# Threshold untraceable signature for group communications

T.-Y. Chang, C.-C. Yang and M.-S. Hwang

**Abstract:** Lee *et al.* have proposed an untraceable $(t, n)$ threshold signature scheme which can be extended to give the original signers the ability to prove they are the true signers. The present authors add the requirement of $(k, l)$ threshold-shared verification to the scheme of Lee *et al.* In the proposed scheme, any $t$ participants can represent a group to sign messages with/without anonymity, and $k$ verifiers can represent another group to authenticate the signature.

## 1 Introduction

Since digital signature techniques can achieve some tasks such as identifying senders, authenticating message contents, preventing denial of message ownership and protecting ownership, they are playing a more and more important role in our modern electronic society. Traditional digital signatures such as RSA [1–4] and DSA [5, 6] only allow a single signer to sign a message, and anyone can verify the signature at anytime. For distributing the power of a single signing, the threshold signatures are motivated by the need that arises in organisations to have a group of employees who agree on a message before signing.

In the threshold signature schemes, it is necessary to predetermine the threshold value $t$ so that at least $t$ participants in the group can collaborate to generate a valid signature on behalf of the group, but $t-1$ or fewer participants will not be enough. Anyone who plays the role of a verifier can use the group's public key to authenticate the signature.

In 1991, Desmedt and Frankel [7] proposed a $(t, n)$ threshold signature scheme based on the RSA cryptosystem [4]. Later, Li *et al.* [8] pointed out that $t$ or more malicious participants can forge the signature without taking any responsibility. In 1994, Harn [9] proposed an alternative $(t, n)$ threshold signature scheme based on Shamir's perfect secret scheme [10] and the modified ElGamal signature [11]. Based on the property of Lagrange polynomials, the group's secret key is distributed into $n$ different shadows to each participant. Any $t$ or more participants can use their shadows to generate individual signatures and combine $t$ individual signatures to obtain a threshold signature. On the other hand, to trace back to find the original signers in case of a forged document, several $(t, n)$ threshold schemes of traceability [12–14] and their comments [15–17] have been proposed.

In 2000, Wang *et al.* [18] proposed a new $(t, n)$ threshold signature scheme with $(k, l)$ threshold-shared verification. According to the security level of a document, not only can the document be signed by some specified signers in the group (signing group), but also it can be verified by some specified verifiers in another group (verifying group). For example, there are two companies connected in business with each other. The power of signing is distributed for several managers to represent a company to sign a contract with the other company via a computer network. The signature of the contract is generated as a threshold signature. For the same reason, the power of verifying is also distributed for several managers to represent the other company to verify the signature of contract.

Unfortunately, Tseng *et al.* [19] and Hsu *et al.* [20] have separately shown that Wang *et al.*'s scheme is insecure; any adversary can compute group secret keys from two valid threshold signatures. They also separately proposed their own improved schemes to withstand the attack. Recently, Lee [21] pointed out that the signing group secret key of the improved scheme of Tseng *et al.* is also apt to be disclosed. Fundamentally, the improved scheme [19] has the weaknesses: $t$ or more malicious participants can actually use the Lagrange polynomial formula to derive other participants' secret keys and system secrets. Furthermore, a shared distribution centre must take part in the generation of each threshold signature to distribute fresh shadows to all participants, which does not seem to fit in practical applications.

In 2000, Lee *et al.* [22] proposed an untraceable $(t, n)$ threshold signature scheme based on the Ohta–Okamoto signature scheme [23]. For the sake of privacy and safety, the identities of the signers should be anonymous in a democratic society. At the same time, their scheme can be extended to give the original signers the ability to prove they are true signers, and any $t$ or more malicious participants cannot reconstruct the polynomial to derive other participants' secret keys and system secrets. Furthermore, Lee *et al.* [22] pointed out that the scheme in [9] can be seen as an untraceable $(t, n)$ threshold signature scheme if the scheme does not provide an individual signature verification mechanism.

In this paper, we will attempt to combine Lee *et al.*'s $(t, n)$ untraceable scheme and the requirement of $(k, l)$ threshold-shared verification. Moreover, our scheme can be easily modified to provide the verification mechanism for individual signatures.



*IEE Proceedings* online no. 20040528

doi:10.1049/ip-com:20040528

Paper first received 17th February and in final revised form 7th August 2003

T.-Y. Chang is with the Department of Computer and Information Science, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu, Taiwan, Republic of China

C.-C. Yang and M.-S. Hwang are with the Department of Management Information Systems, National Chung Hsing University, 250 Kuo Kuang Road 402 Taichung, Taiwan, Republic of China

## 2 Review of Lee *et al.*'s scheme

In this Section, we shall first review Lee *et al.*'s untraceable $(t, n)$ threshold signature scheme. There is a shared distribution centre (SDC) which is responsible for initialising the system and generating the parameters in the system. The notation $G_s$ ($|G_s| = n$) is defined as the signing group of $n$ signers and $g_s$ ($|g_s| = t \leq n$) as any subset of $t$ signers in $G_s$. The scheme is divided into three phases as follows: the parametes generating phase; the individual signature generating phase; and the threshold signature generating and verifying phase.

