SCIS 2003 The 2003 Symposium on Cryptography and Information Security
Hamamatsu,Japan, Jan.26-29,2003
The Institute of Electronics,
Information and Communication Engineers

A New Threshold Blind Signature Scheme from Pairings

Duc Liem Vo * vdliem@icu.ac.kr
Fangguo Zhang * zhfg@icu.ac.kr
Kwangjo Kim * kkj@icu.ac.kr

Abstract— Threshold digital signature and blind signature are playing important roles in cryptography as well as in practical applications such as e-cash and e-voting systems, etc. In this paper, we present a new threshold blind digital signature based on pairings without the third party. Our scheme operates on Gap Deffie-Hellman (GDH) group, where Computational Diffie-Hellman problem is hard but Decision Diffie-Hellman problem is easy. For example, we use pairings that could be built from Weil pairing or Tate pairing. We also analyze security and efficiency of the scheme.

Keywords: Threshold signature, Blind signature, VSS, Bilinear pairings.

1 Introduction

Digital signature is an essential component in cryptography. Depending on its application purpose, the digital signatures can provide the required cryptographic properties.

A threshold signature scheme distributes the signing abilities to a group of signers such that a digital signature on a message cannot be produced by predetermined numbers of signers. A blind signature scheme, on the other hand, gives users ability to get a digital signature from a signer without revealing message content. This property is very important for implementing e-voting, e-commerce, and e-payment systems, etc. When a buyer purchases merchandize from a shop, the buyer gets a bank's signature on the payment given to the shop and keeps secret what merchandize is from the bank.

A threshold blind signature scheme combines a threshold signature scheme and a blind one to take both their properties. Therefore, a threshold blind signature while giving user ability to get signature on a message without revealing its content, still maintains the secret key to be distributed among signers.

In this paper, we propose a threshold blind signature scheme based on pairings. The rest of the paper is organized as follow. Some background on bilinear pairings and relevant tools that we use in our proposed scheme are introduced in Section 2. In Section 3, we describe our proposed threshold blind signature scheme. Section 4 analyzes the security aspects of the proposed scheme. In Section 5, we will evaluate performance of our scheme and compare with other schemes as well. Section 6 will finalize our work.

* International Research center for Information Security (IRIS), Information and Communications Univ. (ICU), 58-4 Hwaam-dong, Yuseong-gu, Daejon, 305-732, Korea.

2 Background and related work

2.1 Concepts of bilinear pairings

We summarize some concepts of bilinear pairings using similar notations in [16].

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be additive and multiplicative groups of the same prime order q, respectively. Let P is a generator of $\mathbb{G}_1$. Assume that the discrete logarithm problems in both $\mathbb{G}_1$ and $\mathbb{G}_2$ are hard. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a pairing which satisfies the following properties:

1. *Bilinear*: $\hat{e}(aP, bP') = \hat{e}(P, P')^{ab}$ for all $P, P' \in \mathbb{G}_1$ and all $a, b \in \mathbb{Z}$.
2. *Non-degenerate*: If $\hat{e}(P, P') = 1 \; \forall \, P' \in \mathbb{G}_1$ then $P = \mathcal{O}$.
3. *Computable*: There is an efficient algorithm to compute $\hat{e}(P, P')$ for any $P, P' \in \mathbb{G}_1$.

To construct the bilinear pairing, we can use the Weil pairing and Tate pairing associated with supersingular elliptic curves.

With such group $\mathbb{G}_1$, we can define the following hard cryptographic problems:

- **Discrete Logarithm (DL) Problem:** Given $P, P' \in \mathbb{G}_1$, find an integer n such that $P = nP'$ whenever such integer exists.
- **Computational Diffie-Hellman (CDH) Problem:** Given a triple $(P, aP, bP) \in \mathbb{G}_1$ for $a, b \in \mathbb{Z}_q^*$, find the element abP.
- **Decision Diffie-Hellman (DDH) Problem:** Given a quadruple $(P, \; aP, \; bP, \; cP) \in \mathbb{G}_1$ for $a, b, c \in \mathbb{Z}_q^*$, decide whether $c = ab \pmod q$ or not.
- **Gap Diffie-Hellman (GDH) Problem:** A class of problems where the CDH problem is hard but DDH problem is easy.

