JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 18, 23-39 (2002)

Provably Secure Blind Threshold Signatures Based on Discrete Logarithm

CHIN-LAUNG LEI, WEN-SHENQ JUANG* AND PEI-LING YU
Department of Electrical Engineering
National Taiwan University
Taipei, 106 Taiwan
**Department of Information Management*
Shih Hsin University
Taipei, 116 Taiwan

In this paper, we propose a provably secure group-oriented blind (*t*, *n*) threshold signature scheme, which is the first scheme whose security is proved to be equivalent to the discrete logarithm problem in the random oracle model. Based on the scheme, any *t* out of *n* signers in a group can represent the group in signing blind threshold signatures, which can be used in anonymous digital e-cash systems or secure voting systems. By means of our proposed scheme, the issue of e-coins is controlled by several authorities. In our scheme, the size of a blind threshold signature is the same as that of an individual blind signature, and the signature verification process is equivalent to that for an individual signature.

Keywords: provably secure blind signatures, threshold signatures, discrete logarithm, secure e-cash systems, secure voting systems

1. INTRODUCTION

A blind signature scheme is an interactive protocol which involves two participants, a signer and a requester. A distinguishing property required by a typical blind signature scheme [1-5] is so-called "unlinkability," which ensures that requesters can prevent the signer from deriving the exact correspondence between the actual signing process performed by the signer and the signature which will later be made public. Blind signatures can make possible secure electronic payment systems [2, 6, 7] that protect customers' anonymity and secure voting systems [8-10] that preserve voters' privacy. In a distributed environment, every signed blind message can be thought of as a fixed amount of electronic money in a secure electronic payment system or as a ticket in an application like a secret voting system. To date, no security proof has been proposed for the schemes described in [1-3]. In [11], a cryptanalysis method for the blind signature schemes proposed in [1, 3] was presented. In [12], it was shown that the claim in [11] was, fortunately, incorrect; that is, the schemes proposed in [1, 3] remain secure. In [4], two provably secure blind signature schemes were 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. In [5], a blind signature scheme was proposed and

Received February 9, 2000; revised June 27 & September 7, 2000; accepted October 26, 2000.
Communicated by Hsu-Chun Yen.

proved to be equivalent to factorization.

Threshold signatures [13, 14] are motivated by the need that arises in organizations to have a group of employees who agree on a message before signing and by the need to protect the group private key from attacks launched by internal and external adversaries. The later becomes more important with the actual deployment of public key schemes in practice. The signing power of some authorities inevitably invites attackers to try to steal this power. The goal of a threshold signature scheme is to increase the availability of the signing authorities and to increase protection against forgery by making it harder for the adversary to learn the group secret key.

So far, the on-line e-cash systems proposed in [2, 6] are the most efficient and practical ones. The aim of these systems is to produce an electronic version of money which retains the properties of paper cash. In real world environments, if the issue of e-coins is controlled by a single person, then he can generate extra e-coins as he wishes. To cope with this dilemma, instead of a unique authority, every customer needs to request blind (t, n) threshold signatures [15, 16] as e-coins from t arbitrary authorities so that t arbitrary authorities can represent the bank in issuing e-coins.

In [8-10], several single-authority voting systems have been proposed. These systems involve voters and the authority, and can be simplified to the following three phases: the registration phase, the voting phase and the publication phase. During the registration phase, voters apply the blind signature technique to get their blind votes. In the voting phase, voters generate their real ballots from the blind votes received in the registration phase and send them to the authority via an untraceable e-mail [17-19]. Finally, in the publication phase, the authority publishes all the valid ballots. Since voters only need to communicate with the authority in these protocols, there is no global computation among voters. However, the authority can impersonate any voter who abstains from voting after the registration phase. To cope with this dilemma, instead of a unique authority, every voter needs to request blind (t, n) threshold signatures [15, 16] as ballots from t arbitrary authorities so that t arbitrary authorities can represent the tally center in issuing ballots. Through the above modifications, the power of a single authority is distributed among several authorities, and registered voters may abstain from voting after the registration phase.

No meta-blind threshold signature schemes [15, 16] have been proven to be secure based on some hard problems, e.g., the discrete logarithm problem. In this paper, we propose a provably secure blind threshold signature scheme, which is the first scheme whose security is proved to be equivalent to the discrete logarithm problem in the random oracle model. Our proposed scheme can be directly applied to secure e-cash systems or voting systems for distributing the power of a single authority. Modified e-cash systems or voting systems can satisfy real world environments without a single trusted authority or with some absent/dishonest authorities. In our scheme, the size of a blind threshold signature is the same as that of an individual blind signature, and the verification process of a blind threshold signature is equivalent to that of an individual blind signature.

This paper is organized as follows. In Section 2, we present the definition of blindness of a threshold signature scheme, and that of unforgeability of blind threshold signatures. In Section 3, we present a provably secure blind threshold signature scheme. Then, we discuss its correctness, security and performance in Section 4. Finally, concluding remarks are given in Section 5.

2. PRELIMINARY

In this section, we present the definition of blindness of a threshold signature scheme, and that of unforgeability of blind threshold signatures. There are two methods for verifying the validity of a signature: the comparison method and the restoration (message recovery) method [20]. In the comparison method, to verify a signature, the corresponding message must be sent to a verifier along with the signature. To reduce the length of the signature, instead of signing the whole message, one can make a signature on the digest of the message, which is the hashed value of a secure one-way hash function [21-24] with the message as input. In the restoration method, only the signature is sent to a verifier. The signed message that is embedded in the signature can be recovered after the verification process. Many signature schemes with message recovery have been proposed [25, 26].

Given a secret ω, we say that the secret shadows (ω_i, $1 \le i \le n$) construct a (t, n) threshold secret sharing ω if $t - 1$(or less) of these values reveal no information about ω, and if there exists a poly-time algorithm that outputs ω, which has t of these values as inputs.

Let there be $n > 1$ players in a distributed system, where player i has his own secret s_i. A secure computing protocol for this system is a procedure for evaluating the function value $f(s_1, s_2, ..., s_n)$ jointly by means of the n players such that the output becomes commonly known while s_i remains secret. A secure computing protocol can be used to define blind threshold signature schemes. We define the blindness of a (t, n) threshold signature scheme with the comparison method as follows:

Definition 1 A blind (t, n) threshold signature scheme with the comparison method is a 12-tuple $P_T = (M, S, \Delta, K, \Lambda, \Psi, \Re, \Omega_T, \partial_T, \Upsilon_T, \Phi_T, \Gamma)$, where:

