Fair Blind Threshold Signatures in Wallet with Observers

Wen-Shenq Juang[†], Horng-Twu Liaw[†], Chin-Laung Lei[‡] and Po-Chou Lin[†]

Department of Information Management[†]
Shih Hsin University
Taipei, Taiwan, 116, R.O.C.
Email: [wsjuang, htlaiw]@cc.shu.edu.tw

Department of Electrical Engineering[‡]
National Taiwan University
Taipei, Taiwan, 116, R.O.C.
Email: lei@cc.ee.ntu.edu.tw

Abstract

In this paper, we propose efficient fair blind (t, n) threshold signature schemes in wallet with observers. By these schemes, any t out of n signers in a group can represent the group to sign fair blind threshold signatures, which can be used in anonymous e-cash systems. Since blind signature schemes provide perfect unlinkability, such e-cash systems can be misused by criminals, e.g. to safely obtain a ransom or to launder money. Our schemes allow the judge (or the judges) to deliver information allowing anyone of the t signers to link his view of the protocol and the message-signature pair.

Keywords: Fair Blind Signatures, Threshold Signatures, Wallet with Observers, Discrete Logarithm, Secure E-Cash Systems.

1 Introduction

The concept of blind signature was introduced by Chaum [4]. It allows a requester to obtain signatures on the messages she/he provides to the signer without revealing these messages. The blind signatures can realize secure electronic payment schemes [4, 5, 8, 17] protecting customers' anonymity. In a distributed environment, the signed blind messages can be regarded as a fixed amount of electronic money in secure electronic payment schemes. The basic assumption of these schemes is that the single money issuer of these schemes is trustworthy. However, the money issuer may issue extra e-coins as she/he wishes. If the money issuer does that, it may cause great danger or hurt for the corporation or society. To cope with this dilemma, instead of a single signer, blind threshold signature schemes and their variations [12, 15] have been proposed in a distributed environment, where several signers work together to sign a blind threshold signature. The schemes proposed in [12, 15] allows t out of n participants in a group cooperating to sign a blind threshold signature without the assistance of a single trusted authority.

Since blind signature schemes provide perfect unlinkability, such e-cash schemes can be misused by criminals, e.g. to safely obtain a ransom or to launder money [21]. To cope with this dilemma, the concept of fair blind signatures is introduced in [22]. In [22], three fair blind signature schemes are introduced to prevent the misuse of the unlinkability property. With the help of the judge, the signer can link a signature to the corresponding signing process. In [14], Juang et. al. proposed a fair blind threshold signature scheme based on the blind threshold signature scheme proposed in [12] and the registration method proposed in [22]. It allows the judge to deliver information allowing anyone of the t signers to link his view of the protocol and the message-signature pair. But the scheme in [14] needs more exponential operations than the scheme in [12].

In [6], Chaum et. al. proposed the concept of wallet databases with observers. It uses the tamper-proof devices, such as Java cards, that the person cannot modify or probe, to keep some correct and secret database. In this concept, a person (customer) can use two modules to handle ordinary consumer transactions: (1) the tamper-proof module, called an observer, whose inner working is programmed by a trusted authority; and (2) the personal workstation whose inner working is totally under control of the person. By this combined device, called a wallet, the two modules owned by a person can keep his personal secret database and ensure the correctness of these databases. In [1], Brands also use this concept and the rep-

resentation problem to design an off-line cash system.

In this paper, we propose a fair blind threshold signature scheme based on the blind threshold signature scheme proposed in [12] and the concept of wallet with observers proposed in [6]. In our scheme, the size of a fair threshold signature and the signature verification process are all the same as that of an individual signature. The security of our schemes relies on the difficulty of computing discrete logarithm and the tamper-proof devices and it is computationally infeasible for signers to derive the exact correspondence between the message they actually sign and all signers' complete views of the execution of the signing process without the assistance of the judge (or judges) or the requester.