### 2.1 Parameter generating phase

In this phase, the SDC is responsible for initialising the system and generating parameters as follows:

Step 1. Randomly choose two large secret primes $p$ and $q$, and compute a public number $N = p \cdot q$. To ensure that $p$ and $q$ are strong primes, let $p = 2p' + 1$ and $q = 2q' + 1$, where $p'$ and $q'$ are also secret primes.

Step 2. Let $\lambda(N) = 2p'q'$ (where $\lambda(N)$ is the Carmichael function) be secret and randomly choose a public number $W \approx 10^{50}$, where $gcd(\lambda(N), W) = 1$.

Step 3. Randomly choose a secret primitive $\alpha$ in both $GF(p)$ and $GF(q)$.

Step 4. Randomly choose a secret polynomial $f_s(x) \bmod \lambda(N)$ of degree $t-1$, where $f_s(0) = d$ and $gcd(\lambda(N), d) = 1$.

Step 5. Compute $S = \alpha^d \bmod N$ as a $G_s$'s secret key and the associated $Y = \alpha^{-d \cdot W} \bmod N$ as this $G_s$'s public key.

Step 6. Randomly select $n$ public and odd integers $x_{si}$ with even $f_s(x_{si})$ [7] for each participant $P_{si}$ in $G_s$ ($i \in G_s$), and their secret keys $K_{si}$ are as follows:

$$K_{si} = \alpha^{s_i} \bmod N, \text{ where}$$

$$s_i = \frac{f_s(x_{si})/2}{\left[\prod_{\substack{j \in G_s \\ j \neq i}} (x_{si} - x_{sj})\right]/2} \bmod p'q' \tag{1}$$

Step 7. Select a public collision-free one-way hash function $H(\cdot)$.

### 2.2 Individual signature generating phase

In this phase, $P_{si}$ generates her/his individual signature. Without loss of generality, assume that $t$ participants $P_{s1}, P_{s2}, \ldots, P_{st}$ in $g_s$ are to sign a message $m$. Each $P_{si}$ randomly chooses an integer $r_{si}$ with $0 < r_{si} < N$, and computes $u_{si}$ as follows:

$$u_{si} = r_{si}^W \bmod N \tag{2}$$

Then, $P_{si}$ broadcasts $u_{si}$ to the other $t-1$ participants in $g_s$. Once each $P_{si}$ receives $u_j$ ($j = 1,2,\ldots,t$ and $j \neq i$), she/he computes $U_s$ and a hash value $e$ as follows:

$$U_s = \prod_{i \in g_s} u_{si} \bmod N \tag{3}$$

$$e = H(U_s, m) \tag{4}$$

Then, each $P_{si}$ uses her/his secret key $K_{si}$ to generate her/his individual signature as follows:

$$z_{si} = r_{si} \cdot K_{si}^{\prod_{\substack{j \in G_s \\ j \notin g_s}} (x_{si} - x_{sj}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} (0 - x_{sj}) \cdot e} \bmod N \tag{5}$$

Each $P_{si}$ sends $(z_{si}, m)$ to a designated clerk who is responsible for the computation of the threshold signature. There is no secret value kept by the clerk, so the clerk can be a general computer.

### 2.3 Threshold signature generating and verifying phase

After receiving $t$ individual signatures, the clerk computes the threshold signature as follows:

$$Z_s = \prod_{i \in g_s} z_{si} \bmod N \tag{6}$$

To verify the threshold signature $\{e, Z_s\}$ for the message $m$, the verifier first computes a value $\tilde{U}_s$ as follows:

$$\tilde{U}_s = Z_s^W \cdot Y^e \bmod N \tag{7}$$

Then, the verifier checks the following equation:

$$e \stackrel{?}{=} H(\tilde{U}_s, m) \tag{8}$$

If (8) holds, the threshold signature $\{e, Z_s\}$ is valid.

## 3 Proposed scheme

In this Section, we shall add the requirement of $(k, l)$ threshold-shared verification to Lee *et al.*'s scheme. The notation $G_v$ ($|G_v| = l$) is defined as the verifying group of $l$ verifiers and $g_v$ ($|g_v| = k \leq l$) as any subset of $k$ verifiers in $G_v$. Here, we first present the properties of an untraceable $(t, n)$ threshold signature scheme with $(k, l)$ threshold-shared verification.

- Only $t$ out of $n$ in $G_s$ can generate the signature on behalf of the group.
- Only $k$ out of $l$ in $G_v$ can verify the threshold signature on behalf of the group.
- The signer of the threshold signature cannot be traced.

Our scheme also consists of three phases as follows: the parameter generating phase; the individual signature generating phase; and the threshold signature generating and verifying phase.

### 3.1 Parameter generating phase

The system notations (SDC, $G_s$, $g_s$) and parameters ($p$, $q$, $p'$, $q'$, $N$, $W$, $\lambda(N)$, $\alpha$, $d$, $S$, $Y$, $x_{si}$, $K_{si}$, $H(\cdot)$) are the same as those in Lee *et al.*'s. scheme. The SDC performs the following steps to initialise the system and generate parameters as follows:

Step 1. Randomly choose two numbers $a$ and $b$ such that the greatest common divisor of $a$ and $b$ is 1. When $gcd(a,b) = 1$, there must be exactly two integers $c$ and $h$ that satisfy the equation $a \cdot c + b \cdot h = 1$. The extended Euclidean algorithm [24] can find such integers.