- M is a message space that is a set of strings (plaintexts).
- S is a signature space that is a set of strings (signatures).
- Δ is a random message space that is a set of strings.
- $K = K_e \times K_d$ is a key space such that K_e is the public key space and K_d is the private key space.
- Λ is a shadow key space.
- $\Psi = \{U_i \mid i = 1, 2, \ldots, n\}$ is a set of n signers.
- $\Re$ is a set of requesters.
- Ω_T: $\Delta^n \to K_e$ is a poly-time distributed key generation protocol (secure computing protocol) used by all the signers Ψ. The secret input of U_i is a random string $\chi_i \in \Delta$. The output of the protocol is the group public key $K_e = \Omega_T(\chi_1, \chi_2, \ldots, \chi_n) \in K_e$. At the end of the protocol, the private output of signer $U_i \in \Psi$ is a secret shadow $\theta_i \in \Lambda$, such that the shadows θ_i, $1 \le i \le n$, form $a(t, n)$ threshold secret sharing $K_d \in K_d$, where K_d is the corresponding private key of K_e.
- ∂_T: $M \times \Delta \times K_e \times \Delta^t \to M$ is a poly-time blinding algorithm such that on input of a message $m \in M$, a random blinding string $\lambda \in \Delta$, a public key $K_e \in K_e$ and $H(\delta_{Pi}) \in \Delta$, $1 \le P_1, P_2, \ldots, P_t \le n$, where H is a one-way hash function $\delta_{Pi} \in \Delta$, constructs the blinded message $m' = \partial_T (m, \lambda, K_e, H(\delta_{P_1}), H(\delta_{P_2}), \ldots, H(\delta_{P_t})) \in M$.
- γ_T: $M \times K_e \times \Lambda^t \times \Delta^t \to S$ is a poly-time distributed signing protocol (secure computing

protocol) used by t signers $\{U_{P_i} \mid 1 \le P_1, P_2, ..., P_t \le n\}$. The private inputs of U_{P_i} are the secret shadow $\theta_{P_i} \in \Lambda$ and the randomizing factor $\delta_{P_i} \in \Delta$. The public inputs consist of a blind message $m' = \partial_T (m, \lambda, K_e, H(\delta_{P_1}), H(\delta_{P_2}), \ldots, H(\delta_{P_t})) \in M$ and the public key $K_e \in K_e$. The output of the protocol is the blind signature $s' = \gamma_T (m', K_e, \theta_{P_1}, \theta_{P_2}, \ldots, \theta_{P_t}, \delta_{P_1}, \delta_{P_2}, \ldots, \delta_{P_t}) \in S$.

- Φ_T: $S \times \Delta \to S$ is a poly-time unblinding algorithm such that on input of a blind signature $s' = \gamma_T (\partial_T (m, \lambda, K_e, H(\delta_{P_1}), H(\delta_{P_2}), \ldots, H(\delta_{P_t})), K_e, \theta_{P_1}, \theta_{P_2}, \ldots, \theta_{P_t}, \delta_{P_1}, \delta_{P_2}, \ldots, \delta_{P_t})) \in S$ and the random blinding string λ, extracts the signature $s = \Phi_T(s', \lambda)$ on m.
- Γ: $M \times S \times K_e \to \{true, false\}$ is a poly-time verification algorithm such that on input of a message- signature pair (m, s) and a public key $K_e \in K_e$, determines if s is a valid signature for message m.

Based on the above, we have the following:

1. In a blind threshold signature generation, the signers' views v and the message- signature pair (m, s), which is later made public, are statistically independent.

❑

Before a requester $R \in \Re$ can request a blind threshold signature from t signers $\Psi_t = \{U_{P_i} \mid 1 \le P_1, P_2, ..., P_t \le n\}$, all the signers in Ψ have to apply Ω_T to construct a group public key $K_e \in K_e$, where the corresponding group private key of K_e is $K_e \in K_e$. At the end of Ω_T, each signer $U_i \in \Psi$ gets a secret shadow $\theta_i \in \Lambda$. In a blind threshold signature generation, each signer $U_{P_i} \in \Psi_t$ first sends a hashed randomizing factor $H(\delta_{P_i})$ to R, where δ_{P_i} is the secret randomizing factor chosen by U_{P_i}. Then, R chooses a random string $\lambda \in \Delta$ for blinding a message m and computes $m' = \partial_T (m, \lambda, K_e, H(\delta_{P_1}), H(\delta_{P_2}), \ldots, H(\delta_{P_t}))$, where K_e is Ψ's group public key, and submits m' to $\Psi_t = \{U_{P_i} \mid 1 \le P_1, P_2, \ldots, P_t \le n\}$. Ψ_t then apply the distributed signing protocol Υ_T to m' and send R the signing result $s' = \Upsilon_T (m', K_e, \theta_{P_1}, \theta_{P_2}, \ldots, \theta_{P_t}, \delta_{P_1}, \delta_{P_2}, \ldots, \delta_{P_t}))$, where θ_{P_i} is the secret shadow of U_{P_i}. After receiving s', R extracts the signature $s = \Phi_T (s', \lambda)$ on the message m. Anyone can verify if a message-signature pair (m, s) is valid for the group public key $K_e \in K_e$ by means of the function Γ.

The digital signature scheme with the restoration method can be defined similarly except that the verification function Γ must be replaced by a restoration function Θ. To verify a signature $s \in S$, one simply computes $m = \Theta(s, K_e)$ and checks if m has some redundancy information.

The notion of security for blind signature schemes was formally defined in [4] based on the random oracle model.

Definition 2 (the "one-more forgery"). For any fixed l, if a probabilistic polynomial time Turing machine A can compute, after l interactions with the signer, $l + 1$ signatures with non-negligible probability, then we say that it has performed an $(l, l + 1)$-forgery. A "one-more forgery" is an $(l, l + 1)$-forgery for some integer l. ❑

Definition 3 (Attacks). Two different attacks can be considered:

1. A sequential attack occurs when the attacker can sequentially interact with the signer.
2. A parallel attack occurs when the attacker can interact l times with the signer and send challenges whenever he wants.

Definition 4 A blind signature scheme $P = (M, S, \Delta, K, \Psi, \Re, \Omega, \partial, \Upsilon, \Psi, \Gamma)$ is unforgeable if no malicious adversary can perform a one-more forgery with non-negligible probability in the random oracle model under sequential or parallel attack. ❑

The notion of security for a blind (t, n) threshold signature scheme P_T can be formally defined as follows.

Definition 5 A blind (t, n) threshold signature scheme is unforgeable if no malicious adversary who corrupts at most $t - 1$ signers can perform a one-more forgery with an honest signer in the random oracle model with non-negligible probability under sequential or parallel attack. ❑

In order to prove unforgeability, we use the concept of the simulatable adversary view [13, 27, 28]. This means that adversary who sees all the information of the corrupted signers and the signature of m can generate by itself all the other information produced by the protocol except for the secret information generated by the honest signer. In other words, the run of the protocol provides no useful information to the adversary other than the final signature on m. According to [27, 28], we define below what the adversary sees as the view of the protocol.

