2002 International Computer Symposium (ICS2002)

(1) Name of the workshop：Workshop on Cryptology and Information Security

(2) Title of the paper：A Specifiable-Verifier Group-Oriented Threshold Signature Scheme Based on the Elliptic Curve Cryptosystem

(3) A short abstract：Aimed at the group-oriented threshold signature, the research is devoted to the specifiable-verifier characteristic in a group-oriented cryptosystem. In light of the characteristic, the group of signers is provided with the limits of authority to specify the group of verifiers. Moreover, the elliptic curve cryptosystem is applied to the integration with the proposed scheme due to the superiority of low-amount operation, so that the performance can be raised to be more efficient than that by the other algorithms.

(4) Name：Tzer-Shyong Chen, Gwo-Shiuan Huang, Yu-Fang Chung, and Nien-Tzu Chang

Current affiliation：Department of Computer Science and Information Engineering, Da-Yeh University

Postal address：Department of Computer Science and Information Engineering, Da-Yeh University, 112 Shan-Jiau Rd, Da-Tsuen, Changhua, Taiwan 515, R.O.C.

E-mail address：r9006013@mail.dyu.edu.tw

Telephone number：0923696355

(5) Name of the contact author：Tzer-Shyong Chen

(6) A list of keywords：threshold signature, elliptic curve cryptosystem, elliptic curve discrete logarithm problem, cryptography

A Specifiable-Verifier Group-Oriented Threshold Signature Scheme Based on the Elliptic Curve Cryptosystem

Tzer-Shyong Chen Gwo-Shiuan Huang* Yu-Fang Chung * Nien-Tzu Chang *

Department of Information management, Tunghai University, Taichung, Taiwan 40744, R.O.C.

* Department of Computer Science and Information Engineering, Da-Yeh University

E-mail: arden@mail.dyu.edu.tw

Abstract

Aimed at the group-oriented threshold signature, the research is devoted to the specifiable-verifier characteristic in a group-oriented cryptosystem. In light of the characteristic, the group of signers is provided with the limits of authority to specify the group of verifiers. Moreover, the elliptic curve cryptosystem is applied to the integration with the proposed scheme due to the superiority of low-amount operation, so that the performance can be raised to be more efficient than that by the other algorithms.

Key words: threshold signature, elliptic curve cryptosystem, elliptic curve discrete logarithm problem, cryptography

1. Introduction

The concept of group-oriented cryptography, initiated by Desmedt [1] in 1987, is devoted to the research for the secure communication between the contrasts of groups. Moreover, the group-orient schemes of further applications are developed into the threshold signature ones. In the case of the application of the perfect secret sharing scheme [2] by Shamir, Harn [3] originated to construct a (*t*, *n*) threshold signature

scheme based on the property of Lagrange Polynomials. The so-called (t, n) threshold signature scheme means that only t members of a n-member group enables to represent the whole group and give the valid signature in the name of the group, in which t is a threshold value located from 1 to n ($1 \leq t \leq n$). In the recent years, lots of related research [4-10] is proposed. However, some of these schemes allow anyone to play the role of verifier for the signature. A (t, n) threshold signature scheme with (k, l) threshold shared verification [11] later is presented by Wang et al. to specify the verifier. In other words, one disables to verify the group signature unless he is the specific verifier. For the scheme by Wang et al., k of a specific l-verifier group enables to act for the verification of group signature, in which k is a threshold value located in the scope from 1 to l($1 \leq k \leq l$). Later in 2002, the scheme is shown to violate the requirements for the (k, l) threshold shared verification by Hsu et al. [12]. That is, an attacker can verify the validity of the group signature alone without the favor of the others in the group of verifiers. Besides, Hsu pointed that the private key of the signer can be easily retrieved from the individual signature for a message. For solving these two secure leaks, an improvement was proposed by Hsu. The improvement inclines the solution to randomly select a number through a system center (SC) so that the above-mentioned weaknesses can be successfully prevented. However, an additional operation through the SC for each generation of individual signature is inefficient for performance. Therefore, a new scheme is proposed in the research to achieve both security and efficiency based on the elliptic curve cryptosystem[13-16], and succeeding in omitting the additional operation by the SC in the generation phase of individual signature.