Step 2. Randomly choose two secret polynomials $f_s(x) \bmod \lambda(N)$ of degree $t-1$ and $f_v(x) \bmod \lambda(N)$ of degree $k-1$, where $f_s(0) = d \cdot a \cdot c$, $f_v(0) = d \cdot b \cdot h$ and $gcd(\lambda(N), d) = 1$.

Step 3. Randomly select $l$ public and odd integers $x_{vi}$ with even $f_v(x_{vi})$ for each participant $P_{vi}$ in $G_v$ ($i \in G_v$), and their secret keys $K_{vi}$ are as follows:

$$K_{vi} = \alpha^{v_i} \bmod N, \text{ where}$$

$$v_i = \frac{f_v(x_{vi})/2}{\left[\prod_{\substack{j \in G_v \\ j \neq i}} (x_{vi} - x_{vj})\right]/2} \bmod p'q' \tag{9}$$

### 3.2 Individual signature generating phase

The message $m$ and the values ($u_{si}$, $U_s$, $e$, $z_{si}$) are separately computed in (2), (3), (4) and (5), respectively, which are the same as the equations in Lee *et al.*'s. scheme.

In order to trace back to find the original signers of a forged document, our scheme can provide an individual signature verification mechanism. In other words, the individual signatures should be verified by a clerk who is responsible for the verification and computation of the threshold signature. Because the clerk has to verify the individual signatures by using the individual signers' public keys before generating the group signature, the clerk knows who has signed the document and can securely record it in a database. In the parameter generating phase, the SDC distributes a public key $y_{si}$ for each $P_{si}$ in $G_s$ as follows:

$$y_{si} = \alpha^{-s_i \cdot W} \bmod N, \text{ where}$$

$$s_i = \frac{f_s(x_{si})/2}{\left[\prod_{\substack{j \in G_s \\ j \neq i}} (x_{si} - x_{sj})\right]/2} \bmod p'q' \tag{10}$$

In the individual signature-generating phase, each $P_{si}$ produces the values ($u_{si}$, $U_s$, $e$, $z_{si}$) separately by (2), (3), (4) and (5), respectively, and added to them is a hash value $e_i$ as follows:

$$e_i = H(u_{si}, m) \tag{11}$$

After receiving ($e$, $e_i$, $z_{si}$, $m$), the clerk uses $P_{si}$'s public key $y_{si}$ to compute a value $\tilde{u_{si}}$ as follows:

$$\tilde{u_{si}} = z_{si}^{W} \cdot y_{si}^{\prod_{\substack{j \in G_s \\ j \notin g_s}} (x_{si} - x_{sj}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} (0 - x_{sj}) \cdot e} \bmod N \tag{12}$$

and checks the following equation:

$$e_i \stackrel{?}{=} H(\tilde{u_{si}}, m) \tag{13}$$

If (13) holds, the individual signature $z_{si}$ on the message $m$ is valid. So the individual signature verifying mechanism is put into the individual signature generating phase. After $t$ individual signatures are verified, the clerk computes the threshold signature $Z_s$ in (6). The remaining steps in the threshold signature generating and verifying phase are the same as those in Lee *et al.*'s scheme.

### 3.3 Threshold signature generating and verifying phase

The generation of the threshold signature $Z_s$ in (6) in Lee *et al.*'s scheme is also present here. Then, the threshold signature $\{e, Z_s\}$ of the message $m$ is transmitted to $G_v$. To verify the group signature, any $k$ out of the $l$ verifiers in $G_v$ should cooperate to authenticate the validity of the signature. Without loss of generality, assume that there are $k$ participants $P_{v1}, P_{v2}, \ldots, P_{vk}$ in $g_v$. Each $P_{vi}$ randomly chooses an integer $r_{vi}$ with $0 < r_{vi} < N$ and computes $u_{vi}$ and $z_{vi}$ as follows:

$$u_{vi} = r_{vi}^{W} \bmod N \tag{14}$$

$$z_{vi} = r_{vi} \cdot K_{vi}^{\prod_{\substack{j \in G_v \\ j \notin g_v}} (x_{vi} - x_{vj}) \cdot \prod_{\substack{j \in g_v \\ j \neq i}} (0 - x_{vj}) \cdot e} \bmod N \tag{15}$$

Then, each $P_{vi}$ transmits $u_{vi}$ and $z_{vi}$ to a clerk who can be randomly chosen from $G_v$ to compute $U_v$ and $Z_v$ as follows:

$$U_v = \prod_{i \in g_v} u_{vi} \bmod N \tag{16}$$

$$Z_v = \prod_{i \in g_v} z_{vi} \bmod N \tag{17}$$

Afterwards, the threshold signature can be verified by using $G_s$'s public key $Y$ to compute a value $\tilde{U_s}$ as follows:

$$\tilde{U_s} = (Z_s \cdot Z_v)^{W} \cdot (U_v)^{-1} \cdot Y^{e} \bmod N \tag{18}$$

To authenticate the validity of the threshold signature is to check (8). If (8) holds, the threshold signature $\{e, Z_s\}$ on the message $m$ is valid.

*Theorem 1.* The proposed scheme is a ($t$, $n$) threshold signature scheme with ($k$, $l$) threshold-shared verification.