Groups where the CDH problem is hard but the DDH problem is easy are called Gap Diffie-Hellman (*GDH*) groups. Details about GDH groups can be found in [2], [3], [8].

2.2 Blind signature scheme based on GDH problem

The blind digital signature was first introduced by Chaum in [5] and becomes essential tools for e-cash. After Chaum suggested a construction method of a blind signature scheme based on RSA problem, there are many researches [1], [13], [14] dealing with blind digital signatures as well as their security. Recently, Boldyreva in [4] introduced a blind digital signature scheme based on Gap Diffie-Hellman problem and proved its security. For our threshold blind signature scheme, we construct a new blind signature which is defined as follows:

Let $\mathbb{G}_1$ be GDH group of prime order q. Public information is $I = (q, P, H)$ where P is generator of $\mathbb{G}_1$ and $H : \{0,1\}^* \to \mathbb{G}_1$ is an one-way hash function. The new blind GDH signature scheme $\mathsf{BGS}[\mathbb{G}_1] = (\mathcal{BK}, \mathcal{BS}, \mathcal{BV})$, where $\mathcal{BK}, \mathcal{BS}$ and $\mathcal{BV}$ are key generation, blind signing and verification algorithms respectively, is defined as:

- $\mathcal{BK}(I)$: Pick randomly $s \xleftarrow{R} Z_q^*$ and compute $Q \leftarrow sP$. Return $(\mathtt{pk} = (q, P, H, Q), \mathtt{sk} = s)$, where $\mathtt{pk}$ and $\mathtt{sk}$ are public key and secret key respectively.
- $\mathcal{BS}(I, \mathtt{sk}, M)$: The user wants a message $M \in \{0,1\}^*$ to be signed "blindly". After picking random number $r \xleftarrow{R} Z_q^*$, he computes $M' = r \cdot H(M)$ and sends it to the signer. The signer computes $\sigma' = s \cdot M'$ and sends it to the user. The user then computes the signature $\sigma = r^{-1} \cdot \sigma'$ and outputs (M, σ).
- $\mathcal{BV}(\mathtt{pk}, M, \sigma)$: If $\mathcal{V}_{\mathcal{DDH}}(P, Q, H(M), \sigma) = 1$ then return 1 else return 0, where $\mathcal{V}_{\mathcal{DDH}}()$ is an efficient algorithm which solves the DDH problem in $\mathbb{G}_1$.

2.3 Threshold cryptosystem

The concept of a threshold scheme was first introduced by Shamir [15]. In the (t, n)-threshold scheme, a secret D is divided into n pieces $D_1, D_2, \ldots, D_n$ such that:

1. Knowledge of any t or more D_i pieces makes D easily computable;
2. Knowledge of any $t-1$ or fewer D_i pieces leaves D uncomputable.

As mentioned above, the threshold scheme enables possession of secret key to be distributed in public key cryptosystem. Consequently, only t parties or more can decrypt a ciphertext encrypted with the corresponding public key or produce a digital signature on a message. With fewer t parties, the work cannot be done. Many relevant researches were found in [6], [9], [10], [11], [12].

However, the threshold scheme proposed by Shamir requires a dealer to distribute shared secrets to parties. Pedersen [12] proposed a threshold cryptosystem without a trusted party. In this scheme, each party acts as a dealer to choose the secret key and distribute it verifiably to other parties. Subsequently, a group of honest parties is formed and the group members recover their secret share.

In the next section, we will describe our proposed digital signature scheme making use of the threshold scheme and the blind signature scheme described before.

3 Proposed Scheme

Our threshold blind signature scheme contains three protocols: Key Generation protocol $\mathcal{TBK}$, Signature Generation protocol $\mathcal{TBS}$ and Signature Verification protocol $\mathcal{TBV}$.

3.1 Key Generation Protocol

The Key Generation protocol makes use of Verifiable Secret Sharing proposed by Pedersen [12], where n players are involved in this protocol to make a (t, n)-threshold scheme under $n \geq 2t - 1$.

Let $\mathbb{G}_1$ be GDH group and P is a generator of $\mathbb{G}_1$. Denote n players involved in Key Generation protocol to be $\{L_1, L_2, \ldots, L_n\}$. The public key and the secret key of this group of players are Q and s, respectively. The public share of the player L_i is Q_i and the corresponding secret share is s_i.

