I.J. Information Technology and Computer Science, **2009, 1, 16-22**
Published Online October 2009 in MECS (http://www.mecs-press.org/)

Two Group Signature Schemes with Multiple Strategies Based on Bilinear Pairings

Jianhua Zhu
College of Computer Science & Technology, HuaZhong University of Science & Technology, WuHan, China
E-mail: zhu-jian-hua@tom.com

Guohua Cui, Shiyang Zhou
College of Computer Science & Technology, HuaZhong University of Science & Technology, WuHan, China
E-mail: zshiyang@tom.com

Abstract—**A group signature scheme and a threshold group signature scheme based on Bilinear Paring are proposed, there are multiple security strategies in these two schemes. These schemes have forward security which minimizes the damage caused by the exposure of any group member's signing key, and does not affect the past signatures generated by this member; meanwhile, ahead signature generated by a group member before the joining date can be prevented via this strategy. Moreover, this scheme support the group member revocable function efficiently and further has no requirement for time period limits.**

Index Terms—**group signature, forward security, member revocation, ahead signature, threshold**

I. Introduction

In 1991, group signature was introduced by Chaum and van Heyst[1] . Compare to traditional single signature, in the group signature scheme, a signer can sign a message on the behalf of the group without revealing his identity, receiver can verify the validity of a signature with group public key, verify whether a signature was signed by a group member or not, but don't know who generated this signature. In addition, it is difficult to determine whether two different signatures were generated by a same group member or not. Anyone (even if group manager) can not forge a valid group member's signature. From this work, a variety of the group signature scheme has been proposed with many improvements by some scholars such as J.Camenish and M.Stadler[2], J.Camenish and M.Michels[3], G.Ateniese and G.Tsudik[4 ,5]. Especially, the two group schemes proposed by J.Camenish and M.Stadler in [2] in which the group public key and the length are independent to numbers of group members, are very meaningful for the development of signature technology.

At present, to group signature scheme practicability research, following aspects research continuously attract domestic and foreign scholar's attention widely:

（1）Group member revocation. Once a group member is dropped from group, its signatures will be regarded as invalid. As a practical group signature scheme, it should permit member to join and quit dynamically. In the existing schemes, the technology to join a group effectively is quite mature. But in member's revocation aspect, although many researches have been conducted, the contribution is not quite ideal, and part of schemes has even been proven to contain some mistakes later on.

（2）Forward secure. If a member's signature key was leaked out, then his whole signatures (the past and the future) would be leaked out. A complementary approach is to reduce the potential damage in case secrets are exposed. In what is often called forward security, the main idea is to ensure that secrets are used only for short time periods, and that compromise the effective range of a secret does not affect anything based on secrets from prior time periods. One of the challenges in designing such a system is to be able to change secret information without the inconvenience of changing public information, such as the public key.

（3）Ahead signature prohibition. A group member can only sign a message on the behalf of the group only after he joins that very group, and can not sign a signature of which the signing time is prior to the time he joins group. In a security group scheme, it is very necessary to prohibit ahead signature.

The concept of forward secure signatures was first proposed by Anderson in [6]. Once the secret key was leaked out, forward security scheme can minimize the influence to the system security, the group signature generated previously remained valid and do not need to be re-signed. The commonly used method is to divide system time into T periods, the signing key evolve over time, a group member's signing key can evolves from SK_i to SK_{i+1} using a public one-way function. Initially Anderson used forward security only for traditional single signature, from this work, this concept was applied gradually in group signature scheme [11-13], meanwhile widely applied to other kind of signature scheme [14].

Up to now, although many scholars have made many related research in the forward security for group signature scheme. But almost all schemes are based on the strong RSA, simultaneously are continuous to use the method in [6] that the system time was divided into T periods, while the period evolve over period T, system

Manuscript received January 17, 2009; revised August 21, 2009; accepted September 18, 2009.

has to be initialized and all member's signing keys have to be assigned again. Moreover there is no scheme that can prohibit the ahead signature efficiently.

In Asiacrypt2001, Boneh et al. proposed a short signature scheme [7]. Using the bilinear pairing algorithm based on hyper-elliptic curves, the length of signature was shortened to 170 bits opposite to 1024 bits in the RSA algorithm, 320 bits in DSA algorithm. Security of scheme was founded by Gap Diffie-Hellman[8,9] group. The short signature which has the obvious superiority in the network transmission, attract scholars' attention widely. In recent years, many signature schemes based on the bilinear pairing were proposed, however since DDH is easy to be solved in bilinear pairing, there are few group signature scheme based on the bilinear pairing, more often than not, these schemes are designed for identity-based signature [7,10]. In [15], two group signature schemes based on the bilinear pairing are proposed in which an on-line third party, called security mediator (SEM) is introduced. Unfortunately, in these two schemes, a group member can not verify the signing key received from group manager (GM) and SEM, and there is possibility that SEM can obtain a group member’s signing key by collaborating with any other group member while SEM is no longer credible. Meanwhile, in order to realize the traceability, all signature messages must be stored by SEM, which cause huge memory burden to SEM. What is more, there is a drawback in these two schemes discovered by Cheng et al in [15], in which a dishonest group member may generate a signature to satisfy verification condition, but SEM can not open it correctly[16]. Then Cheng et al proposed a improved scheme which satisfy traceability In [16]. Vo et al proposed a forward security signature scheme from bilinear pairing, however their scheme is a traditional single signature, his design method is not suitable to group signature.