2 The proposed scheme

In this section, we propose fair blind threshold signature schemes in wallet with observers. For simplicity, the fair blind threshold signature scheme is based on the Nyberg-Rueppel blind signature scheme [2] with message recovery. All secure Meta-ElGamal blind signature schemes proposed in [2, 11] can be used in our scheme. In a typical signing process of a fair blind threshold signature scheme, there are three kinds of participants, the signers, the judge and a requester. Before the requester can obtain a signature from the signers, all the signers have to cooperate to distribute their secret shadows to other signers in advance. Then the requester installs his temper-proof device with the judge and uses the wallet to request a fair blind threshold signature from the signers. The proposed scheme consists of four phases: (1) the shadow distribution phase, (2) the initialization phase, (3) the signature generation phase and (4) the signature verification phase. The shadow distribution phase is performed only once among the signers and then they can use their secret shadows to sign messages. In the initialization phase, the requester requests one pseudonym from the judge. The public key of the pseudonym is signed by the judge by a secure blind signature scheme and the corresponding secret key is stored in the tamper-proof device and known only by this device. These process is performed only once and this tamper-proof device can be used until it expires. In the signature generation phase, a requester requests a fair blind threshold signature from the signers. Before the requester can generate the real threshold signature from signers, she/he must send the unblinding information encrypted by the judge's public key to the judge. This information is also stored in the wallet databases and it contains necessary information to link message-signature pair. Thus, the judge, who knows the corresponding secret key, can link the message-signature pair with the corresponding signer's view when necessary. In the signature verification phase, anyone can use the group public key to verify if a threshold signature is valid.

Let $n^{'}$ be the number of signers before the shadow distribution phase, $QUAL$ be the set of non-disqualified signers after the shadow distribution phase, let n be the number of non-disqualified signers $QUAL$. Let $\mathcal{U}_i, 1 \leq i \leq n^{'}$, be the identification of signer i before the shadow distribution phase. Let $U_i, 1 \leq i \leq n$, be the identification of non-disqualified signer i after the shadow distribution phase. Let C be the computer controlled by the requester, T be the tamper-proof device issued by the judge (or some trusted authority) for the requester, n be the number of signers, t be the threshold value of the fair blind threshold signature scheme, so that at least $(n-t+1)$ signers are honest. Let d_T be the secret key stored in T when T is born and e_T be the corresponding public key. Let m be the blind message to be signed, $\mathcal{H}$ be a secure one-way hashing function [23]. Let p and q be two large strong prime numbers such that q divides $(p-1)$, and let ρ and ζ be two generators of Z_p^* (*i.e.*, $\gcd(\rho, p) = 1, \gcd(\zeta, p) = 1$, $\rho \neq 1, \zeta \neq 1$) and ζ be a random value generated by a generic distributed coin flipping protocol. Let $g \equiv_p \rho^{(p-1)/q}$ and $h \equiv_p \zeta^{(p-1)/q}$. Let " $\cdot$ " denote the ordinal string concatenation. Let d_i be the secret key chosen by $\mathcal{U}_i$ and d_J be the secret key chosen by the judge. In a distributed environment, $\mathcal{U}_i$ and the judge can publish their corresponding public keys e_i and e_J. Anyone can get e_T, e_i and e_J via some authentication service (e.g. the X.509 directory authentication service [23]). Using a secure public key signature scheme [7, 20], T, $\mathcal{U}_i$ and the judge can produce signatures of messages by their own secret keys d_T, d_i and d_J. Anyone can verify these signatures by the corresponding public keys e_T, e_i and e_J. Let $Cert_{d_T}(m)$ be the signature on the message m produced by T, $Cert_{d_z}(m)$ be the signature on the message m produced by T with the secret key d_z of its pseudonym requested in the initialization

phase, and $Cert_J(m)$ be the signature on the message m produced by the judge. For making our scheme clear, we assume that the message transmitted in the following protocol is via an authentication scheme (e.g. the RSA signature scheme); that is, no one can fake any other's messages and no one can deny the messages he really transmitted.

2.1 The shadow distribution phase

Before a requester can request a fair blind threshold signature from the signers, all the signers must cooperate to distribute their secret shadows to other signers without the assistance of a mutually trusted authority. In this phase, signers can detect the incorrect shares by the verification equations. In the shadow distribution phase, each $\mathcal{U}_i$, $1 \le i \le n'$, carries out the following steps:

1. $\mathcal{U}_i$ chooses a secret key $z_i \in Z_q$ and two secret polynomials $f_i(x) = \sum_{k=0}^{t-1} a_{i,k}x^k$ and $f_i^{'}(x) = \sum_{k=0}^{t-1} a_{i,k}^{'}x^k$ such that $a_{i,0} = z_i$, it computes $G_{i,k} \equiv_p g^{a_{i,k}}h^{a_{i,k}^{'}}$, $0 \le k \le t-1$, and it sends $(G_{i,k}, 0 \le k \le t-1))$ to $\mathcal{U}_j$, $1 \le j \le n'$, $j \ne i$.