In the following, the introduction to the scheme by Hsu is briefly discussed for the contrast in Section 2. Section 3 surveys the proposed scheme with special

emphasis on the elliptic curve cryptosystem. Section 4 analyses the security and efficiency of the proposed scheme presented in the previous two Sections. Section 5 furnishes the conclusions.

2. Review of the Scheme by Hsu

In the section, the scheme by Hsu is briefly introduction to the related concepts. The scheme requires for a system center (SC) in charge of the generations of system parameters, individual/group private keys, and individual/group public keys. Firstly, let $G_s = \{u_{s1}, u_{s2}, \ldots, u_{sn}\}$ be denoted as the n-signer group, and $G_v = \{u_{v1}, u_{v2}, \ldots, u_{vl}\}$ be denoted as the *l*-verifier group. Any *t* members of the *n*-signer group enable to give the valid signature for the signer group G_s, and any *k* members of the *l*-verifier group enable to verify the validity of the received group signature for the verifier group G_v. Then, these *t* signers jointly elect a clerk (CLK) from themselves to validate all individual signatures and to combine the *t* valid individual signatures into a group signature. The procedure of performance contains the following three phases: Parameter Generation Phase, Individual Signature Generation and Verification Phase, and Group Signature Generation and Verification Phase.

2.1 Parameter Generation Phase

The SC determine the required parameters and keys according to the following:

(1) two large primes *p* and *q*, where $q \mid p\text{-}1$;

(2) a generator *g* with order *q* over *GF*(*q*);

(3) a one-way hash function *h*;

(4) two secret polynomials $f_s(x) = a_{t-1}x^{t-1} + a_{t-2}x^{t-2} + \cdots + a_1 x + a_0 \bmod q$ and

$f_v(x) = c_{k-1}x^{k-1} + c_{k-2}x^{k-2} + \cdots + c_1 x + c_0 \bmod q$, where $a_i,\ c_j \in [1, q-1]$

for $i = 0,1,2,\ldots,t-1$ and $j = 0,1,2,\ldots,k-1$;

(5) a group private key $f_s(0) = a_0$ and a group public key $Y_s = g^{f_s(0)} \bmod p$ for G_s, and a group private key $f_v(0) = c_0$ and a group public key $Y_v = g^{f_v(0)} \bmod p$ for G_v;

(6) an individual private key $f_s(x_{si})$ and public key $y_{si} = g^{f_s(x_{si})} \bmod p$ for each signer u_{si} in G_s, in which $i = 1,2,\ldots,n$ and x_{si} is the public values associated with each signer u_{si};

(7) an individual private key $f_v(x_{vi})$ and public key $y_{vi} = g^{f_v(x_{vi})} \bmod p$ for each verifier u_{vi} in G_v, in which $i = 1,2,\ldots,l$ and x_{vi} is the public values associated with each signer u_{vi}.

Then, the SC declares the system parameters p, q, g, h, y_{si} (for $i = 1,2,\ldots,n$), y_{vi} (for $i = 1,2,\ldots,l$), Y_s, and Y_v public.

2.2 Individual Signature Generation and Verification Phase

The SC firstly select the required functions and parameters, as follows:

(1) a secret polynomial $f_b(x) = b_{t-1}x^{t-1} + b_{t-2}x^{t-2} + \ldots + b_1 x + b_0 \bmod q$, where $b_i \in [1, q-1]$ for i =0, 1, 2,…, t-1;

(2) a secret value $f_b(0) = b_0$ and a public value $Y_b = g^{f_b(0)} \bmod p$ for G_s;

(3) a secret value $f_b(x_{si})$ and a public value $y_{bi} = g^{f_b(x_{si})} \bmod p$ for each signer u_{si} in G_s, in which $i = 1,2,\ldots,n$.

It is noteworthy that $f_b(0)$, Y_b, $f_b(x_{si})$, and y_{bi} are all random numbers changeable in different time of signature.