By now, there is no group signature scheme based on bilinear pairing and equip with forward security be proposed. In summary, for the group signature based on bilinear pairing, there are several common drawbacks as follows:

（1）The group member can not verify the signing key received from GM and SEM while he joins the group;

（2）In order to open signatures, SEM has to store all signature messages, that result much memory costs to SEM;

（3）There is no forward security.

In 1989, Desmedt and frankel proposed the concept of threshold group signature through combining group signature and secret sharing together[17] . Following this research, many threshold group schemes were proposed by scholars. In now days some bilinear pairings based threshold group signature schemes were proposed, but there are various flaws in their schemes. In [18], a Chameleon threshold signature based on bilinear pairing was proposed, in this scheme, TTP (Trusted Three Party) can generate any threshold group signature even if there are no sub-signature and scheme is not completeness, there are same designing flaws in scheme proposed in [19]. In [20], a forward secure threshold group signature scheme was proposed in which changing member’s private key will cost huge times from a time period to next and a revoked member can still participate in sign and times cost will become bigger and bigger while time period increment.

In this paper, a group signature scheme based on bilinear pairing and a threshold group signature scheme based on bilinear pairing were proposed. There are multiple security strategies in these two schemes, every member can verify the signing key received from GM and SEM when he joins group, meanwhile, these schemes supports the group member revocation, and have the traceability where SEM has not to store any signature messages. Even if GM or SEM cannot obtain a group member’s signing key by collaborate with any other group member. In the group signature scheme, forward security is proposed for the first time, the scheme has forward security with unlimited time periods, meanwhile, ahead signature generated by a group member can be prevented efficiently. In threshold group signature scheme, there is forward security with no necessary to maintain time periods in first time.

Our paper is organized as follows. Section Ⅱ describes the definition and security requirements of a group signature scheme with forward security. We propose a new forward security group signature scheme with multiple forward security strategies. Section Ⅲ describes the definition and security requirements of a threshold group signature scheme with forward security. We propose a new threshold forward security group signature scheme with multiple forward security strategies, and describe the analysis of security and complexity. Finally we give our conclusion in section Ⅳ.

II. GROUP SIGNATURE

A. Security Group Signature Scheme

A security group signature scheme consists of following procedures.

（1）SETUP. A probabilistic procedure, on input a security parameter, outputs the system parameters, the group public key and the secret key for GM and SEM.

（2）JOIN. When a user want to join the group, the group manager and the user execute a protocol interactively. The user receives the signing key and becomes a new group member.

（3）EVOLVE. Given input of a group member’s signing key for time period i, this procedure outputs the corresponding group member’s signing key for time period i + 1.

（4）SIGN. Given input of a group member’s signing key, a message m and a time period i, this probabilistic procedure outputs a signature on message m.

（5）VERIFY. Given input of a group public key, a group signature on a time period i, a message m, this procedure verifies whether signature is a valid on m signed with a group member’s signing key of time period i.

（6）OPEN. Given input of a valid group signature on the message m, a GM's or SEM's secret key, this procedure determines the identity of the signer for the group signature.

A group signature scheme should satisfy the following security requirements:

（1）Correctness. Signatures produced by a group member using SIGN must be accepted by VERIFY.

（2）Unforgeability. Only group members are able to sign messages on behalf of the group.

（3）Anonymity. Given a valid signature of a message, it is computationally hard for everybody but the GM or SEM to identify the actual signer.

（4）Unlinkability. Unless to open signatures, it is computationally hard for anybody but the GM or SEM to decide whether two different valid signatures were generated by the same group member or not.

（5）Exculpability. Neither a coalition of group members nor the group manager can generate a valid signature that will be opened by the OPEN procedure as generated from another group member.

（6）Traceability. GM or SEM can always open a valid signature using the OPEN procedure and identify the actual signer.

B. The Proposed Scheme

Setup: There are three kind roles in group, first is group administrator (GM), second is a set of group member and third is trusted on-line third party, called a security mediator (SEM). First GM selects $x \in Z_q^*$ as his private key and compute

$$X = xP$$

as his public key, defines two cryptographic hash functions:

$$H_1 : Z_q^* \to G1$$

and

$$H_2 : \{0,1\}^* \to Z_q^* .$$

$\{G_1, G_2, \hat{e}, q, P, Y, H_1, H_2\}$ be published as the group public messages. SEM selects $s \in Z_q^*$ as his private key and computes

$$S = sP$$

as his public key. In initial, GM sets time period to 1.