Each player L_i behaves as the following to generate a shared secret.

- At first, L_i sends its information.

G1. Selects randomly (uniformly distributed as in [11]) $a_{i0} \in \mathbb{Z}_q^*$, keeps it secret and broadcasts $a_{i0}P$.

G2. Picks up randomly a polynomial $f_i(x)$ over $\mathbb{Z}_q$ of degree at most $t-1$ such that $f_i(0) = a_{i0}$. Let

$$f_i(x) = a_{i0} + a_{i1}x + \cdots + a_{i,t-1}x^{t-1}$$

G3. Computes and broadcasts $a_{ij}P$ for $j = 1, 2, \ldots, t-1$; sends $f_i(j)$ *secretly* to each player L_j for $j = 1, 2, \ldots, n$; $j \neq i$.

- L_i receives information from other players.

G4. After receiving $f_j(i)$ from L_j for $j = 1, 2, \ldots, n$; $j \neq i$, the player L_i verifies $f_j(i)$ by checking

$$f_j(i)P = \sum_{k=0}^{t-1} i^k \cdot a_{jk}P$$

If the check fails, L_i broadcasts a complaint against L_j. Assume that none of players have a complaint.

G5. Computes the secret share $s_i = \sum_{k=1}^{n} f_k(i)$, the public share $Q_i = s_iP$ and the public key $Q = \sum_{i=1}^{n} a_{i0}P$.

After execution of the Key Generation protocol, the public key is $Q = sP$. The secret key $s = \sum_{i=1}^{n} a_{i0}$ is distributed to n players but does not appear explicitly in the protocol.

3.2 Signature Generation Protocol

Let M be a message to be signed, and $H : \{0,1\}^* \rightarrow \mathbb{G}_1$ be an one-way hash function. The public key output from the Key Generation protocol is $Q = sP$, where s is the implicit secret key constructed by n signers via the threshold scheme. Suppose that a user A wants to get a signature on the message M blindly from t signers. Denote t signers by $S = \{L_i | 1 \leq i \leq t\}$.

S1. User A chooses randomly (uniformly distributed) $r \in \mathbb{Z}_q^*$ and blinds the message M by computing $M' = rH(M)$. A sends M' along with $\omega_i = \prod_{\substack{j \in S \\ j \neq i}} \frac{j}{j-i}$ to every signer L_i for $i \in S$.

S2. Signer L_i, after receiving M', computes a partial signature σ_i and sends it back to the user, where

$$\sigma_i = s_i \omega_i M'$$

S3. User A, after receiving σ_i, verifies σ_i by computing

$$\hat{e}(\sigma_i, P) = \hat{e}(\omega_i M', Q_i)$$

If the above equation does not hold, A sends M' again to get the correct σ_i. Otherwise, A takes summation of all σ_i and unblinds to get the signature σ on the message M.

$$\sigma = r^{-1} \sum_{i \in S} \sigma_i \tag{1}$$

3.3 Signature Verification Protocol

The signature σ on a message M is accepted if and only if:

$$\hat{e}(\sigma, P) = \hat{e}(H(M), Q) \tag{2}$$

3.4 Correctness

Firstly, the correctness of the signature scheme must involve the correctness of verification of Eq.(2) in Signature Verification Protocol. That means the partial signature σ_i is valid if the signer i-th is honest. We have:

$$\begin{aligned} \hat{e}(\sigma_i, P) &= \hat{e}(\omega_i M', Q_i) \\ &= \hat{e}(\omega_i M', s_i P) \\ &= \hat{e}(\omega_i s_i M', P) \end{aligned}$$

Secondly, we verify the correctness of the threshold blind signature scheme. The scheme signature σ has form:

$$\begin{aligned} \sigma &= r^{-1} \sum_{i \in S} \sigma_i \\ &= r^{-1} \sum_{i \in S} s_i \prod_{\substack{j \in S \\ j \neq i}} \frac{j}{j-i} \cdot M' \qquad (3) \\ &= r^{-1} r s H(M) \qquad (4) \\ &= sH(M) \end{aligned}$$

Eq.(4) is derived from Eq.(3) by Lagrange interpolation.