2. Upon receiving $(G_{j,k}, 1 \le j \le n'$, $j \ne i$, $0 \le k \le t-1)$ from all other signers, $\mathcal{U}_i$ sends $\delta_{i,j} \equiv_q f_i(x_j)$ and $\delta'_{i,j} \equiv_q f'_i(x_j)$, where x_j is a unique public number for $\mathcal{U}_j$, secretly to every $\mathcal{U}_j$, $1 \le j \le n'$, $j \ne i$.

3. When $\mathcal{U}_i$ receives all $\delta_{j,i}$ and $\delta'_{j,i}$, $1 \le j \le n'$, $j \ne i$, from other signers, she/he verifies if the shares $\delta_{j,i}$ and $\delta'_{j,i}$received from $\mathcal{U}_j$ is consistent with the certified values $G_{j,l}$, $0 \le l \le t-1$, by checking whether $g^{\delta_{j,i}}h^{\delta'_{j,i}} \equiv_p \prod_{l=0}^{t-1}(G_{j,l})^{{x_i}^l}$. If it fails, $\mathcal{U}_i$ broadcasts that an error has been found, publishes $\delta_{j,i}$ and $\delta'_{j,i}$, the authentication information of $\delta_{j,i}, \delta'_{j,i}$ and $\mathcal{U}_j$. Each signer except the dishonest signer $\mathcal{U}_j$ then marks $\mathcal{U}_j$ as a disqualified signer and builds the set of non-disqualified signers $QUAL$.

4. Every signer $\mathcal{U}_i, i \in QUAL$, broadcasts $A_{i,l} \equiv_p g^{a_{i,l}}$, $0 \le l \le t-1$.

5. When $\mathcal{U}_i$,$i \in QUAL$, receives all $A_{j,l}$, $j \in QUAL$, $j \ne i, 0 \le l \le t-1$, from other signers in $QUAL$, she/he verifies whether $g^{\delta_{j,i}} \equiv_p \prod_{l=0}^{t-1}(A_{j,l})^{{x_i}^l}$. If this check fails for an index j, $\mathcal{U}_i$ broadcasts that an error has been found, publishes $\delta_{j,i}$, the authentication information of $\delta_{j,i}$ and $\mathcal{U}_j$. Any t signers in $QUAL$ can compute $z_j, f_j(x), A_{j,k}, 0 \le k \le t-1$. Anyone then computes the public shadows $\mathcal{P}_{j,i} \equiv_p g^{\delta_{j,i}} \equiv_p \prod_{l=0}^{t-1}(A_{j,l})^{{x_i}^l}$, where i and $j \in QUAL$, and the group public key $y \equiv_p \prod_{j \in QUAL} y_j \equiv_p \prod_{j \in QUAL} A_{j,0}$. The group public key y and all public shadows $\mathcal{P}_{j,i}$, where i and $j \in QUAL$, the personal public key $y_i \equiv_p A_{i,0} \equiv_p g^{z_i}$ can then be published by each signer $\mathcal{U}_i$. Without loss of generality, we assume that n non-disqualified signers $QUAL$ are U_i, $1 \le i \le n$. It can be done by renaming the index of each signer $\mathcal{U}_i, i \in QUAL$.

2.2 The initialization phase

Before a requester can request a fair blind threshold signature from the signers, she/he must acquire one pseudonym from the judge. The public key of the pseudonym is signed by the judge by a secure blind signature scheme [2, 4, 11] and the corresponding secret key is stored in the temper-proof device T issued by some organization (e.g. the judge) and known only by this device T. The requester and the judge then carry out the following steps:

1. T sends a request information including the certificate $Cert_{d_T}(\mathcal{H}(RD))$, where RD contains some redundancy information indicating the registration, for a pseudonym to the judge.

2. The judge first verifies $T's$ identification by the certificate $Cert_{d_T}(\mathcal{H}(RD))$ using his corresponding public key e_T, and then use any secure blind signature scheme to issue a pseudonym for T. Let d_z be the secret key chosen by T and e_z be the corresponding public key. After the blind signature generation process, the secret key d_Z and the certificate $Cert_J(\mathcal{H}(e_z))$ of the corresponding public key e_z is stored in T.