*Proof.* According to (2), (3) and (4), we can rewrite the hash value $e$ as follows:

$$e = H\left(\prod_{i \in g_s} r_{si}^{W}, m\right)$$

From (1), (5) and (6), the values $z_{si}$ and $Z_s$ can be derived as follows:

$$z_{si} = r_{si} \cdot K_{si}^{\prod_{\substack{j \in G_s \\ j \notin g_s}} (x_{si} - x_{sj}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} (0 - x_{sj}) \cdot e} \bmod N$$

$$= r_{si} \cdot \alpha^{s_i \cdot \prod_{\substack{j \in G_s \\ j \notin g_s}} (x_{si} - x_{sj}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} (0 - x_{sj}) \cdot e} \bmod N$$

$$= r_{si} \cdot \alpha^{\frac{f_s(x_{si})}{\prod_{\substack{j \in G_s \\ j \neq i}} (x_{si} - x_{sj})} \cdot \prod_{\substack{j \in G_s \\ j \notin g_s}} (x_{si} - x_{sj}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} (0 - x_{sj}) \cdot e} \bmod N$$

$$= r_{si} \cdot \alpha^{f_s(x_{si}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} \frac{(0 - x_{sj})}{(x_{si} - x_{sj})} \cdot e} \bmod N$$

and

$$Z_s = \prod_{i \in g_s} r_{si} \cdot \alpha^{\sum_{i \in g_s} f_s(x_{si}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} \frac{(0 - x_{sj})}{(x_{si} - x_{sj})} \cdot e} \bmod N$$

By using the Lagrange interpolating polynomial, with the knowledge of $t$ pairs of ($x_{si}$, $f_s(x_{si})$), the unique ($t-1$)th degree polynomial $f_s(x_{si})$ and $f_s(0)$ can be determined as follows:

$$f_s(x) = \sum_{i \in g_s} f_s(x_{si}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} \frac{(x - x_{sj})}{(x_{si} - x_{sj})} \bmod \lambda(N)$$

$$f_s(0) = \sum_{i \in g_s} f_s(x_{si}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} \frac{(0 - x_{sj})}{(x_{si} - x_{sj})} \bmod \lambda(N)$$

Thus,

$$Z_s = \prod_{i \in g_s} r_{si} \cdot \alpha^{\sum_{i \in g_s} f_s(x_{si}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} \frac{(0 - x_{sj})}{(x_{si} - x_{sj})} \cdot e} \bmod N$$

$$= \prod_{i \in g_s} r_{si} \cdot \alpha^{d \cdot a \cdot c \cdot e} \bmod N$$

For the same reason, by using Lagrange interpolating polynomial, the value $Z_v$ can be derived from (9), (15) and

(17) as follows:

$$Z_v = \prod_{i \in g_v} r_{vi} \cdot \alpha^{d \cdot b \cdot h \cdot e} \bmod N$$

If the threshold signature $\{e, Z_s\}$ is valid and the value $Z_v$ is produced by $k$ verifiers in $G_v$, the values $U_s$ and $\tilde{U}_s$, separately computed in (3) and (18), should be as follows:

$$\begin{aligned} U_s &= \prod_{i \in g_s} u_{si} \bmod N \\ &= \prod_{i \in g_s} r_{si}^W \bmod N \end{aligned}$$

and

$$\begin{aligned} \tilde{U}_s &= (Z_s \cdot Z_v)^W \cdot (U_v)^{-1} \cdot Y^e \bmod N \\ &= \left( \prod_{i \in g_s} r_{si} \cdot \alpha^{d \cdot a \cdot c \cdot e} \cdot \prod_{i \in g_v} r_{vi} \cdot \alpha^{d \cdot b \cdot h \cdot e} \right)^W \\ &\quad \cdot \left( \prod_{i \in g_v} r_{vi}^W \right)^{-1} \cdot (\alpha^{-d \cdot W})^e \bmod N \\ &= \prod_{i \in g_s} r_{si}^W \cdot \prod_{i \in g_v} r_{vi}^W \cdot \alpha^{d \cdot e \cdot W \cdot (a \cdot c + b \cdot h)} \\ &\quad \cdot \left( \prod_{i \in g_v} r_{vi}^W \right)^{-1} \cdot \alpha^{-d \cdot W \cdot e} \bmod N \\ &= \prod_{i \in g_s} r_{si}^W \bmod N \end{aligned}$$

Therefore, the correctness of (8) can be verified. In this case, $\{e, Z_s\}$ must be a signature generated by $t$ signers in $G_s$, and only $k$ verifiers in $G_v$ can verify it. Q.E.D.

*Theorem 2*. The proposed $(t, n)$ threshold signature scheme with $(k, l)$ threshold-shared verification is untraceable.