The verification equation gives us:

$$\begin{aligned} \hat{e}(\sigma, P) &= \hat{e}(H(M), Q) \\ &= \hat{e}(H(M), sP) \\ &= \hat{e}(sH(M), P) \end{aligned}$$

Hence, if σ is valid signature on M, the verification equation always holds.

4 Security Analysis

In this section we discuss about the security aspects of our proposed scheme. The security consideration includes of blindness, robustness and unforgeability of the signature scheme.

On *blindness*, the user can get a valid signature on a message without revealing the content of message to signers. *Robustness* of the scheme ensures that the scheme can tolerate even $t-1$ of $n \geq 2t-1$ signers are corrupted. On the other hand, *unforgeability* not only is secure *against one-more-forgery* attack but also is secure even when $t-1$ signers were corrupted by an adversary. The first kind of attack was introduced in [13] and [14], which means that after getting ℓ blind signatures from signers, the user must not be able to produce more than ℓ signatures. The second attack indicates that even an adversary can corrupt up to $t-1$ signers, the adversary cannot produce a valid signature on a message.

Let $\mathcal{A}$ be an adversary who can corrupt up to $t-1$ signers as well as acts as a user in execution of the Signature Generation protocol. We have the following definition:

Definition 1 *Let* $\mathsf{TBS} = (\mathcal{TBK}, \mathcal{TBS}, \mathcal{TBV})$ *be the threshold blind signature scheme.* TBS *is secure threshold blind signature scheme if:*

1. Unforgeability. *No adversary who corrupts at most* $t-1$ *signers, with non-negligible probability, can do one-more forgery attack, that is an adversary cannot produce more than* ℓ *signature after executing* $\mathcal{TBS}$ *protocol* ℓ *times.*

2. Robustness. *Even there exists an adversary who can corrupt up to* $t-1$ *signers, the Key Generation and Signature Generation protocols complete successfully.*

4.1 Blindness

First of all, we state that our proposed signature scheme is blind. As pointed out in [4], the proposed signature scheme is blind. Since r is chosen randomly from $\mathbb{Z}_q^*$, therefore $M' = rH(M)$ is also a random element in group $\mathbb{G}_1$. Thus signers only receive the random information from the user and there is no way to know the original message. The signers also cannot link between the information they received and the message which is output by the user.

4.2 Robustness

The robustness of the proposed scheme is shown by the following theorem:

Theorem 1 *The threshold blind signature scheme* TBS *is robust for an adversary who can corrupt* $t-1$ *signers among* n *signers such that* $n \geq 2t-1$ *signers.*

Proof. As in [12], every signer chooses randomly a secret a_{i0} uniformly distributed in $\mathbb{Z}_q^*$ during $\mathcal{TBK}$ protocol. Therefore even there exists an adversary who can corrupt up to $t-1$ signers among $n \geq 2t-1$ signers, any subset of t signers constructs the unique secret key s uniformly distributed in $\mathbb{Z}_q^*$, thus the public key Q is uniformly distributed in $\mathbb{G}_1$. That means $\mathcal{TBK}$ completes successfully in case at most $t-1$ signers are corrupted.

In the signing protocol $\mathcal{TBS}$, every partial signature σ_i is verified by correspondent public key $Q_i = s_i P$. Even at most $t-1$ signers can be corrupted, the adversary still needs partial signatures from other signers to form t valid signature shares. With t valid signature shares, the signature $\sigma = sH(M)$ can be produced by Eq.(1) at step S3 of $\mathcal{TBS}$, and its correctness was shown in Section 3.4. Therefore $\mathcal{TBS}$ protocol completes successfully too. These showed that TBS is robust. □

4.3 Unforgeability

To show unforgeability, we utilize the fact that if the underlying signature scheme is secure then the corresponding threshold signature scheme is secure if it is simulatable, which was used in [10].

First, we consider the simulatable condition. The view $\mathcal{VIEW}_{\mathcal{A}}(\mathcal{TBS}(s_1, s_2, \ldots, s_n, (M, Q), \sigma)$ of the adversary $\mathcal{A}$ during the Signature Generation protocol consists of a message M, the public key Q, the information of corrupted signers s_i for $i = 1, 2, \ldots, t-1$ and the signature σ. Now we construct a simulator SIM which simulates $\mathcal{TBS}$:

SIM's input is a public key Q, a message M, a signature σ on M, secret shares $s_1, s_2, \ldots, s_{t-1}$ of corrupted signers.