2.3 The signature generation phase

Without loss of generality, we assume that t out of the n signers are U_i, $1 \le i \le t$. When a requester (C and T) requests a fair blind threshold signature, she/he, the judge, and the t sign-

ers perform the following steps during the signature generation phase.

1. Each U_i randomly chooses a number $k_i \in Z_q$, computes $\widehat{r_i} \equiv_p g^{k_i}$ and sends $\widehat{r_i}$ to the requester.

2. After receiving all $\widehat{r_i}, 1 \le i \le t$, C does the following.

   (a) Choose two random numbers $\alpha \in Z_q$ and $\beta \in Z_q^*$, compute $r \equiv_p m\prod_{i=1}^{t} r_i \equiv_p mg^{t\alpha}(\prod_{i=1}^{t}\widehat{r_i})^{\beta}$ and $\widehat{m} \equiv_q \beta^{-1}r$, where $r_i \equiv_p g^{\alpha}\widehat{r_i}^{\beta}$ and $1 \le i \le t$.

   (b) Check if $\widehat{m} \neq 0$. If yes, sends $(\alpha, \beta, \widehat{r_i}, 1 \le i \le t, m)$ to T. Otherwise, go back to step (a).

   (c) T also computes $r \equiv_p m\prod_{i=1}^{t} r_i \equiv_p mg^{t\alpha}(\prod_{i=1}^{t}\widehat{r_i})^{\beta}$, $\widehat{m} \equiv_q \beta^{-1}r$, where $r_i \equiv_p g^{\alpha}\widehat{r_i}^{\beta}$ and $1 \le i \le t$, $Cert_{d_Z}(\mathcal{H}(\widehat{m}))$, and sends $E_{e_J}(\alpha \cdot \beta \cdot \widehat{r_1} \cdot ... \cdot \widehat{r_t} \cdot m)$ to the judge.

   (d) After receiving the receipt from the judge, T sends $Cert_{d_Z}(\mathcal{H}(\widehat{m}))$ back to C.

   (e) C then sends $Cert_J(\mathcal{H}(e_z))$, $e_z, Cert_{d_Z}(\mathcal{H}(\widehat{m}))$, $\widehat{m}$ to all $U_i, 1 \le i \le t$.

3. Upon receiving $\widehat{m}$, each U_i verifies if $Cert_{d_Z}(\mathcal{H}(\widehat{m}))$ is valid. If yes, she/he computes $\widehat{s_i} \equiv_q \widehat{m}(z_i + \sum_{j=t+1}^{n} f_j(x_i)(\frac{-x_k}{x_i - x_k}))) + k_i$ and sends $\widehat{s_i}$ back to the requester.

4. After receiving all $\widehat{s_i}$, C computes $s_i \equiv_q \widehat{s_i}\beta + \alpha$, and checks if $g^{-s_i}y_i^r r_i \equiv_p (\prod_{j=t+1}^{n}(\mathcal{P}_{j,i}))^{(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))(-r)}$, $1 \le i \le t$.

If any of the $\widehat{s_i}$ is not valid, it has to ask the corresponding signer to send it again. Otherwise, it computes $s \equiv_q \sum_{i=1}^{t} s_i$. The threshold signature of m is (r, s).

2.4 The signature verification phase

To verify the threshold signature (r, s), one simply computes $m \equiv_p g^{-s}y^r r$ and checks if m has some redundancy information. If m has no proper redundancy, a secure one-way hashing function $\mathcal{H}$ can be applied to m. But this approach can not provide the message recovery capability. To verify the threshold signature (r, s) on m without redundancy, one must send m along with (r, s) to the verifier.

3 Analysis

We examine the correctness and security of our scheme in this section. We also show how to link a given signature to its corresponding signing process under the assistance of the judge.

3.1 Correctness

To prevent a signer from sending an invalid partial signature to the requester, the partial signature must be checked in step 4 of the signature generation phase. The following lemma ensures the correctness of partial signatures.