*Proof*. Assume that there are two subsets $g_s$ and $g'_s$ in $G_s$, where $|g_s| = |g'_s| = t$. Each $P_{si}$ in $g_s$ generates her/his individual signature $z_{si}$ and the threshold signature $\{e, Z_s\}$. If it is impossible to tell which individuals collaborate to generate the threshold signature $\{e, Z_s\}$, our scheme is untraceable. In other words, the pair $(r'_{si}, u'_{si})$ generated by $P'_{si}$ in $g'_s$ is indistinguishable from $(r_{si}, u_{si})$ originally generated by $P_{si}$ in $g_s$. From $z_{si}$ in (5), each $P'_{si}$ in $g'_s$ can compute $(r'_{si}, u'_{si})$ as follows:

$$r'_{si} = z_{si} \cdot \left( \frac{\prod_{j \in G_s} (x_{si} - x_{sj}) \cdot \prod_{\substack{j \in g'_s \\ j \neq i}} (0 - x_{sj}) \cdot e}{K'^{\,j \notin g'_s}_{si}} \right)^{-1} \bmod N$$

and

$$u'_{si} = r'^W_{si} \bmod N$$

Since the threshold signature $Z_s$ can be expressed as:

$$\begin{aligned} Z_s &= \prod_{i \in g_s} z_{si} \bmod N \\ &= \prod_{i \in g_s} \left( \frac{\prod_{j \in G_s} (x_{si} - x_{sj}) \cdot \prod_{\substack{j \in g_s \\ j \neq i}} (0 - x_{sj}) \cdot e}{r_{si} \cdot K_{si}^{j \notin g_s}} \right) \bmod N \\ &= \prod_{i \in g_s} \left( \frac{\prod_{j \in G_s} (x_{si} - x_{sj}) \cdot \prod_{\substack{j \in g'_s \\ j \neq i}} (0 - x_{sj}) \cdot e}{r'_{si} \cdot K'^{\,j \notin g'_s}_{si}} \right) \bmod N \\ &= \prod_{i \in g_s} r_{si} \cdot \alpha^{d \cdot a \cdot c \cdot e} \bmod N \\ &= \prod_{i \in g'_s} r'_{si} \cdot \alpha^{d \cdot a \cdot c \cdot e} \bmod N \end{aligned}$$

It implies that

$$\prod_{i \in g_s} r_{si} = \prod_{i \in g'_s} r'_{si} \bmod N$$

so the equation

$$\prod_{i \in g_s} u_{si} = \prod_{i \in g'_s} u'_{si} \bmod N$$

is also true. Q.E.D.

Like Lee *et al.*'s scheme, our scheme can also be further extended (See [21] for a more detailed description) to give the original signers the ability to prove that they are the true signers. In the individual signature generating phase, each $P_{si}$ randomly chooses an integer $\overline{r_{si}}$ and computes a value $\overline{u_{si}} = \overline{r_{si}}^L \bmod N$, additionally. Each $P_{si}$ broadcasts $\overline{u_{si}}$ to the other $t-1$ participants in $g_s$ to produce $U_s$ in (3) and a new hash value $E$ as follows:

$$E = H(\overline{u_{s1}}, \overline{u_{s2}}, \cdots, \overline{u_{st}})$$

Then, each $P_{si}$ in $g_s$ replaces (4) with the following equation:

$$e = H(U_s, E, m)$$

After the computation of (5) and (6), $\{e, Z_s, E\}$ becomes the threshold signature of $m$. After (16), (17) and (18) have been performed, the threshold signature can be verified by the following equation in place of (8).

$$e = H(\tilde{U}_s, E, m)$$

If the above equation holds, the threshold signature $\{e, Z_s, E\}$ on the message $m$ is valid. If the original signers agree to make it public that they are the true signers, they can show $(\overline{r_{si}}, \overline{u_{si}})$ to an arbiter. The arbiter checks the following equations:

$$E \stackrel{?}{=} H(\overline{u_{s1}}, \overline{u_{s2}}, \cdots, \overline{u_{st}})$$

and

$$\overline{u_{si}} \stackrel{?}{=} \overline{r_{si}}^L \bmod N$$

If the above equations hold, the arbiter will believe that they are the true signers.

*Theorem 3*. The individual signatures can be verified by the clerk.

*Proof.* The value $u_{si}$ separately computed in (2) and (18) is

$$u_{si} = r_{si}^{W} \bmod N$$

and

$$\begin{aligned} u_{si} &= z_{si}^{W} \cdot y_{si}^{\prod_{\substack{j\in G_s \\ j\notin g_s}}(x_{si}-x_{sj})\cdot\prod_{\substack{j\in g_s \\ j\neq i}}(0-x_{sj})\cdot e} \bmod N \\ &= \left(r_{si}\cdot K_{si}^{\prod_{\substack{j\in G_s \\ j\notin g_s}}(x_{si}-x_{sj})\cdot\prod_{\substack{j\in g_s \\ j\neq i}}(0-x_{sj})\cdot e}\right)^{W} \\ &\quad \cdot y_{si}^{\prod_{\substack{j\in G_s \\ j\notin g_s}}(x_{si}-x_{sj})\cdot\prod_{\substack{j\in g_s \\ j\neq i}}(0-x_{sj})\cdot e} \bmod N \\ &= \left(r_{si}\cdot \alpha^{s_i\cdot\prod_{\substack{j\in G_s \\ j\notin g_s}}(x_{si}-x_{sj})\cdot\prod_{\substack{j\in g_s \\ j\neq i}}(0-x_{sj})\cdot e}\right)^{W} \\ &\quad \cdot \alpha^{-s_i\cdot W\cdot\prod_{\substack{j\in G_s \\ j\notin g_s}}(x_{si}-x_{sj})\cdot\prod_{\substack{j\in g_s \\ j\neq i}}(0-x_{sj})\cdot e} \bmod N \\ &= r_{si}^{W} \bmod N \end{aligned}$$

Therefore, the correctness of (13) can be verified. That is, ($e$, $e_i$, $z_{si}$) must be an individual signature on message $m$. Q.E.D.

