurai, *"Partially Blind Signature Schemes for the DSS and for a Discrete Log. based Message Recovery Signature,"* Proceedings of the 1997 Korea-Japan Joint Workshop on Information Security and Cryptology, pp. 217-226, 1997.

[NIST] NIST FIPS PUB XX, Digital Signature Standard(DSS), National Institute of Standards and Technology, U.S. Department of Commerce, DRAFT, 1993.

[NMV97] K. Q. Nguyen, Y. Mu, V. Varadharajan, *A new digital cash scheme based on blind Nyberg-Rueppel digital signature,"* Pre-Proceedings of 1997 Information Security Workshop, pp. 219-226, 1997.

[NR93] K. Nyberg, R. A. Rueppel, *"A new signature scheme based on the DSA giving message recovery,"* Proceedings of 1st ACM CCS manuscript, 1993.

[PS96] D. Pointcheval, J. Stern, "*Provably Secure Blind Signature Schemes*," Advances in Cryptology – ASIACRYPT '96, LNCS 1163, pp. 252-265, 1996.

[Sch91] C. P. Schnorr, *"Efficient signature generation by smart cards,"* Journal of Cryptology, pp. 161-174, 1991.

[Scho95] B. Schoenmakers, "*An efficient electronic payment system with standing parallel attacks*," Technical report, CWI, 1995.
`http://www.cwi.nl/ftp/CWIreports/AA/CSR9522.ps.Z`

[Yac94] Y. Yacobi, *"Efficient electronic money,"* Advances in Cryptology – ASIACRYPT '94, LNCS 917, pp. 153-163, 1994.

A The Double-registration problem on Nguyn-Mu-Varadharajan's digicash scheme

In the system [NMV97], Bank chooses two random integers w_1, w_2 and computes $g_1 = g^{w_1} \mod p, g_2 = g^{w_2} \mod p$, where (p, q, g) are public informations satisfying that $g^q = 1 \mod p$. Then, Bank computes $h_1 = g_1^x \mod p, h_2 = g_2^x \mod p$ with his secret key x. (g_1, g_2, h_1, h_2) are made public. The user has own secret key u and public key $w = g_1 g_2^u \mod p$. Bank registers u with the database as the user's identity, then sends the certificate $w = v^x \mod p$ to the user. The user makes withdrawals and payments with (v, u, w). In case the E-money has been double-spent in the system, Bank derives u from the partial informations of the E-money double-spent and computes the User's identity v finally. We tried to apply the partially blind signature scheme to the this system[NMV97], but found out that this system has a vulnerability to the attack as follows.

User A, who has (v_A, u_A, w_A), might obtain the another keys and certificate (v_B, u_B, w_B) by the double-registration or conspiring with the other(User B). Then, User A can generate the unauthorized keys and certificate $(\tilde{v}, \tilde{u}, \tilde{w})$ by himself from the distinct components, (v_A, u_A, w_A) (v_B, u_B, w_B), as follows.

$$\begin{aligned}
\tilde{u} &= (u_A + u_B)2^{-1} \pmod q \\
\tilde{v} &= (v_A v_B)^{2^{-1}} \pmod p \\
&= (g_1^2 g_2^{u_A + u_B})^{2^{-1}} \pmod p \\
&= g_1 g_2^{(u_A + u_B)2^{-1}} \pmod p
\end{aligned}$$

$$
\begin{aligned}
&= g_1 g_2^{\tilde{u}} \pmod{p} \\
\tilde{w} &= (w_A w_B)^{2^{-1}} \pmod{p} \\
&= ((g_1 g_2^{u_A})^x (g_1 g_2^{u_B})^x)^{2^{-1}} \pmod{p} \\
&= (g_1 g_2^{(u_A+u_B)2^{-1}})^x \pmod{p} \\
&= \tilde{v}^x \pmod{p}
\end{aligned}
$$

Here, User A withdraws the E-money with $(\tilde{v}, \tilde{u}, \tilde{w})$ and makes payments with this E-money and the generated identity $\tilde{u}$ in order to double-spend it without trouble.

This article was processed using the LaTeX macro package with LLNCS style