Lemma 1. *The partial signature (r_i, s_i) is valid if U_i is honest.*

By means of our scheme, we have

$$\begin{aligned}
& g^{-s_i}y_i^r r_i \\
\equiv_p\ & g^{-(\widehat{s_i}\beta+\alpha)}g^{z_i r}\ g^{\alpha}\widehat{r_i}^{\beta} \\
\equiv_p\ & g^{-(\widehat{m}(z_i+\sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k})))+k_i)\beta} \\
& g^{z_i r}g^{k_i\beta} \\
\equiv_p\ & g^{-\widehat{m}(z_i+\sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k})))\beta}g^{z_i r} \\
\equiv_p\ & g^{-\widehat{m}z_i\beta-\widehat{m}\sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))\beta}g^{z_i r} \\
\equiv_p\ & g^{\sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))(-\widehat{m}\beta)} \\
\equiv_p\ & (\prod_{j=t+1}^{n}(\mathcal{P}_{j,i}))^{(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))(-r)}
\end{aligned}$$

□

After the signature generation phase, the blind threshold signature can be verified by the group public key in the signature verification phase. Lemma 2 ensures the correctness of the scheme.

Lemma 2. The signature (r, s) generated in the signature generation phase is a valid blind threshold signature on message m for the Nyberg-Rueppel signature scheme.

Proof. The validity of the signature (r, s) can easily be established as follows.

$$\begin{aligned}
& g^{-s}y^r r \\
\equiv_p\ & g^{-(\sum_{i=1}^{t}(\widehat{s_i}\beta+\alpha))}g^{\sum_{i=1}^{n} z_i r}m(\prod_{i=1}^{t} r_i) \\
\equiv_p\ & mg^{-(\widehat{m}(\sum_{i=1}^{t} z_i+\sum_{i=1}^{t}(\sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\\
& \frac{-x_k}{x_i-x_k})))+\sum_{i=1}^{t} k_i)\beta-t\alpha}g^{\sum_{i=1}^{n} z_i r}(\prod_{i=1}^{t} g^{\alpha}\widehat{r_i}^{\beta}) \\
\equiv_p\ & mg^{-(\widehat{m}(\sum_{i=1}^{t} z_i+\sum_{j=t+1}^{n}(\sum_{i=1}^{t} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\\
& \frac{-x_k}{x_i-x_k})))+\sum_{i=1}^{t} k_i)\beta}g^{\sum_{i=1}^{n} z_i r}(\prod_{i=1}^{t} g^{k_i\beta})
\end{aligned}$$

$\equiv_p mg^{-(\widehat{m}(\sum_{i=1}^{t} z_i + \sum_{i=t+1}^{n} z_i))\beta} g^{\sum_{i=1}^{n} z_i r}$
$\equiv_p mg^{-\widehat{m} \sum_{i=1}^{n} z_i \beta} g^{\sum_{i=1}^{n} z_i r}$
$\equiv_p mg^{-r \sum_{i=1}^{n} z_i} g^{\sum_{i=1}^{n} z_i r}$
$\equiv_p m.$ □

3.2 Security considerations

In [10], Gennaro et. al. proposed an improved distributed key generation scheme based on discrete logarithm. In this scheme, they use the information-theoretic verifiable secret sharing protocol [18] to guarantee that no bias for a bit in the output group public key of the protocol is possible. The shadow distribution phase of our proposed scheme is based on the distributed key generation scheme in [10]. Different from the scheme in [10], in order to do cheater detection when some signer cheats, the public shadows ($\mathcal{P}_{j,i} \equiv_p g^{\delta_{j,i}} \equiv_p \prod_{l=0}^{t-1} (A_{j,l})^{x_i{}^l}$,where i and $j \in QUAL$) will be published by all signers. All the public shadows ($\mathcal{P}_{j,i}$, where j and $i \in QUAL$) can be computed by the public values $A_{j,l} \equiv_p g^{a_{j,l}}, j \in QUAL\ , 0 \leq l \leq t-1$, broadcasted in Step 4 of the shadow distribution phase. This public shadows will not disclose any extra information of the group secret key.