Obviously, in our scheme, if $P_{si}$ in $G_s$ has computed $e_i$ and sent it to the clerk to verify her/his individual signature, the properties of our scheme are the same as those of the schemes in [19, 20]. In other words, the participants in the signing group can determine whether or not to stay anonymous. Furthermore, the SDC can be revoked after the parameters generating phase in our scheme.

## 4 Security analysis

The security of our proposed scheme is the same as that of Lee *et al.*'s scheme, which is based on the difficulty of breaking the RSA scheme [4]. Theorem 1 shows that any subset of $t$ participants in $G_s$ can generate the threshold signature and any subset of $k$ participants in $G_v$ can verify the group signature. Note that the attackers on the proposed schemes may come from inside (denoted as 'impostors') or outside (denoted as 'adversaries') of the signing/verifying group. In the rest of this Section, some possible attacks are raised and fought against to prove the security of our scheme.

*Attack 1:* An adversary tries to reveal $G_s$'s secret key $S$ from the known information as the following cases:

Case 1: The equation $Y = S^{-W} \bmod N$ and $G_s$'s keys $Y$, $N$ and the parameter $W$ are known. It is as difficult as breaking the RSA scheme to reveal $G_s$'s secret key $S$.

Case 2: The equation $S = \alpha^{d} \bmod N$ is known. The adversary should first reconstruct the polynomial $f_s(x)$ mod $\lambda(N)$ to obtain $f_s(0) = d \cdot a \cdot c$. Then, she/he has to calculate the multiplicative inverse for $a \cdot c \bmod \lambda(N)$. However, $f_s(x)$, $\lambda(N)$, $a$, $c$ and the primitive element $\alpha$ are secret.

Case 3: The equations $Z_s = \prod_{i\in g_s} \cdot S^{a\cdot c\cdot e} \bmod N$, $U_s = \prod_{i\in g_s} r_{si}^{W} \bmod N$ and a valid signature $\{e, Z_s\}$ are known. The adversary should first find out the random product $\prod_{i\in g_s} r_{si}$ from $U_s$. Then, she/he has to calculate the $(a\cdot c\cdot e)$th root of $Z_s \cdot (\prod_{i\in g_s} r_{si})^{-1} \bmod N$. However, retrieving $\prod_{i\in g_s} r_{si}$ from $U_s$ is as difficult as breaking the RSA scheme, and the difficulty of extracting the $(a\cdot c\cdot e)$th root of $Z_s \cdot (\prod_{i\in g_s} r_{si})^{-1} \bmod N$ is equivalent to breaking the RSA scheme when $gcd(a\cdot c\cdot e, \lambda(N)) = 1$ and equivalent to factoring $N$ if $gcd(a\cdot c\cdot e, p-1) = 1$ or $gcd(a\cdot c\cdot e, q-1) = 1$. Moreover, the parameters ($a$, $c$) are secret.

*Attack 2:* An adversary tries to reveal $P_{si}$'s secret key $K_{si}$ in $G_s$ from the known information.

Case 1: The equation $y_{si} = K_{si}^{-W} \bmod N$ and $P_{si}$'s public keys $y_{si}$, $N$ and the parameter $W$ are known. It is as difficult as breaking the RSA scheme to reveal $P_{si}$'s secret key $K_{si}$.

Case 2: Equation (1) and the public value $x_{si}$ are known. It is infeasible for the adversary to derive $P_{si}$'s secret key $K_{si}$ if $f_s(x_{si})$, $\alpha$, $p'$ and $q'$ are unknown.

Case 3: Equations (2) and (5) and the individual signature $z_{si}$ are known. The adversary should first find out the random number $r_{si}$ from $u_{si}$. Then, she/he has to calculate the

$$\left(\prod_{\substack{j\in G_s \\ j\notin g_s}}(x_{si}-x_{sj})\cdot\prod_{\substack{j\in g_s \\ j\neq i}}(0-x_{sj})\cdot e\right)\text{th}$$

root of $z_{si}\cdot r_{si}^{-1} \bmod N$. However, retrieving $r_{si}$ from $u_{si}$ is as difficult as breaking the RSA scheme. The difficulty of extracting the

$$\left(\prod_{\substack{j\in G_s \\ j\notin g_s}}(x_{si}-x_{sj})\cdot\prod_{\substack{j\in g_s \\ j\neq i}}(0-x_{sj})\cdot e\right)\text{th}$$

root of $z_{si}\cdot r_{si}^{-1} \bmod N$ is equivalent to breaking the RSA scheme.

*Attack 3:* An adversary tries to reveal $P_{vi}$'s secret key $K_{vi}$ in $G_v$ from the known information. As with cases 1 and 2 in attack 1, the adversary should face the difficulty of breaking the RSA scheme.

*Attack 4:* An adversary tries to impersonate $P_{si}$ in $g_s$.