Definition 6 Given a blind (t, n) threshold signature scheme $P_T = (M, S, \Delta, K, \Lambda, \Psi, \Re, \Omega_T, \partial_T, \Upsilon_T, \Phi_T, \Gamma)$, we define the view of an adversary who sees all the information of the $c < t$ corrupted signers $\Psi_c = \{U_{P_i} \mid 1 \le P_1, P_2, ..., P_c \le n\}$ on input m as the string $((\gamma_1, \gamma_2, \ldots, \gamma_c), (a_1, a_2, ..., a_j), (b_{P_1,1}, b_{P_1,2}, \ldots, b_{P_1,j}, b_{P_2,1}, b_{P_2,2}, \ldots, b_{P_2,j}, \ldots, b_{P_t,1}, b_{P_t,2}, \ldots, b_{P_tj}))$, where γ_i is the string of coin tosses of the corrupted signers U_{P_i} and $b_{P_i,k}$ (resp. a_k) is the message sent by U_{P_i} (resp. a requester $R \in \Re$) in the kth round of the protocol.

Indeed, one can prove that if the underlying signature scheme P of a simulatable threshold signature scheme P_T is unforgeable, then P_T is unforgeable [13, 27]. This predicate is equivalent to "if P_T is forgeable and P_T is simulatable, then P is forgeable" and can be simply proved by means of the construction method.

Definition 7 A blind (t, n) threshold signature scheme is simulatable if there exists a simulator *SIM* such that on input of the public key y, the public input m, the partial secret shadows provided by the $t - 1$ corrupted signers and the signature s of m, can simulate the view of the adversary on execution of the scheme that generates s as an output.

❑

3. THE PROPOSED SCHEME

In this section, we will propose a blind threshold signature scheme based on the Okamoto-Schnorr blind signature scheme [4]. In a typical signing process of a blind threshold signature scheme, there are two kinds of participants, signers and a requester. Before the requester can obtain a blind threshold signature from the signers, all the sign-

ers have to cooperate to distribute their secret shadows to other signers in advance. Then, the requester requests a blind threshold signature from the signers. The proposed scheme consists of three phases: (1) the shadow distribution phase, (2) the signature generation phase and (3) the signature verification phase. The shadow distribution phase is performed only once by the signers, and then they can use their secret shadows to sign messages. In the signature generation phase, a requester requests a blind threshold signature from the signers, and the signers cooperate to issue the blind threshold signature to the requester. In the signature verification phase, anyone can use the group public key to verify if a blind threshold signature is valid.

Let U_i be the identification of signer i, let n be the number of signers, let t be the threshold value of the blind threshold signature scheme, let m be the blind message to be signed, let H be a secure one-way hashing function [21-24], let p, q be two large prime numbers such that q divides $(p-1)$, and let ξ, ξ' be two generators of Z_p^*. Let $x \equiv_p y$ denote $x = y \bmod p$. Let $g \equiv_p \xi^{(p-1)/q}$ and $h \equiv_p \xi'^{(p-1)/q}$. Let d_i be the secret key chosen by U_i. In a distributed environment, U_i can publish the corresponding public key e_i. Anyone can get e_i via some authentication service (e.g., the X.509 directory authentication service [29]). Using a secure public key signature scheme [26, 30], U_i can produce signatures of messages using his own secret key d_i. Anyone can verify these signatures using the corresponding public key e_i. Let $C(m, \gamma)$ denote a commitment to $m \in Z_p^*$ using the random string γ, and let $Cert_{U_i}(H(c))$ denote the signature on $H(c)$ signed by U_i.

3.1 The Shadow Distribution Phase

Before a requester can request a blind threshold signature from the signers, all the signers must cooperate to distribute their secret shadows to other signers. In the shadow distribution phase, each U_i, $1 \le i \le n$, carries out the following steps:

1. U_i randomly chooses two secret keys r_i, $s_i \in Z_q$ 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 $a_{i,0} = r_i, a'_{i,0} = s_i$ and $a_{i,j}, a'_{i,j} \in Z_q$, $1 \le j \le t-1$; it computes $\Psi_{i,k} \equiv_p g^{-a_{i,k}}, \Psi'_{i,k} \equiv_p h^{-a'_{i,k}}, 0 \le k \le t-1$ and the signatures $Cert_{U_i}(H(\Psi_{i,k}))$ on $\Psi_{i,k}$, $Cert_{U_i}(H(\Psi'_{i,k}))$ on $\Psi'_{i,k}$, $1 \le k \le t-1$, the commitments $C_i = C(\Psi_{i,0}, \gamma_i)$, $C'_i = C(\Psi'_{i,0}, \gamma'_i)$ and the signatures $Cert_{U_i}(H(C_i))$ on C_i and $Cert_{U_i}(H(C'_i))$ on C'_i, and it sends $(Cert_{U_i}(H(C_i)), C_i, Cert_{U_i}(H(C'_i)), C'_i, (\Psi_{i,k}, \Psi'_{i,k}, Cert_{U_i}(H(\Psi_{i,k})), Cert_{U_i}(H(\Psi'_{i,k})), 1 \le k \le t-1))$ to U_j, $1 \le j \le n, j \ne i$.
2. Upon receiving $(Cert_{U_j}(H(C_j)), C_j, Cert_{U_j}(H(C'_j)), C'_j, (\Psi_{j,k}, \Psi'_{j,k}, Cert_{U_j}(H(\Psi_{j,k})), Cert_{U_j}(H(\Psi'_{j,k}))), 1 \le j \le n, j \ne i, 1 \le k \le t-1)$ from all other signers, U_i verifies whether all $Cert_{U_j}(H(C_j))$, $Cert_{U_j}(H(C'_j))$, $Cert_{U_j}(H(\Psi_{j,k}))$, and $Cert_{U_i}(H(\Psi'_{i,k}))$ are valid. If they are valid, he opens C_i, C'_i and sends both $\delta_{i,j} \equiv_q f_i(x_j)$, $\delta'_{i,j} \equiv_q f'_i(x_j)$, where x_j is a unique public number for U_j, and a signature $Cert_{U_i}(H(\delta_{i,j}))$ on $\delta_{i,j}$, $Cert_{U_i}(H(\delta'_{i,j}))$ on $\delta'_{i,j}$ secretly to every U_j, $1 \le j \le n, j \ne i$. Otherwise, he publishes the invalid signatures and stops.
3. When U_i receives all $\delta_{j,i}$, $\delta'_{j,i}$, $Cert_{U_j}(H(\delta_{j,i}))$, and $Cert_{U_j}(H(\delta'_{i,i}))$, $1 \le j \le n, j \ne i$, from other signers, he verifies whether the shares $\delta_{j,i}$, $\delta'_{j,i}$, received from U_j are consistent with the certified values $\Psi_{j,l}$, $\Psi'_{j,l}$, $0 \le l \le t-1$, by checking whether $g^{\delta_{j,i}} \equiv_p \prod_{l=0}^{t-1} (\Psi_{j,l})^{x_i^l}$ and $h^{\delta'_{j,i}} \equiv_p \prod_{l=0}^{t-1} (\Psi'_{j,l})^{x_i^l}$. If this fails, U_i broadcasts that

an error has been found, publishes $\delta_{j,i}$, $Cert_{U_j}(H(\delta_{j,i}))$ or $\delta'_{j,i}$, $Cert_{U_j}(H(\delta'_{j,i}))$ and the identification of U_j, and then stops. Otherwise, U_i computes the signature $Cert_{U_i}(H(y))$ on the group public key $y \equiv_p \prod_{l=1}^{n} y_l \equiv_p \prod_{l=1}^{n} \Psi_{l,0} \prod_{l=1}^{n} \Psi'_{l,0}$ and the signatures $Cert_{U_i}(H(\Phi_{j,i}))$ on $\Phi_{j,i} \equiv_p g^{\delta_{j,i}}$ and $Cert_{U_i}(H(\Phi'_{j,i}))$ on $\Phi'_{j,i} \equiv_p h^{\delta'_{j,i}}$, $1 \le j \le n$. He then sends $(Cert_{U_i}(H(y)), (\Phi_{j,i}, \Phi'_{j,i}, Cert_{U_i}(H(\Phi_{j,i}), Cert_{U_i}(H(\Phi'_{j,i})), 1 \le j \le n))$ to all other signers.