Since blind threshold signature schemes without the fairness property provide perfect unlinkability, such e-cash schemes can be misused by criminals, e.g. to safely obtain a ransom or to launder money. For example, a criminal can safely obtain a ransom by joining a blind threshold signature scheme where the request is via an untraceable mail (e.g. an ordinary mail or an untraceable e-mail [3, 13]) and the signers put the blind threshold signature on a public board. Then the criminal can easily obtain the blind threshold signature from the public board and derive the corresponding e-coins. To cope with this dilemma, in our proposed scheme, anyone of the t signers U_i can first send the messages $(\widehat{r_i}, \widehat{m})$ requested by the criminal to the judge and then the judge sends all the corresponding view $(\alpha \cdot \beta \cdot \widehat{r_1} \cdot ... \cdot \widehat{r_t} \cdot m)$ back to the signer. The signer can verify validity of the corresponding view by computing $r \equiv_p m \prod_{i=1}^{t} r_i \equiv_p mg^{t\alpha} (\prod_{i=1}^{t} \widehat{r_i})^{\beta}$, $\widehat{m} \equiv_q \beta^{-1} r$. When the criminal withdraws these e-coins from the signer, the signer can easily identify the criminal by linking the message-signature pair (m, r, s) with the corresponding signer's view k_i, $\widehat{r_i} \equiv_p g^{k_i}$,and $\widehat{m}$. If the judge is honest, all crimes by misusing the unlinkability property of blind threshold signatures will be prevented and the anonymity of honest customers will also be preserved.

4 Discussions

4.1 Extension schemes

The blind signature scheme proposed in [2] with message recovery is used in our proposed scheme. The modification of DSA-type blind signature scheme proposed in [2] can also be used in our scheme. In [11] some extensions of the blind signature schemes in [2] were introduced. As mentioned in [11], not all variants of Meta-Message recovery signature schemes can be transformed to blind signature schemes. For example, there is no blind signature scheme for the original ElGamal signature scheme yet. All extensions of secure blind signature schemes proposed in [11], except that $\widetilde{B}$ contains $\widetilde{s}$ in the signature generation equation, can be used in our scheme. The security considerations and performance analysis of these extended schemes are similar to those of our proposed scheme. In [19], two provably secure blind signature schemes are proposed. One has been proved to be equivalent to the discrete logarithm problem in a subgroup. The other has been proved to be equivalent to the RSA problem. By suitable modifications for our scheme, the secure blind signature scheme based on discrete logarithm in [19] can also be used in the modified scheme [16]. Since the security of the underlying blind signature scheme has been proven to be equivalent to the discrete logarithm problem in the random oracle model, the security of this modified scheme [16] is also equivalent to the discrete logarithm problem in the random oracle model.

4.2 Distributing the power of a single judge to multi-judges

In our scheme, the duty of the judge is to issue pseudonyms to requesters and keep the unblinding information sent from the requesters. In some situations, it is hard to find a trusted judge. We can modify the scheme in Section 2 as follows: (1) Instead of a unique judge, the modified system consists of κ judges and at least $\lceil \frac{\kappa}{2} \rceil$ judges are honest. (2) These κ judges execute a distributed key generation protocol similar to the shadow distribution phase in Section 2 to generate a group public key e_{Js} and the corresponding group secret key d_{Js}.

Table 1: Cost of the signature generation phase and the signature verification phase in our scheme and that in [14].

	The requester (verifier)				
	EXP	INV	ENC	MUL	ADD
Cost of the signature generation phase					
$\mathcal{A}$	2	1	1	$t+5$	t
$\mathcal{B}$	5	1	0	$3t+6$	t
Cost of the verification phase					
$\mathcal{A}$	2	1	0	2	0
$\mathcal{B}$	4	1	0	3	0

where
EXP = no. of modulo exponentiations,
INV = no. of modulo inversions,
ENC = no of message encryption,
MUL = no. of modulo multiplications,
ADD = no. of modulo additions,
$\mathcal{A}$ = Our scheme,
$\mathcal{B}$ = The scheme in [14].

(3) During the initialization phase, a requester must acquire one threshold pseudonym from $\lceil \frac{\kappa}{2} \rceil$ judges. The public key of the threshold pseudonym is signed by these judges by a blind threshold signature scheme and the corresponding secret key is stored in the temperproof device T and known only by this device. (4) In the signature generation phase, any requester must send the unblinding information $E_{e_{Js}}(\alpha \cdot \beta \cdot \widehat{r_1} \cdot ... \cdot \widehat{r_t} \cdot m)$ to any honest judge (just as a database manager). By the above modifications, the power of a single judge is distributed to several judges. When anyone of the t signers U_i sends the messages $(\widehat{r_i}, \widehat{m})$ requested by the criminal to the judge (as a database manager) and then $\lceil \frac{\kappa}{2} \rceil$ judges can first decrypt the unblinding information $E_{e_{Js}}(\alpha \cdot \beta \cdot \widehat{r_1} \cdot ... \cdot \widehat{r_t} \cdot m)$ received soon, find $(\alpha \cdot \beta \cdot \widehat{r_1} \cdot ... \cdot \widehat{r_t} \cdot m)$ and send all the corresponding view $(\alpha \cdot \beta \cdot \widehat{r_1} \cdot ... \cdot \widehat{r_t} \cdot m)$ back to the signer. By this approach, the power of a single judge is distributed to several judges. The fair blind threshold signature scheme proposed in [14] is just a single judge. It is still an open problem that whether there exists an efficiently multi-judges fair blind threshold signature scheme without the assistance of a tamperproof device like the scheme in [14].