Assume that there are t signers ($u_{s1}, u_{s2}, \ldots, u_{st}$). In order to give a valid

signature for a message m, each of u_{si} $(i=1,2,\ldots,t)$ uses his private key $f_s(x_{si})$, the group public key Y_v of G_v, and the random integer $f_b(x_{si})$ to compute the commitment value, as follows:

$$r_{si} = Y_v^{(f_b(x_{si})+f_s(x_{si}))\prod_{j=1,j\neq i}^{t}\frac{0-x_{sj}}{x_{si}-x_{sj}}} \bmod p$$

Then, each of the t signers sends the r_{si} to the other associates via a secure channel. After receiving all r_{si} $(i=1,2,\ldots,t)$, each of u_{si} $(i=1,2,\ldots,t)$ computes r and s_i, as follows:

$$r = \prod_{i=1}^{t} r_{si} \bmod p$$

$$s_i = h(m) f_s(x_{si}) \prod_{j=1,j\neq i}^{t} \frac{0-x_{sj}}{x_{si}-x_{sj}} - r f_b(x_{si}) \prod_{j=1,j\neq i}^{t} \frac{0-x_{sj}}{x_{si}-x_{sj}} \bmod q$$

where r can be regarded as the common session key between the groups of G_s and G_v because $r = \prod_{i=1}^{t} r_{si} = \prod_{i=1}^{t} Y_v^{(f_b(x_{si})\prod_{j=1,j\neq i}^{t}\frac{0-x_{sj}}{x_{si}-x_{sj}}+f_s(x_{si})\prod_{j=1,j\neq i}^{t}\frac{0-x_{sj}}{x_{si}-x_{sj}})} = Y_v^{(f_b(0)+f_s(0))} = g^{f_v(0)f_b(0)} g^{f_v(0)f_s(0)}$

$= Y_b^{f_v(0)} Y_s^{f_v(0)} \pmod p$.

After that, u_{si} sends s_i, which is regarded as the individual signature for m, to the CLK who then verifies the validity of s_i according to the equality:

$$y_{si}^{h(m)\prod_{j=1,j\neq i}^{t}\frac{0-x_{sj}}{x_{si}-x_{sj}}} \stackrel{?}{=} g^{s_i} y_{bi}^{r\prod_{j=1,j\neq i}^{t}\frac{0-x_{sj}}{x_{si}-x_{sj}}} \pmod p$$

If the equality is certifiable, then the individual signature s_i can be verified to be valid.

2.3 Group Signature Generation and Verification Phase

If these t individual signatures are all verified to be valid, the CLK computes the group signature for m, as follows:

$$s = \sum_{i=1}^{t} s_i \bmod q$$

Then, he sends the group signature to the group G_v.

For verifying its validity, each verifier of u_{vi} ($i = 1,2,\ldots,k$) firstly computes as follows:

$$r_{vi} = (Y_s Y_b)^{f_v(x_{vi}) \prod_{j=1, j\neq i}^{k} \frac{0-x_{vj}}{x_{vi}-x_{vj}}} \bmod p$$

Then, every verifier sends it to the other associates via a secure channel. Upon all of the receiving r_{vi} ($i = 1,2,\ldots,k$), each associated verifier computes $r = \prod_{i=1}^{k} r_{vi} \bmod p$. Afterwards, the validity of the group signature s for m can be verified according to the following equality:

$$Y_s^{h(m)} \stackrel{?}{=} g^s Y_b^r \pmod p$$

If the equality can be satisfied, the group signature is valid.

3. The Proposed Scheme

Herein is the introduction to the proposed scheme. The scheme requires for a system center (SC) to execute the generation of the necessary parameters of the system and users. Let a group of n signers be indicated as $G_s = \{u_{s1}, u_{s2}, \ldots, u_{sn}\}$, in which a association of any t members ($1 \le t \le n$) can give a valid group signature for a message in the name of the whole group, and let a group of l verifiers be indicated as $G_v = \{u_{v1}, u_{v2}, \ldots, u_{vl}\}$, in which a association of any k members ($1 \le k \le l$) can verify the validity of the received group signature for the whole verifier group. Then, these t signers jointly elect a clerk (CLK) from themselves to validate all individual signatures and to combine the t valid individual signatures into a group

signature. The procedure of the performance is concluded into three phases: Parameter Generation Phase, Individual Signature Generation and Verification Phase, and Group Signature Generation and Verification Phase.