4. Upon receiving all $((Cert_{U_j}(H(y)), 1 \le j \le n, j \ne i)$, $(\Phi_{j,i}, \Phi'_{j,i}, Cert_{U_j}(H(\Phi_{j,i}), Cert_{U_j}(H(\Phi'_{j,i})), 1 \le l \le n, 1 \le j \le n, j \ne i))$, U_i verifies whether all $((Cert_{U_j}(H(y)), 1 \le j \le n, j \ne i)$, $Cert_{U_j}(H(\Phi_{j,i}), Cert_{U_j}(H(\Phi'_{j,i})), 1 \le l \le n, 1 \le j \le n, j \ne i))$ are valid. If they are, the shadow keys corresponding to the group secret keys $s \equiv_q \sum_{j=1}^{n} s_j$ and $r \equiv_q \sum_{j=1}^{n} r_j$ have been securely and correctly distributed. The group publi key $y \equiv_p \prod_{l=1}^{n} \Psi_{l,0} \Psi'_{l,0}$, all signers' public keys $\Psi_{l,0}$, $\Psi'_{l,0}$, $1 \le l \le n$, and all public shadows $\Phi_{l,j} \equiv_p g^{\delta_{l,j}}$, $\Phi'_{l,j} \equiv_p g^{\delta'_{l,j}}$, $1 \le l, j \le n$, can then be published by each signer. Otherwise, U_i publishes the invalid signatures and stops.

3.2 The Signature Generation Phase

Without loss of generality, we assume that t out of n signers are U_i, $1 \le i \le t$. The t signers perform the following steps during the signature generation phase.

1. Each U_i randomly chooses two random numbers t_i, $u_i \in Z_q$, computes $a_i \equiv_p g^{t_i} h^{u_i}$ and sends a_i to the requester.
2. After receiving all a_i, $1 \le i \le t$, the requester chooses three random numbers γ, β and $\delta \in Z_q$, computes $a \equiv_p \prod_{i=1}^{t} a_i, \alpha \equiv_p g^{\beta} h^{\gamma} y^{\delta} a, \varepsilon \equiv_p H(m, \alpha)$ and $e \equiv_q \varepsilon - \delta$, and sends e to all U_i, $1 \le i \le t$.
3. Upon receiving e, each U_i computes $R_i \equiv_q e(r_i + \sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\ne i}^{t}(\frac{-x_k}{x_i - x_k}))) + t_i$, $S_i \equiv_q e(s_i + \sum_{j=t+1}^{n} f'_j(x_i)(\prod_{k=1,k\ne i}^{t}(\frac{-x_k}{x_i - x_k}))) + u_i$ and sends S_i and R_i back to the requester.
4. After receiving all S_i and R_i, the requester checks if

$$g^{R_i} h^{S_i} y_i^e \equiv_p a_i ((\prod_{j=t+1}^{n} \Psi_{j,i})^{\prod_{k=1,k\ne i}^{t}(\frac{-x_k}{x_i - x_k})})^e ((\prod_{j=t+1}^{n} \Psi'_{j,i})^{\prod_{k=1,k\ne i}^{t}(\frac{-x_k}{x_i - x_k})})^e, 1 \le i \le t.$$

If any of S_i and R_i is not valid, he has to ask the corresponding signer to send it again. Otherwise, he computes $\rho \equiv_q \beta + \sum_{i=1}^{t} R_i, \sigma \equiv_q \gamma + \sum_{i=1}^{t} S_i$. The blind Threshold signature of m is (α, ρ, σ).

3.3 The Signature Verification Phase

To verify the blind threshold signature (α, ρ, σ) on message m, one simply checks if $\alpha \equiv_p g^{\rho} h^{\sigma} y^{\varepsilon}$.

4. DISCUSSION

We will discuss the correctness, security and performance of our blind threshold signature scheme in this section.

4.1 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.

Proof. By means of our scheme, we have

$$
\begin{aligned}
& g^{R_i} h^{s_i} y_i^e \\
\equiv_p\ & g^{e(r_i+\sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k})))+t_i} \\
& h^{e(S_i+\sum_{j=t+1}^{n} f'_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k})))+u_i}\ y_i^e \\
\equiv_p\ & g^{er_i} g^{e\sum_{i=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))} g^{t_i}\ h^{es_i} h^{e\sum_{j=t+1}^{n} f'_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))}\ h^{u_i} \\
& (g^{-r_i} h^{-s_i})^e \\
\equiv_p\ & g^{e\sum_{i=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))} g^{t_i}\ h^{e\sum_{j=t+1}^{n} f'_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))}\ h^{u_i} \\
\equiv_p\ & a_i((\prod_{j=t+1}^{n}\Psi_{j,i})^{\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k})})^e((\prod_{j=t+1}^{n}\Psi'_{j,i})^{\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k})})^e.
\end{aligned}
$$

❑

After the signature generation phase, the blind threshold signatures can be verified using the group public key in the signature verification phase. Let (m, (α, ρ, σ)) denote the message-signature pair generated in that execution. Theorem 2 ensures the correctness of the scheme.

Theorem 2. The 3-tuple (α, ρ, σ) is a valid blind threshold signature on message m.