4.3 Performance Considerations

In this subsection we give an analysis of the computational effort required to compute fair blind threshold signatures in our scheme. Table 1 illustrates the comparison of our proposed scheme and the scheme in [14]. For reducing the computational cost needed by the requester, the partial signature verification in Step 4 would not be done except the final threshold signature can not pass the verification equation in the signature verification phase. The requester does not need to know the public shadows $\mathcal{P}_{l,j}$, where l and $j \in QUAL$, in advance except there exists some dishonest signer in the signature generation phase. In this approach, the requester only needs to compute 2 modular exponentiations and 1 modular inverse in step 2 of the signature generation phase. Since the blind threshold verification functions of our schemes all are the same as those of the underlying blind signature schemes, the verification cost of our blind threshold signature is the same as that of the underlying blind signature. Comparative to the scheme in [14], the extra cost for requesting a fair blind threshold signature in our scheme is to compute $E_{e_J}(\alpha \cdot \beta \cdot \widehat{r_1} \cdot ... \cdot \widehat{r_t} \cdot m)$ which contains one public key encryption. For reducing the computation cost, T can first negotiate a session key with the judge and then send the unblinding information to the judge by a secret key cryptosystem. This approach will greatly reduce the computation cost when the number of signers t is large. But if we want to change our scheme to multi-judges environments, instead of the above approach, T must send the unblinding information $E_{e_{Js}}(\alpha \cdot \beta \cdot \widehat{r_1} \cdot ... \cdot \widehat{r_t} \cdot m)$ to a judge (as a database manager) by the judges' group public key e_{Js} for distributing the power of a single judge.

Let the fair blind threshold signature of m in [14] be $(\Omega_1, Cert_J(h(\Omega_1)), v_1, v_2, s, u)$. Let the prime p be 1024-bits long and the prime q be 160-bits long. Totally, the fair threshold signature in [14] and our proposed scheme in section 2 are 1024+1024+1024+1024+160+1024= 5280 bits and 160+1024=1184 bits, respectively. Hence, our proposed scheme reduces the length of the fair blind threshold signature by $\frac{5280-1184}{5280} = 78\%$.

In [9], three robust threshold signature protocols, namely, DSS-Thresh-Sig-1, DSS-Thresh-Sig-2 and DSS-Thresh-Sig-3, are proposed. One approach to generate blind threshold sig-

natures is to take robust threshold signature schemes [9] and turn them into fair blind signature schemes. The advantage of this approach is that it is quite robust and can deal with the situation where there are many cheaters. However, in DSS-Thresh-Sig-1, $2t + 3$ modular exponentiations are required for each signer to generate a threshold signature and it is even worse for DSS-Thresh-Sig-2 and DSS-Thresh-Sig-3 which requires $O(nt)$ modular exponentiations. It is clear that this approach is quite inefficient compared to our proposed schemes.

5 Conclusion

We have proposed efficient fair blind threshold signature schemes in wallet with observers. In our schemes, the size of a fair threshold signature and the signature verification process are all the same as that of an individual signature. The security of our schemes relies on the difficulty of computing discrete logarithm and the tamper-proof devices and it is computationally infeasible for signers to derive the exact correspondence between the message they actually sign and all signers' complete views of the execution of the signing process without the assistance of the judge (or judges) or the requester. Our proposed schemes can be easily applied to current efficient single-authority e-cash schemes for distributing the power of a single authority without changing the underlying structure and degrading the overall performance.

Acknowledgment

This work was supported in part by the National Science Council of the Republic of China under contract NSC-90-2213-E-128-004.