3.1 Parameter Generation Phase

The SC is responsible for the generation of the required parameters of the system and the keys of the users. The generation phase is as follows:

(1) a field size p, which is a large odd prime;

(2) two field elements a and $b \in F_p$ to define the elliptic curve equation E over F_p, (i.e. $y^2 = x^3 + ax + b \pmod{p}$ where $p > 3$ and $4a^3 + 27b^2 \neq 0 \pmod{p}$);

(3) a finite point $G = (x_g, y_g)$ whose order is a large prime number over $E(F_p)$, where $G \neq O$ (It is because O denotes an infinite point);

(4) the order of $G = q$;

(5) a one-way hash function h;

(6) two secret polynomials $f_s(x) = a_{t-1}x^{t-1} + a_{t-2}x^{t-2} + \cdots + a_1x + a_0 \bmod q$ and $f_v(x) = c_{k-1}x^{k-1} + c_{k-2}x^{k-2} + \cdots + c_1x + c_0 \bmod q$, in which $a_i, c_j \in [1, q-1]$ for $i = 0,1,2,\ldots,t-1$ and $j = 0,1,2,\ldots,k-1$;

(7) a group private key $f_s(0) = a_0$ and a group public key $Y_s = f_s(0)G$ for G_s, and a group private key $f_v(0) = c_0$ and a group public key $Y_v = f_v(0)G$ for G_v;

(8) an individual private key $f_s(x_{si})$ and public key $y_{si} = f_s(x_{si})G$ for each signer u_{si} in G_s, in which $i = 1,2,\ldots,n$ and x_{si} is the public values associated with each signer u_{si};

(9) an individual private key $f_v(x_{vi})$ and a public key $y_{vi} = f_v(x_{vi})G$ for each

verifier u_{vi} in G_v, in which $i = 1,2,\ldots,l$ and x_{vi} is the public values associated with each signer u_{vi}.

Then, the SC declares the system parameters p, E, G, q, h, y_{si} (for $i = 1,2,\ldots,n$), y_{vi} (for $i = 1,2,\ldots,l$), Y_s, and Y_v public.

3.2 Individual Signature Generation and Verification Phase

Assume that there are t signers ($u_{s1}, u_{s2}, \ldots, u_{st}$). In order to give a valid signature for the message m, each of u_{si} ($i = 1,2,\ldots,t$) generates the individual signature, as follows:

Step 1: Randomly select an integer $b_{si} \in [1, q-1]$ to compute $B_{si} = b_{si}G$ in which B_{si} is a point, and sends the B_{si} to the other associates through a broadcast channel;

Step 2: Combine all received B_{si} ($i = 1,2,\ldots,t$) to obtain the B, as follows:

$$B = \sum_{i=1}^{t} B_{si} = (x_b, y_b)$$

Step 3: Compute the following commitment value r_{si} which is a point using the private key $f_s(x_{si})$, the group public key Y_v of G_v, and the random integer b_{si}, then send r_{si} to the other associates through a secure channel;

$$r_{si} = (b_{si} + f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}})Y_v$$

Step 4: Respectively derive the common session key r of G_s and of G_v using all of the received r_{si} ($i = 1,2,\ldots,t$) so as to respectively generating the individual signature s_i which is a point both and send s_i to the CLK, in which

$$r = \sum_{i=1}^{t} r_{si} = (x_r, y_r)$$

$$s_i = x_b h(m) f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} - x_r b_{si} \mod q ,$$

where the session key r is achieved due to the following equality:

$$r = \sum_{i=1}^{t} r_{si} = \sum_{i=1}^{t} b_{si} Y_v + \sum_{i=1}^{t} f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} Y_v = f_v(0)B + f_v(0)Y_s$$

After receiving all individual signatures for the message m, the CLK has to verify the validity of each signature through the following determinant equality:

$$x_b h(m) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} y_{si} \stackrel{?}{=} s_i G + x_r B_{si}$$

If the equality is certifiable, then the individual signature s_i can be verified to be valid.

Theorem 1: If the individual signature indeed results from the valid signer, then the signature verification equality holds.