Proof. The validity of the blind threshold signature (α, ρ, σ) on message m can easily be established as follows:

$$
\begin{aligned}
& g^{\rho} h^{\sigma} y^{\varepsilon} \\
\equiv_p\ & g^{\beta+\sum_{i=1}^{t} R_i} h^{\gamma+\sum_{i=1}^{t} S_i} y^{\varepsilon} \\
\equiv_p\ & g^{\beta+\sum_{i=1}^{t} R_i} h^{\gamma+\sum_{i=1}^{t} S_i} y^{e+\delta} \\
\equiv_p\ & g^{\beta} h^{\gamma} g^{\sum_{i=1}^{t} t_i + e\sum_{i=1}^{t} r_i} h^{\sum_{i=1}^{t} u_i + e\sum_{i=1}^{t} s_i} g^{e\sum_{i=1}^{t}\sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))} \\
& h^{e\sum_{i=1}^{t}\sum_{j=t+1}^{n} f'_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))} y^{e+\delta} \\
\equiv_p\ & a g^{\beta} h^{\gamma} g^{e(\sum_{i=1}^{t} r_i + \sum_{i=t+1}^{n} r_i)} h^{e(\sum_{i=1}^{t} s_i + \sum_{i=t+1}^{n} s_i)} y^{e+\delta} \\
\equiv_p\ & a g^{\beta} h^{\gamma} y^{-e} y^{e+\delta} \\
\equiv_p\ & a g^{\beta} h^{\gamma} y^{\delta} \\
\equiv_p\ & \alpha
\end{aligned}
$$

❑

4.2 Security Analysis

In the shadow distribution phase, since $\Psi_{i,0}$ and $\Psi'_{i,0}$ are committed using γ_i and γ'_i, after U_i receives all other commitments $C_j = C(\Psi_{i,0}, \gamma_j)$ and $C'_j = C(\Psi'_{i,0}, \gamma'_j)$, $1 \le j \le n$, $j \ne i$, then he opens the commitments. If U_i chooses his secret keys r_i and s_i at random, then the distributions of the group secret keys $s \equiv_q \sum_{j=1}^{n} s_j$ and $r \equiv_q \sum_{j=1}^{n} r_j$ are both polynomially indistinguishable from the uniform distribution. Given the secret information of a group of $l < t$ members, Lemma 3 ensures that the threshold cryptosystem constructed in the shadow distribution phase will not disclose any extra information about the group secret keys $s \equiv_q \sum_{j=1}^{n} s_j$ and $r \equiv_q \sum_{j=1}^{n} r_j$.

Lemma 3. Given a group of $\sigma < t$ members $G = \{p_i \mid p_i \in [1, n], 1 \le i \le \sigma\}$ and the set of shares $\{\delta_{j,i}, \delta'_{j,i} \mid 1 \le j \le n, i \in G\}$, for any fixed j, $1 \le j \le n$, it takes polynomial time on $|p|$ to generate two random sets $\{g^{\hat{a}_{j,k}} \mid 1 \le k \le t-1\}$ and $\{h^{\hat{a}'_{j,k}} \mid 1 \le k \le t-1\}$ satisfying $g^{\delta_{j,i}} \equiv_p \prod_{k=0}^{t-1} (g^{\hat{a}_{j,k}})^{x_i{}^k}$ and $h^{\delta'_{j,i}} \equiv_p \prod_{k=0}^{t-1} (h^{\hat{a}'_{j,k}})^{x_i{}^k}$ for $i \in G$.

Proof. In step 3 of the shadow distribution phase, after U_i has received all $\delta_{j,i}$, he verifies whether the share $\delta_{j,i}$ received from U_j is consistent with the certified values $\Psi_{j,l}$, $1 \le l \le t - 1$, by checking if $g^{\delta_{j,i}} \equiv_p \prod_{l=0}^{t-1} (\Psi_{j,l})^{x_i{}^l}$. Therefore,

$$g^{\delta_{j,i}} \equiv_p \prod_{l=0}^{t-1} (g^{a_{j,l}})^{x_i{}^l} \equiv_p g^{\sum_{l=0}^{t-1} a_{j,l} * x_i{}^l}. \tag{1}$$

Since $g \equiv_p \xi^{(p-1)/q}$ and ξ is a generator of Z_p^*, g generates a cyclic subgroup S_q of Z_p^* with $|S_q| = q$. From (1), we have

$$\delta_{j,i} \equiv_q \sum_{l=0}^{t-1} a_{j,l} * x_i^{\,l}. \tag{2}$$

From (2), we know that given a fixed index j, the shares $\delta_{j,i}$, $i \in G$, will use the same variables $\hat{a}_{j,k}$, $0 \le k \le t-1$, as follows:

$$\delta_{j,i} \equiv_q \sum_{k=0}^{t-1} \hat{a}_{j,k} * x_i^{\,k}. \tag{3}$$

Given a fixed index j, we can get at most σ linear equations with t variables as follows:

$$\delta_{j,i} \equiv_q \sum_{k=0}^{t-1} \hat{a}_{j,k} * x_i^{\,k} (i \in G). \tag{4}$$

Since the linear equations have at least one solution $\hat{a}_{j,k} = a_{j,k}$, $0 \le k \le t-1$, we can solve linear equations (4) and get a random solution $\hat{a}_{j,k}$, $1 \le k \le t-1$, by assigning random values to all free variables. From (4), it is clear that $g^{\delta_{j,i}} \equiv_p g^{\sum_{k=0}^{t-1} \hat{a}_{j,k} * x_i^{\,k}} \equiv_p \prod_{k=0}^{t-1} (g^{\hat{a}_{j,k}})^{x_i^{\,k}}$. ❑

Similar to the above proof, we can get a random solution $\hat{a}'_{j,k}$, $0 \le k \le t-1$, such that $h^{\delta'_{j,i}} \equiv_p h^{\sum_{k=0}^{t-1} a'_{j,k} * x_i^{\,k}} \equiv_p \prod_{k=0}^{t-1} (h^{a'_{j,k}})^{x_i^{\,k}}$. ❑

Let ν denote the signers' complete views of an execution in the signature generation phase, and let $(m, (\alpha, \rho, \sigma))$ denote the message-signature pair generated in that execution. Theorem 4 ensures the blindness of our proposed scheme.

Theorem 4. The threshold signature scheme proposed in Section 4 is blind.

Proof. To prove the blindness of the scheme, we will show that given any view ν and any valid message-signature pair $(m, (\alpha, \rho, \sigma))$, there exists a unique trio of blinding factors β, γ, and δ. Since the requester chooses the blinding factors β, γ, and δ randomly, the blindness of the signature scheme follows.

Given a valid message-signature pair $(m, (\alpha, \rho, \sigma))$, and a view ν, the following equations must hold for β, γ, and δ. Without loss of generality, assume that the blind signature (α, ρ, σ) has been generated by t signers U_i, $1 \le i \le t$, with the view ν consisting of $R_i \equiv_q e(r_i + \sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t} (\frac{-x_k}{x_i - x_k}))) + t_i, S_i \equiv_q e(s_i + \sum_{j=t+1}^{n} f'_j(x_i)(\prod_{k=1,k\neq i}^{t} (\frac{-x_k}{x_i - x_k}))) + u_i, t_i$ and u_i, $1 \le i \le t$ and e:

$$\alpha \equiv_p g^{\beta} h^{\gamma} y^{\delta} a, \tag{5}$$

$$\rho \equiv_q \beta + \sum_{i=1}^{t} R_i, \tag{6}$$

$$\sigma \equiv_q \gamma + \sum_{i=1}^{t} S_i, \tag{7}$$

$$\varepsilon \equiv_q e + \delta. \tag{8}$$

By equation (5), (6), (7) and (8), the unique solution for β, γ and δ is

$$\beta \equiv_q \rho - \sum_{i=1}^{t} R_i, \tag{9}$$

$$\gamma \equiv_q \sigma - \sum_{i=1}^{t} S_i, \tag{10}$$

$$\delta \equiv_q \varepsilon - e. \tag{11}$$

In the following, we will show that the solutions of γ, δ and β in equations (9), (10) and (11) also satisfy equation (5):