1. SIM chooses $r' \in \mathbb{Z}_q^*$ randomly.
2. SIM computes partial signature:

$$\sigma_i' = r' s_i \omega_i H(M)$$

for $1 \leq i \leq t-1$.

3. For an uncorrupted signer, SIM computes partial signature as

$$\sigma_t' = r'\sigma - \sum_{i=1}^{t-1} \sigma_i'$$

Denote the information produced by the above simulator SIM as $\mathcal{SIM}(M, Q, s_1, s_2 \ldots, s_{t-1}, \sigma)$. We have the following lemma:

Lemma 1 $\mathcal{VIEW}_{\mathcal{A}}(\mathcal{TBS}(s_1, s_2 \ldots, s_n, (M, Q), \sigma))$ *and* $\mathcal{SIM}(M, Q, s_1, s_2, \ldots, s_{t-1}, \sigma)$ *have the same probability distribution.*

Proof. By comparing the information produced by SIM and $\mathcal{TBS}$ protocol we have:

1. Both the protocol and the simulator choose a blind factor randomly from $\mathbb{Z}_q^*$, r in $\mathcal{TBS}$ and r' in SIM. The probability distribution of r and r' are the same.
2. The $\mathcal{TBS}$ generates t partial signatures σ_i for $1 \leq i \leq t$. Each of them contains the blind factor r and the shared secret s_i, $1 \leq i \leq t$. The simulator SIM also produces t partial signatures σ_i' for $1 \leq i \leq t$. Each of them contains the blind factor r' and the share secret s_i, $1 \leq i \leq t$. Because blind factors r and r' have same probability distribution, partial signature σ_i and σ_i', $1 \leq i \leq t$ have same probability distribution too.

These complete the proof of Lemma 1. □

Now, we consider the blind signature presented in Section 2.2. We will use similar technique in [4], where the author defined "Chosen target CDH" assumption and proved that the blind signature scheme is secure assuming the hardness of the chosen-target CDH problem. According to the assumption, given a group $G = \langle g \rangle$ of prime order q, a random hash function $H : \{0,1\}^* \to G^*$ and a secret key $x \in \mathbb{Z}_q^*$ with the corresponding public key $y = g^x$, there is no polynomial-time adversary B which is given public key y, the target oracle $\mathcal{T}_G$ which outputs random point in G and the "helper" oracle $(\cdot)^x$ can output any subset of target points such that the number of queries to the helper oracle is strictly less than the number of queries to the target oracle. We propose the problem and assumption as follows:

Definition 2 *Let* $\mathbb{G}_1$ *be* GDH *group of prime order* q *and* P *is a generator of* $\mathbb{G}_1$. *Let* s *be a random element of* $\mathbb{Z}_q^*$ *and* $Q = sP$. *Let* $H : \{0,1\}^* \to \mathbb{G}_1$ *be a random hash function. The adversary* $\mathcal{B}$ *is given input* (q, P, Q, H) *and has access to the target oracle* $T_{\mathbb{G}_1}$ *that returns a random point* U_i *in* $\mathbb{G}_1$ *and the helper oracle* cdh-$s(\cdot)$. *Let* q_T *and* q_H *be the number of queries* $\mathcal{B}$ *made to the target oracle and the helper oracle respectively. The advantage of the adversary attacking the chosen-target CDH problem* $\mathsf{Adv}_{\mathbb{G}_1}^{ct-cdh}(\mathcal{B})$ *is defined as the probability of* $\mathcal{B}$ *to output a set of* l *pairs* $((V_1, j_1), (V_2, j_2), \ldots, (V_l, j_l))$, *for all* $i = 1, 2, \ldots, l$ $\exists$ $j_i = 1, 2, \ldots, q_T$ *such that* $V_i = sU_{j_i}$ *where all* V_i *are distinct and* $q_H < q_T$.
The chosen-target CDH assumption states that there is no polynomial-time adversary $\mathcal{B}$ *with non-negligible* $\mathsf{Adv}_{\mathbb{G}_1}^{ct-cdh}(\mathcal{B})$.