[Proof]

$$s_i = x_b h(m) f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} - x_r b_{si} \mod q$$

$$\Leftrightarrow \quad s_i G = x_b h(m) f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} G - x_r b_{si} G$$

$$\Leftrightarrow \quad s_i G = x_b h(m) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} y_{si} - x_r B_{si}$$

$$\Leftrightarrow \quad x_b h(m) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} y_{si} = s_i G + x_r B_{si}$$

3.3 Group Signature Generation and Verification Phase

If all of the t individual signatures are shown as valid, the CLK computes the group signature s for the message m so as to sending it to the verifier group G_v, as

follows:

$$s = \sum_{i=1}^{t} s_i \mod q$$

At the same time, the CLK also has to declare the B public. While the verification group G_v intends to verify the received group signature s, any k verifiers can verify it for the whole verifier group. Each verifier of u_{vi} ($i = 1,2,\ldots,k$) computes a commitment value r_{vi} using the private key $f_v(x_{vi})$, public parameter B, and group public key Y_s of G_s, then sends the r_{vi} to the other associates through a secure channel, in which

$$r_{vi} = f_v(x_{vi}) \prod_{j=1, j\neq i}^{k} \frac{0 - x_{vj}}{x_{vi} - x_{vj}} (B + Y_s)$$

For verifying the validity of the group signature for the message m, each associated verifier computes r after receiving all r_{vi} ($i = 1,2,\ldots,k$), as follows:

$$r = \sum_{i=1}^{k} r_{vi} = (x_r, y_r)$$

If the following determinant equality can be certifiable, then the group signature for the message m can be verified to be valid:

$$x_b h(m) Y_s \stackrel{?}{=} sG + x_r B$$

Theorem 2: If the group signature indeed results from the valid signer group, then the signature verification equality holds.

[Proof] $$s_i = x_b h(m) f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} - x_r b_{si} \mod q$$

$$\Leftrightarrow \quad s_i G = x_b h(m) f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} G - x_r b_{si} G$$

$$\Leftrightarrow \sum_{i=1}^{t} s_i G = \sum_{i=1}^{t} (x_b h(m) f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} G) - \sum_{i=1}^{t} (x_r b_{si} G)$$

$$\Leftrightarrow sG = x_b h(m) Y_s - x_r B$$

$$\Leftrightarrow x_b h(m) Y_s = sG + x_r B$$

4. Estimation of Security and Performance

4.1 Analyses of Security

The security of the proposed scheme is based on the difficulty by the elliptic curve discrete logarithm problem (ECDLP). The following are the analyses aimed at the possible attacks and the factors why the proposed scheme enables to overcome.

(1) Plaintext Attack

The so-called plaintext attacks can be formed from different ways, such as the derivation of individual private keys $f_s(x_{si})$ and $f_v(x_{vi})$ using the individual public keys $y_{si} = f_s(x_{si})G$ and $y_{vi} = f_v(x_{vi})G$, and the derivation of the group private keys $f_s(0)$ and $f_v(0)$ using the group public keys $Y_s = f_s(0)G$ and $Y_v = f_v(0)G$. Besides, an attacker can force to derive the signer's private key $f_s(x_{si})$ or the verifier's private key $f_v(x_{vi})$ according to the commitment value $r_{si} = (b_{si} + f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}}) Y_v$ or $r_{vi} = f_v(x_{vi}) \prod_{j=1, j\neq i}^{k} \frac{0 - x_{vj}}{x_{vi} - x_{vj}} (B + Y_s)$. Such kinds of solutions are infeasible under the defense of the ECDLP.

(2) Forgery Attack

Assume that an attacker forge a group signature s to make the following determinant equality certifiable. Such an attack is to randomly select x_r, $h(m)$, and the point $B = (x_b, x_b)$ so as to deriving the value of s which can satisfy the determinant

equality. However, the difficulty of the derivation is concluded to the solution of the ECDLP so it is infeasible.

$$x_b h(m) Y_s \stackrel{?}{=} sG + x_r B$$

(3) Equation Attack

Assume that an attacker intends to derive the private key $f_s(x_{si})$ of the signer through the following the individual signature:

$$s_i = x_b h(m) f_s(x_{si}) \prod_{j=1, j\neq i}^{t} \frac{0 - x_{sj}}{x_{si} - x_{sj}} - x_r b_{si} \mod q$$

Such kinds of solution are infeasible because these two data $f_s(x_{si})$ and b_{si} in the equality are secret and unknown to any others.