The adversary first chooses a random integer $r'_{si}$ with $0 < r'_{si} < N$, and broadcasts $u_{si} = r_{si}'^{W} \bmod N$. The adversary can obtain the productive value

$$U'_s = \prod_{\substack{j\in g_s \\ j\neq i}} u_{sj}\cdot u_{si} \bmod N$$

and the hash value $e' = H(U'_s, m)$. Without knowing $P_{si}$'s secret key $K_{si}$, the adversary should face the difficulty of generating a valid value $z'_{si}$ in (18) to satisfy the following equation:

$$z_{si}'^{W}\cdot\prod_{\substack{j\in g_s \\ j\neq i}} z_{sj}^{W} = U'_s\cdot Y^{-e}\cdot Z_v^{-W}\cdot U_v \bmod N$$

Moreover, if the broadcasting cryptosystems [25, 26] or the secret channel exist in $G_s$, it can keep the adversary from knowing the productive values

$$\prod_{\substack{j\in g_s \\ j\neq i}} u_{sj} \bmod N$$

and

$$\prod_{\substack{j\in g_s \\ j\neq i}} z_{sj}^{W} \bmod N.$$

*Attack 5:* An adversary tries to impersonate $P_{vi}$ in $g_v$.

As with attack 4, the adversary should face the difficulty of generating a valid value $z'_{vi}$ in (18).

*Attack 6:* In $G_s$, $t$ or more malicious impostors try to reconstruct the secret polynomial $f_s(x)$ of degree $t-1$ to obtain other participants' secret keys $K_{si}$. Since the SDC distributes $K_{si} = \alpha^{si} \bmod N$ in place of $s_i$, $t$ or more pairs $(x_{si}, K_{si})$ cannot help reconstruct $f_s(x)$. If the impostor tries to reveal $s_i$ from $K_{si}$, it is as difficult as the problem of solving the discrete logarithm modulo a composite number $N$ if $\alpha$ is known. Furthermore, $\alpha$ and $\lambda(N)$ are secret in our scheme. On the other hand, $t$ or more malicious impostors cannot collude to retrieve $\lambda(N)$.

*Attack 7:* In $G_v$, $k$ or more malicious impostors try to reconstruct the secret polynomial $f_k(x)$ of degree $k-1$ to obtain other participants' secret keys $K_{vi}$. As with attack 6, $k$ or more malicious impostors in $G_v$ cannot reconstruct the secret polynomial of degree $k-1$ to obtain other participants' secret keys $K_{vi}$.

*Attack 8:* An adversary tries to forge the group signature $\{e, Z_s\}$ for the message $m$.

The adversary randomly computes the productive value $U_s$ and the hash value $e = H(U_s, m)$. Then, the adversary has to figure out $Z_s$ from $Z_s^W = U_s \cdot Y^{-e} \cdot Z_v^{-W} \cdot U_v \bmod N$. It is as difficult as breaking the RSA scheme. On the other hand, the adversary may also try to randomly generate a threshold signature $\{e, Z_s\}$ and compute $U_s = (Z_s \cdot Z_v)^W \cdot Y^e \cdot (U_v)^{-1} \bmod N$. However, $H$ is a collision-free one-way hash function; it is difficult to find a message $m'$ such that $e = H(U_s, m')$.

*Attack 9:* All $P_{si}$ and $P_{vi}$ should separately keep the random values $r_{si}$ and $r_{vi}$ secret.

From two valid threshold signatures $\{e_1, Z_{s1}\}$ and $\{e_2, Z_{s2}\}$, the adversary can obtain the following equations:

$$\begin{cases} S^{e_1} = \left(\prod_{i \in g_s} r_{si}\right)^{-1} \cdot Z_{s1} \bmod N \\ S^{e_2} = \left(\prod_{i \in g_s} r_{si}\right)^{-1} \cdot Z_{s2} \bmod N \end{cases}$$

$$\begin{cases} (\alpha^{d \cdot b \cdot h})^{e_1} = \left(\prod_{i \in g_v} r_{vi}\right)^{-1} \cdot Z_{v1} \bmod N \\ (\alpha^{d \cdot b \cdot h})^{e_2} = \left(\prod_{i \in g_v} r_{vi}\right)^{-1} \cdot Z_{v2} \bmod N \end{cases}$$

If $gcd(e_1, e_2) = 1$, the group secret key $S$ and the value $\alpha^{d \cdot b \cdot h}$ can be revealed by the Euclidean algorithm. Then, anyone can easily use $S$ and $\alpha^{d \cdot b \cdot h}$ to generate and verify other threshold signatures without cooperation, respectively. To separately reveal $r_{si}$ and $r_{vi}$ from $u_{si}$ in (2) and $u_{vi}$ in (14), it is as difficult as breaking the RSA scheme. Under the same protection, the adversary has to break the RSA scheme to separately reveal $\prod_{i \in g_s} r_{si}$ and $\prod_{i \in g_v} r_{vi}$ from $U_s$ in (3) and $U_v$ in (16).

## 5 Conclusion

In this paper, we have added the requirement of $(k, l)$ threshold-shared verification to Lee *et al.*'s scheme by using the extended Euclidean algorithm and demonstrated the ability of our new scheme to work against some possible attacks. Our security analysis has revealed that our scheme can withstand these attacks under factorisation. Moreover, our scheme provides both traceability mode and untraceability mode for the participants to choose from. With untraceability, the original signers also have the ability to prove they are the true signers.

## 6 Acknowledgment

This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract NSC90-2213-E-324-004.