Under assumption that the chosen-target CDH problem is hard for all groups where CDH problem is hard, including GDH groups, we will show the blind signature proposed in Section 2.2 is secure by the following theorem:

Theorem 2 *If the chosen-target CDH assumption is true in the group $\mathbb{G}_1$ then the blind signature scheme $\mathsf{BGS}[\mathbb{G}_1]$ is secure against one-more forgery under chosen message attack.*

Proof. Let $\mathcal{A}$ be a polynomial time adversary attacking $\mathsf{BGS}[\mathbb{G}_1]$ against one-more forgery under chosen message attack. We will construct a polynomial time adversary $\mathcal{B}$ for chosen-target CDH problem such that $\mathsf{Adv}^{blind}_{\mathcal{BS},I}(\mathcal{A}) = \mathsf{Adv}^{ct-cdh}_{\mathbb{G}_1}(\mathcal{B})$.

The adversary $\mathcal{A}$ has access to a blind signing oracle $\mathtt{cdh}\text{-}s(\cdot)$ and the random hash oracle $H(\cdot)$. Then the adversary $\mathcal{B}$ can solve the chosen-target CDH problem by simulating $\mathcal{A}$. First, $\mathcal{B}$ provides $\mathtt{pk} = (q, P, H, Q)$ to $\mathcal{A}$ and $\mathcal{B}$ has to simulate the random hash oracle and the blind signing oracle for $\mathcal{A}$.

Each time $\mathcal{A}$ makes a new hash oracle query which differs from previous one, $\mathcal{B}$ will forward to its target oracle and returns the reply to $\mathcal{A}$. $\mathcal{B}$ stores the pair query-reply in the list of those pairs. If $\mathcal{A}$'s query is same as previous one, $\mathcal{B}$ will take and send the correspondent reply which $\mathcal{B}$ stored before.

If $\mathcal{A}$ makes a query to blind signing oracle, $\mathcal{B}$ will forward to its helper oracle $\mathtt{cdh}\text{-}s(\cdot)$ and returns the answer to $\mathcal{A}$.

At some point, $\mathcal{A}$ outputs a list of message-signature pairs $((M_1, \sigma_1), (M_2, \sigma_2), \ldots, (M_l, \sigma_l))$. $\mathcal{B}$ can find M_i in the list stored hash oracle query-reply for $i = 1, 2, \ldots, l$. Let j_i be the index of the found pair, then $\mathcal{B}$ can output its list as $((\sigma_1, j_1), (\sigma_2, j_2), \ldots, (\sigma_l, j_l))$.

In the view of $\mathcal{A}$, the above simulation and real protocol are indistinguishable and $\mathcal{B}$ is successful only if $\mathcal{A}$ is successful. Thus, $\mathsf{Adv}^{blind}_{\mathcal{BS},I}(\mathcal{A}) = \mathsf{Adv}^{ct-cdh}_{\mathbb{G}_1}(\mathcal{B})$. □

Theorem 3 *The threshold blind signature scheme TBS is as secure as the blind signature scheme $\mathsf{BGS}[\mathbb{G}_1]$ against one-more forgery under chosen message attack.*

Proof. The proof of Theorem 3 can be easily derived from Lemma 1 and Theorem 2. □

By Theorems 1 and 3, we can say that the proposed threshold blind signature scheme is secure and robust.

5 Performance evaluation

This section evaluates performance of the proposed scheme. The following tables show the comparison of computation in the Signature Generation protocol.

Oper.	KKL scheme	LLJ scheme	Our scheme
$\mathsf{A_m}$	$2t+1$	$2t+1$	0
M	$t+5$	$2n-t+6$	1
E	6	8	0
I	0	0	1
A	N/A	N/A	$t-1$
S	N/A	N/A	2

Table 1: Computation in the user side

Oper.	KKL scheme	LJY scheme	Our scheme
$\mathsf{A_m}$	2	$2(n-t+1)$	0
M	5	$2n-1$	1
E	8	6	0
I	0	0	0
A	N/A	N/A	0
S	N/A	N/A	1

Table 2: Computation in the signer side

In the above tables, $\mathsf{A_m}$, M, E and I mean modular addition, multiplication, exponentiation and inversion respectively. A and S denote point addition and scalar multiplication on an elliptic curve. KKL and LJY schemes are the threshold blind signature schemes in [9], [10] based on discrete logarithm problems. N/A means Not Available.