$$\begin{aligned}
& g^{\beta} h^{\gamma} y^{\delta} a \\
\equiv_p\ & g^{\rho-\sum_{i=1}^{t} R_i} h^{\sigma-\sum_{i=1}^{t} S_i} y^{\varepsilon-e} \prod_{i=1}^{t} a_i \\
\equiv_p\ & g^{\rho-\sum_{i=1}^{t} R_i} h^{\sigma-\sum_{i=1}^{t} S_i} y^{\varepsilon-e} \prod_{i=1}^{t} g^{t_i} h^{u_i} \\
\equiv_p\ & g^{\rho} g^{-\sum_{i=1}^{t} R_i} h^{\sigma} h^{-\sum_{i=1}^{t} S_i} y^{-e} \prod_{i=1}^{t} g^{t_i} h^{u_i} \\
\equiv_p\ & g^{\rho} h^{\sigma} y^{\varepsilon} y^{-e} g^{\sum_{i=1}^{t} t_i} h^{\sum_{i=1}^{t} (e(r_i + \sum_{j=t+1}^{n} f_j(x_i)} {}^{(\prod_{k=1,k\neq i}^{t} (\frac{-x_k}{x_i - x_k})) + t_i)} \\
& h^{-\sum_{i=1}^{t} (e(s_i + \sum_{j=t+1}^{n} f_j'(x_i)(\prod_{k=1,k\neq 1}^{t} (\frac{-x_k}{x_i - x_k}))) + u_i)} \\
\equiv_p\ & g^{\rho} h^{\sigma} y^{\varepsilon} y^{-e} g^{\sum_{i=1}^{t} t_i} h^{\sum_{i=1}^{t} u_i} g^{-\sum_{i=1}^{t} t_i} h^{-\sum_{i=1}^{t} u_i} g^{-e\sum_{i=1}^{t} (r_i + \sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t} (\frac{-x_k}{x_i - x_k})))} \\
& h^{-e\sum_{i=1}^{t} (s_i + \sum_{j=t+1}^{n} f_j'(x_i)(\prod_{k=1,k\neq 1}^{t} (\frac{-x_k}{x_i - x_k})))} \\
\equiv_p\ & g^{\rho} h^{\sigma} y^{\varepsilon} y^{-e} (g^{\sum_{i=1}^{n} r_i})^{-e} (h^{\sum_{i=1}^{n} s_i})^{-e} \\
\equiv_p\ & g^{\rho} h^{\sigma} y^{\varepsilon} y^{-e} (g^{-\sum_{i=1}^{n} -r_i})^{-e} (h^{-\sum_{i=1}^{n} -s_i})^{-e} \\
\equiv_p\ & g^{\rho} h^{\sigma} y^{\varepsilon} y^{-e} y^{e} \\
\equiv_p\ & g^{\rho} h^{\sigma} y^{\varepsilon} \\
\equiv_p\ & \alpha.
\end{aligned}$$

❑

Our proposed blind threshold signature scheme is based on a provably secure blind signature scheme under the random oracle model [4].

Theorem 5. Consider the Okamato-Schnorr blind signature scheme in the random oracle model. A "one-more forgery," even under parallel attack, is equivalent to the discrete logarithm problem in a subgroup. [4] ❑

Since the Okamato-Schnorr blind signature scheme is unforgeable in the random oracle model, if our proposed blind threshold signature scheme is simulatable, our proposed scheme is unforgeable.

Let *Threshold_gen* denote the protocol in the signature generation phase. Without loss of generality, we assume that the adversary has corrupted $t-1$ signers U_i, $1 \le i \le t-1$, and the requester with the view consisting of $m, y, (r_i, s_i, 1 \le i \le t-1), (\delta_{i,j}, \delta'_{i,j}, 1 \le i \le t-1, 1 \le j \le n)$. To prove the unforgeability of our proposed scheme, we will now construct a simulator *SIM* as follows. The simulator *SIM* is described as a two-phase protocol. The first phase computes all the necessary information, and the second phase carries out communication with the adversary in accordance with *Threshold_gen*.

Simulator SIM
SIM_Computation $(m, y, (r_i, s_i, 1 \le i \le t-1), (\delta_{i,j}, \delta'_{i,j}, 1 \le i \le t-1, 1 \le j \le n), (\alpha, \rho, \sigma))$:

1. Randomly choose $\tilde{t}_i$ and $\tilde{u}_i \in Z_q$, $1 \le i \le t-1$.
2. Randomly choose $\tilde{\gamma}, \tilde{\beta}$ and $\tilde{\delta} \in Z_q$ and compute $\tilde{e} \equiv_q \varepsilon - \tilde{\delta}$.
3. Compute $\tilde{R}_i \equiv_q \tilde{e}(r_i + \sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))) + \tilde{t}_i$, $1 \le i \le t-1$.
4. Compute $\tilde{S}_i \equiv_q \tilde{e}(s_i + \sum_{j=t+1}^{n} f'_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))) + \tilde{u}_i$, $1 \le i \le t-1$.
5. Compute $\tilde{R}_t \equiv_q \rho - \tilde{\beta} - \sum_{i=1}^{t-1}\tilde{R}_i$.
6. Compute $\tilde{S}_t \equiv_q \sigma - \tilde{\gamma} - \sum_{i=1}^{t-1}\tilde{S}_i$.

end of SIM_Computation.

SIM_Conversation

Comment: In each of the following steps, we describe the information which *SIM* gives to the adversary. Each of these steps corresponds to the same numbered step in protocol *Threshold_gen*:

1. The $2(t-1)$ random numbers $\tilde{t}_i$ and $\tilde{u}_i \in Z_q$, $1 \le i \le t-1$.
2. The three blinding factors $\tilde{\gamma}, \tilde{\delta}$ and $\tilde{\beta}$ and the blind message $\tilde{e}$.
3. The $2t$ blind partial signatures $\tilde{R}_i \equiv_q \tilde{e}(r_i + \sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))) + \tilde{t}_i$, $1 \le i \le t-1$, $\tilde{R}_t \equiv_q \rho - \tilde{\beta} - \sum_{i=1}^{t-1}\tilde{R}_i, \tilde{S}_i \equiv_q \tilde{e}(s_i + \sum_{j=t+1}^{n} f'_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))) + \tilde{u}_i$, $1 \le i \le t-1$ and $\tilde{S}_t \equiv_q \sigma - \tilde{\gamma} - \sum_{i=1}^{t-1}\tilde{S}_i$.

4. Do nothing.

end of SIM_Conversation.

end of SIM.