4.2 Analyses of Performance

In the scheme [12] by Hsu, whenever the signer group signs a message in the individual signature generation and verification phase, the SC must assign the group with a secret polynomial $f_b(x)$ so as to computing a secret value $f_b(x_{si})$ and a public value $y_{bi} = g^{f_b(x_{si})} \bmod p$. Note that these two values should be different for different time of signature to avoid the individual private key $f_s(x_{si})$ from being easily derived. However, the same phase in the proposed scheme no longer asks for the participation of the SC. In the below, for the convenience to make a comparison, a contrast aimed at the analyses of performance between the scheme by Hsu and that by us is presented.

Table 1 is the definitions of the given notations, and Table 2 shows the comparison of different operations. Then, the required time complexity in the different phases is estimated in Table 3, so that the efficiency in performance can be specifically analyzed.

Table 1: Definitions of Notions

Notations	Definitions
T_{MUL}	the time for the modular multiplication
T_{EXP}	the time for the modular exponentiation
T_{ADD}	the time for the modular addition
T_{INV}	The time for the modular inversion
T_{EC_MUL}	the time for the multiplication of a number and an elliptic curve point
T_{EC_ADD}	the time for the addition of two points in an elliptic curve

According to the following conditions, the time complexity for the different operations can be roughly united into the multiplication operation [17][18]:

- g^x mod p, where p is a 1024-bit prime and x is a random 160-bit integer.
- $k \nearrow B$ is given, where $B \in E(Z_p)$, E is an elliptic curve defined over Z_p, $p \approx 2^{160}$, and k is a random 160-bit integer.

Thus, a comparison between different kinds of operations and multiplication operation is given, as follows:

Table 2: Comparison Between the Other Operation and Multiplication Operation

$T_{EXP} \approx 240T_{MUL}$	$T_{EC\_MUL} \approx 29T_{MUL}$	$T_{EC\_ADD} \approx 0.12T_{MUL}$	T_{ADD} is negligible

Table 3: Estimation of Performance Aimed at Time Complexity

Items	Scheme by Hsu		Scheme by us	
	Time Complexity	Roughly Estimation	Time Complexity	Roughly Estimation
Parameter Generation Phase	$(n+l+2)\ T_{EXP}$	$240(n+l+2)\ T_{MUL}$	$(n+l+2)\ T_{EC\_MUL}$	$29(n+l+2)\ T_{MUL}$

Individual Signature Generation and Verification Phase	$(n+5)\ T_{EXP}$ + $(7t-3)\ T_{MUL}$ + $(6t-4)\ T_{ADD}$ + $(3t-3)\ T_{INV}$ + 2 *Hashing*	$(240n+7t+1197)T_{MUL}$ + $(3t-3)\ T_{INV}$ + 2 *Hashing*	5 T_{EC_MUL} + $(2t-1)\ T_{EC\_ADD}$ + $(6t-2)\ T_{MUL}$ + $(6t-4)\ T_{ADD}$ + $(3t-3)\ T_{INV}$ + 2 *Hashing*	$(6.24t+142.88)T_{MUL}$ + $(3t-3)\ T_{INV}$ + 2 *Hashing*
Group Signature Generation and Verification Phase	4 T_{EXP} + $(3k-1)\ T_{MUL}$ +$(t+2k-3)T_{ADD}$ + $(k-1)\ T_{INV}$ + 1 *Hashing*	$(3k+959)\ T_{MUL}$ + $(k-1)\ T_{INV}$ + 1 *Hashing*	4 T_{EC_MUL} + $(k+1)\ T_{EC\_ADD}$ + $(2k-1)\ T_{MUL}$ + $(t+2k-3)\ T_{ADD}$ + $(k-1)\ T_{INV}$ + 1 *Hashing*	$(2.12k+115.12)T_{MUL}$ + $(k-1)\ T_{INV}$ + 1 *Hashing*

5. Conclusions

The proposed group-oriented threshold signature scheme achieves the ability to specify the verifier group. Except for the specific group, no one enables to verify the group signature. Such a characteristic can be fit to some certain situation. Moreover, the integrated application with the elliptic curve causes the cryptosystem secure and efficient.