In the proposed scheme, to produce a signature, a user has to perform verification of partial signatures. The verification spends t scalar multiplication and $2t$ pairing computations. This may be burden for users. Therefore, the verification computation for users can be taken over by signers if we let signers perform verification of partial signatures. This can be done and save computation by designating a signer to do verification job.

Since the proposed scheme works on an elliptic curve, the advantage of the scheme is small key size. Moreover, the signature size produced by the proposed scheme is small, since it is an element in $\mathbb{G}_1$.

6 Concluding Remarks

We have proposed a secure and robust threshold blind signature scheme based on pairings. The scheme was proven as secure as the blind GDH signature scheme in random oracle model. In addition, our scheme exhibits robustness. Even there exists an adversary who can corrupt up to $t-1$ signers among $n \geq 2t-1$ signers, the scheme still completes successfully. As further work, we can design another threshold key generation protocol using secure Distributed Key Generation protocol, which probably makes our scheme more secure.

References

[1] M. Bellare, C. Namprempre, D. Pointcheval, M. Semanko, "The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme", *Cryptology ePrint Archive – 2001/02*.

[2] D. Boneh and M. Franklin, "ID-based Encryption from the Weil-pairing", *Advances in Cryptology – CRYPTO'2001*, LNCS 2139, Springer-Verlag, pp. 213–229, 2001.

[3] D. Boneh, H. Shacham, and B. Lynn, "Short Signatures from the Weil-pairing", *Advances in Cryptology – ASIACRYPT'2001*, LNCS 2248, Springer-Verlag, pp. 514–532, 2001.

[4] A. Boldyreva, "Threshold Signature, Multisignature and Blind Signature Schemes Based on the Gap-Diffie-Hellman-group Signature Scheme", *Public Key Cryptography - PKC 2003*, LNCS 2567, Springer-Verlag, pp. 31–46, 2003.

[5] D. Chaum, "Blind Signatures for Untraceable Payments", *Advances in Cryptology – CRYPTO'82*, pp. 199–203, Plenum, 1983.

[6] R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, "Robust Threshold DDS Signatures, *Advances in Cryptology'96 – EUROCRYPT'96*, LNCS 1070, Springer-Verlag, pp. 354-371, 1996.

[7] R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, "Secure Distributed Key Generation for Discrete-log Based Cryptosystems", *Advances in Cryptology'99 – EUROCRYPT'99*, LNCS 1592, Springer-Verlag, pp. 295-310, 1999.

[8] A. Joux and K. Nguyen, "Separating Decision Diffie-Hellman from Diffie-Hellman in Cryptographic Groups", *Cryptology ePrint Archive – 2001/03*.

[9] J. Kim, K. Kim and C. Lee, "An Efficient and Provably Secure Threshold Blind Signature", ICISC'2001, LNCS 2288, Springer-Verlag, pp. 318-327, 2002.

[10] C.L. Lei, W.S. Juang and P.L. Yu, "Provably Secure Blind Threshold Signatures Based on Discrete Logarithm", *National Computer Symposium 1999*, pp. C198–C205, 1999.

[11] T.P. Pedersen, "A Threshold Cryptosystem without a Trusted Party", *Advances in Cryptology – EUROCRYPT'91*, LNCS 547, Springer-Verlag, pp. 522–526, 1991.

[12] T.P. Pedersen, "Non-interactive and Information-theoretic Secure Verifiable Secret Sharing", *Advances in Cryptology – CRYPTO'91*, LNCS 576, Springer-Verlag, pp. 129–140, 1991.

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

[14] D. Pointcheval and J. Stern, "Security Argument for Digital Signatures and Blind Signatures", *Journal of Cryptology*, Springer-Verlag, Vol. 13 No. 3, pp. 361–396, 2000.

[15] A. Shamir, "How to Share a Secret", Commmunication of the ACM, Vol. 22, No. 11, pp. 612–613, Nov. 1979.

[16] F. Zhang and K. Kim, "ID-Based Blind Signature and Ring Signature from Pairings", *Advances in Cryptology – ASIACRYPT'2002*, LNCS 2501, Springer-Verlag, pp. 533–547, 2002.