Let $View_A$(*Threshold_gen*($m, y,(r_i, s_i, 1 \le i \le t-1), (\delta_{i,j}, \delta'_{i,j}, 1 \le i \le t-1, 1 \le j \le n), (\alpha, \rho, \sigma)$) be all the information of the corrupted signers and the requester in the signature generation phase, and let *SIM*($m, y,(r_i, s_i, 1 \le i \le t-1), (\delta_{i,j}, \delta'_{i,j}, 1 \le i \le t-1, 1 \le j \le n), (\alpha, \rho, \sigma)$) be the information constructed by the simulator *SIM* with ($m, y, (r_i, s_i, 1 \le i \le t-1), (\delta_{i,j}, \delta'_{i,j}, 1 \le i \le t-1, 1 \le j \le n), (\alpha, \rho, \sigma)$) as input. Theorem 6 ensures that *Threshold_gen* in Section 3.2 is simulatable.

Theorem 6. $View_A$(*Threshold_gen*($m, y,(r_i, s_i, 1 \le i \le t-1), (\delta_{i,j}, \delta'_{i,j}, 1 \le i \le t-1, 1 \le j \le n), (\alpha, \rho, \sigma)$)) is computationally indistinguishable from *SIM* ($m, y,(r_i, s_i, 1 \le i \le t-1), (\delta_{i,j}, \delta'_{i,j}, 1 \le i \le t-1, 1 \le j \le n), (\alpha, \rho, \sigma)$).

Proof. We shall analyze the information generated by *Threshold_gen* and *SIM* in each step.

1. Both *Threshold_gen* and *SIM* choose $2(t-1)$ random numbers. Thus, the same probability distribution is generated for sets of size $2(t-1)$.
2. *Threshold_gen* randomly chooses three blinding factors, γ, β and $\delta \in Z_q$, and *SIM* also randomly chooses three blinding factors, $\tilde{\gamma}, \tilde{\beta}$ and $\tilde{\delta} \in Z_q$. These three probability distributions are the same. *Threshold_gen* computes the blind message $e \equiv_q \varepsilon - \delta$, and *SIM* computes the blind message $\tilde{e} \equiv_q \varepsilon - \tilde{\delta}$. These two blind messages are both blinded with random blind factor δ or $\tilde{\delta}$. Therefore, these two probability distributions are the same.
3. *Threshold_gen* generates t blind partial signatures $R_i \equiv_q e(r_i + \sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))) + t_i$, $1 \le i \le t$, which consist of the blind message e, the partial secrets $r_i + \sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))$, $1 \le i \le t$, and the random numbers t_i, $1 \le i \le t$. *SIM* also generates t blind partial signatures $\tilde{R}_i \equiv_q \tilde{e}(r_i + \sum_{j=t+1}^{n} f_j(x_i)\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))) + t_i$, $1 \le i \le t-1$, which consist of the blind message $\tilde{e}$, the partial secrets $r_i + \sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))$, $1 \le i \le t-1$, and the random numbers $\tilde{t}_i$, $1 \le i \le t-1$, and $\tilde{R}_t \equiv_q \rho - \tilde{\beta} - \sum_{i=1}^{t-1}\tilde{R}_i$. Since the blind messages $\tilde{e}$ and e are in the same probability distribution, the partial signatures R_i and R_i, $1 \le i \le t-1$, are in the same probability distribution. In step 3, we can know that R_t and $\tilde{R}_t$, are in the same probability distribution since β and $\tilde{\beta}$ are in the same probability distribution. Similarly, we can show that the partial signatures S_i and $\tilde{S}_i$, $1 \le i \le t$, are in the same probability distribution.

This completes the proof of Theorem 6. ❑

Since the underlying blind signature scheme is unforgeable and our proposed threshold signature scheme is simulatable, the security problem in the proposed threshold

signature scheme is equivalent to the discrete logarithm problem in the random oracle model.

4.3 Performance Analysis

In this subsection, we will analyze the computational cost required to compute blind (t, n) threshold signatures using our scheme. We will use as a measure the number of modular exponentiations and that of modular inverses required by a single player during execution of our signature generation protocol. Table 1 shows a comparison between the blind threshold signature scheme and its underlying blind signature scheme. In this table, Scheme 1 denotes the blind threshold signature scheme described in Section 4, and Scheme 1^* denotes its corresponding underlying blind signature scheme. To reduce the computational cost due to each signer, the value $-x_k/(x_i - x_k)$, $1 \le k \le n$ and $k \ne i$, in Step 3 of the signature generation phase can be computed off-line. In this case, each signer needs to compute only 2 modular exponentiations in our scheme, which is the same as in the underlying blind signature scheme. Compared with the underlying blind signature scheme, the extra cost for signing a blind threshold signature is determined by computing $\sum_{j=t+1}^{n} f_j(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))$ and $\sum_{j=t+1}^{n} f_j^{'}(x_i)(\prod_{k=1,k\neq i}^{t}(\frac{-x_k}{x_i-x_k}))$ in Step 3, which contains $2(n-2)$ modular multiplications and $2(n-t)$ additions. To reduce the computational cost due to the requester, the partial signature verification task in step 4 is not done except when the final threshold signature can not satisfy the verification equation in the signature verification phase. In this approach, the requester only needs to perform 3 modular exponentiations in Step 2 of the signature generation phase, which is the same as in the underlying blind signature scheme. Since the blind threshold verification function of our scheme is the same as that of the underlying blind signature scheme, the verification cost of our blind threshold signature is the same as that of the underlying blind signature. Compared with the underlying blind signature scheme, the extra cost of requesting a blind threshold signature in our scheme proposed in Section 4 is incurred in computing $\prod_{i=1}^{t} a_i$, $\sum_{i=1}^{t} R_i$ and $\sum_{i=1}^{t} S_i$, which contains $t-1$ modular multiplications and $2(t-1)$ modular additions. In our scheme, the size of the threshold signature is the same as that of an individual signature, and the verification process for a threshold signature is equivalent to that for an individual signature.

In [13], three robust threshold signature protocols, namely, DSS-Thresh-Sig-1, DSS-Thresh-Sig-2 and DSS-Thresh-Sig-3, were proposed. One approach generates blind threshold signatures by taking take robust threshold signature schemes [13] and turning them into 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 the situation is even worse for DSS-Thresh-Sig-2 and DSS-Thresh-Sig-3, which require $O(nt)$ modular exponentiations. It is clear that this approach is quite inefficient compared to our proposed scheme.

Table 1. Cost of the signature generation phase in the blind threshold signature scheme and that in the underlying blind signature scheme.

	The requester				The signer or U_i			
	EXP	INV	MUL	ADD	EXP	INV	MUL	ADD
Scheme 1	3	0	$t + 2$	$2t + 1$	2	0	$2n - 1$	$2(n - t + 1)$
Scheme 1*	3	0	3	3	2	0	3	2

where
EXP = the number of modulo exponentiations,
INV = the number of modulo inversions (divisions),
MUL = the number of modulo multiplications,
ADD = the number of modulo additions.

5. CONCLUSIONS

We have proposed an efficient and provably secure blind threshold signature scheme based on the discrete logarithm problem. In our scheme, the size of a blind threshold signature is the same as that of an individual blind signature, and the signature verification process is equivalent to that for an individual signature. Ours is the first scheme whose security problem has been proved to be equivalent to the discrete logarithm problem in the random oracle model. Our proposed scheme can be easily applied to current efficient single-authority e-cash systems or secure voting systems for distributing the power of a single authority without changing the underlying structure or degrading the overall performance.