6. Acknowledgement

This work was supported partially by National Science Council of Republic of China under Grants NSC 90-2213-E-129-003.

References

[1] Y. Desmedt, *Society and Group Oriented Cryptography: A New Concept*, "Advances in Cryptology, Proc. Of Crypto'87," 1987, pp.120-127.

[2] A. Shamir, *How to Share a Secret*, "Commun. ACM," Vol. 22, 1979, pp.

612-613.

[3] L. Harn, *Group-Oriented* (*t*, *n*) *Threshold Signature and Digital Multisignature*, "IEE Proc.-Comput. Digit. Tech.," Vol. 141, No. 5, 1994, pp. 307-313.

[4] C.C. Chang, J.J. Leu, P.C. Hwang, and W.B. Lee, *A Scheme for Obtaining a Message from the Digital Multisignature*, "International Workshop on Practice and Theory Public Key Cryptography," Springer-Verlag, 1998, pp. 154-163.

[5] L. Harn, *Digital Signature with* (*t*, *n*) *Shared Verification Based on Discrete Logarithms*, "Electron. Lett.," Vol. 29, No. 24, 1993, pp. 2049-2095.

[6] P. Hoster, M. Michels, and H. Peterson, *Comment*: *Digital Signature with* (*t*, *n*) *Shared Verification Based on Discrete Logarithms*, "Electron. Lett.," Vol. 31, No. 14, 1995, pp. 1137.

[7] S.J. Hwang, C.C. Chang, and W.P. Yang, *Authenticated Encryption Schemes with Message Linkage*, "Inf. Process. Lett.," Vol. 58, 1996, pp. 189-194.

[8] W.B. Lee and C.C. Chang, *Comment*: *Digital Signature with* (*t*, *n*) *Shared Verification Based on Discrete Logarithms*, "Electron. Lett.," Vol. 31, No. 3, 1995, pp. 176-177.

[9] W.B. Lee and C.C. Chang, *Authenticated Encryption Scheme Without Using a One-Way Function*, "Electron. Lett.," Vol. 31, No. 19, 1995, pp. 1656-1657.

[10] C.M. Li, T. Hwang, and N.Y. Lee, *Threshold Multisignature Scheme Where Suspected Forgery Implies Tractability of Adversarial Shareholders*, "Advances in Cryptology, Proc. of Eurocrypt '94," 1995, pp. 194-203.

[11] C.T. Wang, C.C. Chang, and C.H. Lin, *Generalization of Threshold Signature and Authenticated Encryption for Group Communications*, "IEICE Trans. Fundamentals," Vol. E83-A, No. 6, 2000, pp. 1228-1237.

[12] C.L. Hsu, T.S. Wu, and T.C. Wu, *Improvements of Generalization of Threshold Signature and Authenticated Encryption for Group Communications,* "Inf. Process. Lett.," Vol. 81, 2002, pp. 41-45.

[13] V.S. Miller, *Uses of Elliptic Curves in Cryptography*, "Advances in Cryptology-CRYPTO'85, Proceedings, Lecture Notes in Compute Science, New York, NY: Springer-Verlag," No. 218, 1985, pp. 417-426.

[14] N. Koblitz, *Elliptic Curve Cryptosystems*, "Mathematics of Computation," Vol. 48, 1987, pp. 203-209.

[15] A.J. Menezes, T. Okamoto, and S.A. Vanstone, *Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field*, "IEEE Transactions on Information Theory," Vol. 39, 1993, pp. 1639-1646.

[16] J.S. Brickell and K.S. McCurely, *ECC: Do We Need to Count?*, "Advances in Cryptology-ASIACRYPT'99, Lecture Notes in Compute Science, Springer-Verlag," No. 1716, 1999, pp. 122-134.

[17] C. Lin and C. Lee, *Elliptic-Curve Undeniable Signature Schemes*, "Proceedings of the Eleventh National Conference on information Security," 2001, pp. 331-338.

[18] Aleksandar Jurisic and Alfred J. Menezes, *Elliptic Curves and Cryptography*, "http://www.certicom.com."