References

[1] S. Brands, "Untraceable off-line cash in wallet with observers," *Advances in Cryptology: Proc. of EuroCrypt'94*, LNCS 950, pp. 428-432, Springer-Verlag, 1995.

[2] J. Camenisch, J. Pivereau and M. Stadler, "Blind signatures based on the discrete logarithm problem," *Advances in Cryptology: Proc. of EuroCrypt'94*, LNCS 950, pp. 428-432, Springer-Verlag, 1995.

[3] D. Chaum. "Untraceable electronic mail, return addresses, and digital pseudonyms," *Commu. of the ACM*, Vol. 24, No. 2, pp. 84-88, 1981.

[4] D. Chaum, "Blind signatures for untraceable payments," *Advances in Cryptology: Proc. of Crypt'82*, pp. 199-203, Plenum, NY, 1983.

[5] D. Chaum, "Privacy protected payments: unconditional payer and/or payee untraceability," *In Smartcard 2000*, North Holland, 1988.

[6] D. Chaum and T. Pedersen, "Wallet databases with observers," *Advances in Cryptology: Proc. of Crypt'92*, LNCS 740, pp. 89-105, Springer-Verlag, 1993.

[7] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithm," *IEEE Trans. on Information Theory*, Vol. IT-31, No. 4, pp. 469-472, 1985.

[8] N. Ferguson, "Single term off-line coins," *Advances in Cryptology: Proc. of EuroCrypt'93*, LNCS 765, pp. 318-328, Springer-Verlag, 1993.

[9] R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, "Robust threshold DSS signatures," *Advances in Cryptology: Proc. of EuroCrypt '96*, LNCS 1070, pp. 354-371, Springer Verlag, 1996.

[10] R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, "Secure distributed key generation for discrete-log based cryptosystems," *Advances in Cryptology: Proc. of EuroCrypt'99*, LNCS 1592, pp. 295-310, Springer-Verlag, 2000.

[11] P. Horster, M. Michels and H. Petersen, "Meta-message recovery and meta-blind signature schemes based on the discrete logarithm problem and their applications," *Advances in Cryptology: Proc. of AisaCrypt'94*, LNCS 917, pp. 224-237, Springer-Verlag, 1994.

[12] W. Juang and C. Lei, "Blind threshold signatures based on discrete logarithm," *Proc. of Second Asian Computing Science Conference on Programming, Concurrency and Parallelism, Networking and Security*, LNCS 1179, pp. 172 -181, Springer-Verlag, 1996.

[13] W. Juang, C. Lei and C. Chang, "Anonymous Channel and Authentication in Wireless Communications," *Computer Communications*, Vol. 22, No. 15-16, pp. 1502-1511, 1999.

[14] W. Juang, C. Lei and H. Liaw, "Fair blind threshold signatures based on discrete logarithm," *International Journal of Computer Systems Sciences & Engineering,* Vol. 16, No. 6, pp. 371-379, 2001.

[15] W. Juang and C. Lei, "Partially blind threshold signatures based on discrete logarithm," *Computer Communications*, Vol. 22, No. 1, pp. 73-86, 1999.

[16] C. Lei, W. Juang and P. Yu, "Provably secure blind threshold signatures based on discrete logarithm," to appear in *Journal of Information Science and Engineering* (A preliminary version was presented at the National Computer Symposium, Taipei, Taiwan, pp. (C-198)-(C-205), 1999.)

[17] T. Okamoto and K. Ohta, "Universal Electronic cash," *Advances in Cryptology: Proc. of Crypt'91*, LNCS 576, pp. 324-337, Springer-Verlag, 1992.

[18] T. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," *Advances in Cryptology: Proc. of Crypt'91*, LNCS 576, pp. 129-140, Springer-Verlag, 1991.

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

[20] R. Rivest, A. Shamir and L. Adelman, "A method for obtaining digital signatures and public key cryptosystem," *Commun. ACM*, Vol. 21, No. 2, pp. 120-126, 1978.

[21] S. Solms and D. Naccache, "On blind signatures and perfect crime," *Computer & Security*, Vol. 11, pp. 581-583, 1992.

[22] M. Stadler, J. Piveteau and J. Camenisch, "Fair blind signatures," *Advances in Cryptology-EuroCrypt'95,* LNCS 921, pp. 209-219, Springer-Verlag, 1995.

[23] W. Stallings, *Cryptography and network security*, 2nd Edition, Prentice Hall International, 1999.