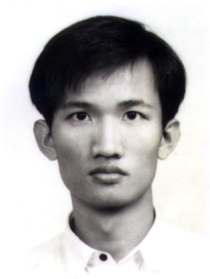
REFERENCES

1. J. Camenisch, J. Pivereau, and M. Stadler, "Blind signatures based on the discrete logarithm problem," in A. D. Santis (ed.), *Advances in Cryptology, EuroCrypt'94*, LNCS 950, Springer-Verlag, 1995, pp. 428-432.
2. D. Chaum, "Blind signatures for untraceable payments," in D. Chaum, R. L. Rivest, and A. T. Sherman (ed.), *Advances in Cryptology, Crypt '82*, 1983, pp. 199-203.
3. 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, AisaCrypt'94*, LNCS 917, Springer-Verlag, 1994, pp. 224-237.
4. D. Pointcheval and J. Stern, "Provably secure blind signature schemes," Advances in Cryptology, AisaCrypt '96, LNCS 1163, Springer-Verlag, 1996, pp. 252-265.
5. D. Pointcheval and J. Stern, "New blind signatures equivalent to factorization," in *Proceedings of the 4th ACM Conference on Computer and Communications Security*, 1997, pp. 92-99.
6. D. Chaum, "Privacy protected payments: unconditional payer and/or payee untraceability," in *Smartcard 2000*, North Holland, 1988, pp. 69-92.
7. N. Ferguson, "Single term off-line coins," in T. Helleseth (ed.), *Advances in Cryptology, EuroCrypt '93*, LNCS 765, Springer-Verlag, 1993, pp. 318-328.
8. 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.
9. W. Juang and C. Lei, "A secure and practical electronic voting scheme for real world environments," *IEICE Transactions on Fundamentals*, Vol. E80-A, 1997, pp. 64-71.
10. K. Sako, "Electronic voting scheme allowing open objection to the tally," *IEICE Transactions on Fundamentals*, Vol. E77-A, 1994, pp. 24-30.
11. L. Harn, Cryptanalysis of the blind signatures based on the discrete logarithm problem," *Electronics Letters*, Vol. 31, 1995, pp. 1136-1136.
12. P. Horster, M. Michels, and H. Petersen, "Comment on cryptanalysis of the blind signatures based on the discrete logarithm problem," *Electronics Letters*, Vol. 31, 1995, pp.1827-1827.
13. R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, "Robust threshold DSS signatures," in U. Maurer (ed.), *Advances in Cryptology, EuroCrypt* '96, LNCS 1070, Springer Verlag, 1996, pp. 354-371.
14. L. Harn, "Group-oriented (t, n) threshold digital signature scheme and digital multisignature," *IEE Proceeding on Computer Digital Techniques*, Vol. 141, 1994, pp. 307- 313.
15. W. Juang and C. Lei, "Blind threshold signatures based on discrete logarithm," in *Proceedings of Second Asian Computing Science Conference on Programming, Concurrency and Parallelism, Networking and Security*, LNCS 1179, Springer-Verlag, 1996, pp. 172 -181.
16. W. Juang and C. Lei, "Partially blind threshold signatures based on discrete logarithm," *Computer Communications*, Vol. 22, 1999, pp. 73-86.
17. D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, Vol. 24, 1981, pp. 84-88.
18. D. Chaum. "The dining cryptographers problem: unconditional sender and recipient untraceability," *Journal of Cryptology*, Vol. 1, 1988, pp. 65-75.
19. W. Juang, C. Lei, and C. Chang, "Anonymous channel and authentication in wireless communications," *Computer Communications*, Vol. 22, 1999, pp. 1502-1511.
20. T. Okamoto, "A digital multisignature scheme using bijective public-key cryptosystems, " *ACM Transactions on Computer Systems*, Vol. 6, 1988, pp. 432-441.
21. R. C. Merkle, "One way hash functions and DES," in G. Brassard (ed.), *Advances in Cryptology, Crypt '89*, LNCS 435, Springer-Verlag, 1990, pp. 428-446.
22. NIST FIPS PUB 180, "Secure hash standard," National Institute of Standards and Technology, U. S. Department of Commerce, DRAFT, 1993.
23. S. Pohlig and M. E. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance," *IEEE Transactions on Information Theory*, Vol. IT-24, 1978, pp. 106-110.
24. R. L. Rivest, "The MD5 message-digest algorithm," RFC 1321, Internet Activities Board, Internet Privacy Task Force, 1992.
25. K. Nyberg and R. A. Rueppel, "Message recovery for signature schemes based on the discrete logarithm problem," in A. D. Santis (ed.), *Advances in Cryptology, EuroCrypt '94*, LNCS 950, Springer-Verlag, 1995, pp. 182-193.
26. R. L. Rivest, A. Shamir, and L. Adelman, "A method for obtaining digital signatures and public key cryptosystem," *Communications of the ACM*, Vol. 21, 1978, pp. 120-126.

27. S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof-systems," *SIAM Journal on Computing*, Vol. 18, 1989, pp. 186-208.
28. S. Micali and P. Rogaway, "Secure computation," in J. Feigenbaum (ed.), *Advances in Cryptology, Crypt '91*, LNCS 576, Springer-Verlag, 1992, pp. 392-404.
29. W. Stallings, *Network and Internetwork Security*, Prentice Hall International, 1995.
30. T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithm," *IEEE Transactions on Information Theory*, Vol. IT-31, 1985, pp. 469-472.

Chin-Laung Lei (雷欽隆) was born in Taipei, Taiwan in 1958. He received his B.S. degree in electrical engineering from National Taiwan University in 1980 and his Ph.D. degree in computer science from the University of Texas in 1986. From 1986 to 1988 he was an assistant professor of computer and information science at the Ohio State University. In 1988, he joined the department of electrical engineering, Naitonal Taiwan University, where he is now a professor. His current research interests include network security, cryptography, design and analysis of algorithms, and operating system design. Dr. Lei is a member of the Institute of Electrical and Electronic Engineers, and the Association for Computing Machinery.



Wen-Shenq Juang (莊文勝) was born in Taichung, Taiwan in 1969. He received his B.S. degree in Computer Science and Information Engineering from Tatung Institute of Technology in 1991, his M.S. degree in Computer Information Science from National Chiao Tung University in 1993, and his Ph.D. degree in electrical engineering from National Taiwan University in 1998. He is now an assistant professor of information management at Shih Hsin University. His current research interests include information security, Internet technology, and electronic commerce. Dr. Juang is a member of Chinese Cryptology and Information Security Association.

Pei-Ling Yu (尤焙麟) was born in Taichung, Taiwan in 1973. He received his B.S. degree in Department of Computer and Information Science from National Chiao Tung University, Taiwan, in 1996. He is now a Ph.D. candidate of electrical engineering at National Taiwan Unviersity. His current research interests include information security and electronic payment. He is also a member of Chinese Cryptology and Information Security Association.