## 7 References

1 Chang, C.C., and Hwang, M.S.: 'Parallel computation of the generating keys for RSA cryptosystems', *Electron. Lett.*, 1996, **32**, (15), pp. 1365–1366
2 Changchien, S.W., Hwang, M.S., and Hwang, K.F.: 'A batch verifying and detecting multiple RSA digital signatures', *Int. J. Comput. Numer. Anal. Appl.*, 2002, **2**, (3), pp. 303–307
3 Hwang, M.S., Lin, I.C., and Hwang, K.F.: 'Cryptanalysis of the batch verifying multiple RSA digital signatures', *Informatica*, 2000, **11**, (1), pp. 15–19
4 Rivest, R.L., Shamir, A., and Adleman, L.: 'A method for obtaining digital signatures and public key cryptosystems', *Commun. ACM*, 1978, **21**, pp. 120–126
5 Hwang, M.S., Lee, C.C., and Lai, Y.C.: 'Traceability on low-computation partially blind signatures for electronic cash', *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, 2002, **E85-A**, (5), pp. 1181–1182
6 National Institute of Satndards and Technology (NIST): 'The digital signature standard proposed by NIST', *Commun. ACM*, 1992, **35**, (7), pp. 36–40
7 Desmedt, Y., and Frankel, Y.: 'Shared generation of authenticators'. Proc. Advances in Cryptology, CRYPTO'91, Santa Barbara, CA, USA, 1991, pp. 457–469
8 Li, C.M., Hwang, T., and Lee, N.Y.: 'Remark on the threshold RSA signature scheme'. Proc. Advances in Cryptology CRYPTO'93, Santa Barbara, CA, USA, 1993, pp. 413–420
9 Harn, L.: 'Group-oriented $(t, n)$ threshold signature and digital multisignature', *IEE Proc., Comput. Digit. Tech.*, 1994, **141**, (5), pp. 307–313
10 Shamir, A.: 'How to share a secret', *Commun. ACM*, 1979, **22**, pp. 612–613
11 Agnew, G.B., Mullin, B.C., and Vanstone, S.A.: 'Improved digital signature scheme based on discrete exponentiation', *Electron. Lett.*, 1990, **26**, (14), pp. 1024–1025
12 Lee, W.B., and Chang, C.C.: '$(t, n)$ threshold digital signature with traceability property', *J. Inf. Sci. Eng.*, 1999, **15**, (5), pp. 669–678
13 Li, C.M., Hwang, T., and Lee, N.Y.: 'Threshold-multisignature schemes where suspected forgery implies traceability of adversarial shareholders'. Proc. Advances in Cryptology, Eurocrypt'94, Perugia, Italy, 1994, pp. 194–204
14 Michels, M., and Horster, P.: 'On the risk of disruption in several multiparty signature schemes'. Proc. Asiacrypt'96, Kyongju, South Korea, 1996, pp. 334–345
15 Li, Z.C., Hui, L.C.K., Chow, K. P., Chong, C.F., Tsang, W.W., and Chan, H.W.: 'Security of Wang *et al.*'s group-oriented $(t, n)$ threshold signature schemes with traceable signers', *Inf. Proc. Lett.*, 2001, **80**, (6), pp. 295–298
16 Tseng, Y.M., and Jan, J.K.: 'Attacks on threshold signature schemes with traceable signers', *Inf. Process. Lett.*, 1999, **71**, (1), pp. 1–4
17 Wang, C.T., Lin, C.H., and Chang, C.C.: 'Research note threshold signature schemes with traceable signers in group communications', *Comput. Commun.*, 1998, **21**, (8), pp. 771–776
18 Wang, C.T., Chang, C.C., and Lin, C.H.: 'Generalization of threshold signature and authenticated encryption for group communications', *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, 2000, **E83-A**, (6), pp. 1228–1237
19 Tseng, Y.M., Jan, J.K., and Chien, H.Y.: 'On the security of generalization of threshold signature and authenticated encryption', *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, 2001, **E84-A**, (10), pp. 2606–2609
20 Hsu, C.L., Wu, T.S., and Wu, T.C.: 'Improvements of threshold signature and authenticated encryption for group communications', *Inf. Proc. Lett.*, 2002, **81**, (1), pp. 41–45
21 Lee, N.Y.: 'The security of the improvement on the generalization of threshold signature and authenticated encryption', *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, 2002, **E85-A**, (10), pp. 2364–2367
22 Lee, N.Y., Hwang, T., and Li, C.M.: '$(t, n)$ threshold untraceable signatures', *J. Inf. Sci. Eng.*, 2000, **16**, (6), pp. 835–845
23 Ohta, K., and Okamoto, T.: 'A modification of the Fiat-Shamir scheme'. Proc. Crypto'88, Santa Barbara, CA, USA, 1988, pp. 232–243
24 Menezes, A.J., van Oorschot, P.C., and Vanstone, S.A.: 'Handbook of applied cryptography' (CRC Press, 1996)
25 Hwang, M.S., Lee, C.C., and Chang, T.Y.: 'Broadcasting cryptosystem in computer networks using geometric properties of lines', *J. Inf. Sci. Eng.*, 2002, **18**, (3), pp. 373–378
26 Lee, C.C., Chang, T.Y., and Hwang, M.S.: 'A simple broadcasting cryptosystem in computer networks using exclusive-or', *Int. J. Comput. Appl. Technol.*, 2003, (to be published)