Join: Suppose that user u_i with identifier $ID_i \in Z_q^*$ is a user who wants to join the group in time period j. GM computes

$$y_i = H_1(ID_i)$$

as u_i 's public key, and computes

$$x_i = x^{-1} y_i ,$$

sends x_i as a signing sub-key to u_i secretly. u_i can believes that the x_i received from GM is a signing sub-key after verified the correctness of x_i by equation

$$\hat{e}(Y, x_i) = H(P, y_i) .$$

Meanwhile SEM selects a random number $e_i \in Z_q^*$ for u_i and saves pair (e_i, y_i) secretly. SEM computes

$$s_{i,j} = s e_i^{-j} y_i$$

and

$$v_{i,j} = e_i^{\,j} P ,$$

then sends $s_{i,j}$ as another signing sub-key and $v_{i,j}$ as verification factor to u_i secretly. u_i can verifies the correctness of $s_{i,j}$ received from SEM by equation:

$$\hat{e}(v_{i,j}, s_{i,j}) = H(S, y_i) .$$

After the correctness of x_i and $s_{i,j}$ are verified, user u_i become a group member and save the pair $(x_i, s_{i,j})$ as his signing key for time period j.

Revoke: In scheme, there is a Certificate Revocation List (CRL) which record the information of revoked group members, the item of CRL is (y_i, t) means a group member with public key y_i was revoked in time period t.

Evolve: While time periods evolve from j to $j+1$, Group member u_i 's signing key $(x_i, s_{i,j})$ be evolved to $(x_i, s_{i,j+1})$ by SEM with equation

$$s_{i,j+1} := e_i^{-1} s_{i,j}$$

and $s_{i,j}$ be destroyed by u_i .

Sign: To generate a group signature on message m in time period j ,Group member u_i selects a random number $k \in Z_q^*$, computes:

$$r_1 = k y_i$$

$$\sigma = k H_2(m \| j) s_{i,j}$$

$$c = k H_2(m \| j) x_i \text{ ,}$$

then sends (y_i, r_1, σ, c, j) to SEM secretly. Firstly, SEM checks whether signer is a valid group member by CRL, then by equation

$$r_1 H_2(m \| j) = s^{-1} e_i^{\,j} \sigma$$

to verify whether $s_{i,j}$ was used to signature, finally, compute

$$r_2 = k' P$$

$$r_3 = s^{-1} e_i r_1$$

$$r_4 = s^2 P + k' (r_1 + r_3 + c)$$

$(r_1, r_2, r_3, r_4, c, j)$ is group member u_i 's signature for message m in time period j .

Verification: To verify the correctness of signature $(r_1, r_2, r_3, r_4, c, j)$ by equations:

$$\hat{e}(P, r_4) = \hat{e}(S, S)\, \hat{e}(r_2, r_1)\, \hat{e}(r_2, r_3)\, \hat{e}(r_2, c) \qquad (1)$$

and

$$\hat{e}(X, c) = \hat{e}(P, H_2(m \| j) r_1) \qquad (2)$$

Through Equation (1) to verify the signature is signed by a valid member, and signing sub-key $s_{i,j}$ was used to signature. Then through equation (2) to verify signing sub-key x_i was used to signature.

Open: In the case of a dispute, SEM has to open a signature (r_1, r_2, r_3, c, j) according the saved (e_i, y_i) . If there is a e_i satisfies equation :

$$s e_i^{-1} r_3 = r_1 ,$$

then signer is y_i .

C. Analysis of Security and Efficiency

In the following, we will show that our proposed scheme satisfies the all security requirements of a forward security group signature.

Correctness： In proposed scheme, when SEM received (y_i, r_1, σ, c, j) from signer, SEM confirms if signer is a valid group member by check the CRL. If there is a (y_k, t) in CRL where $y_i = y_k$ and $j >= t$, SEM refuse (y_i, r_1, σ, c, j) immediately, otherwise SEM continues to verify whether the signing sub-key $s_{i,j}$ was used for signature by equation $r_1 = s^{-1}e_i{}^j\sigma$. This guarantees that (r_1, r_2, r_3, c, j) was generated by a valid group member.

Verifier verifies whether the signature sub-key x_i was used for signature by signer with equation (2). We note that:

$$\begin{aligned}\hat{e}(X,c) &= \hat{e}(xP, kH_2(M)x_i)\\ &= \hat{e}(xP, kH_2(M)x^{-1}y_i)\\ &= \hat{e}(xx^{-1}P, kH_2(M)y_i)\\ &= \hat{e}(P, H_2(M)ky_i)\\ &= \hat{e}(P, H_2(M)r_1)\end{aligned}$$

That the verification of equation (1) and equation (2) means that signature was signed by a valid group member with signing key $(x_i, s_{i,j})$, signature can be accepted.

Anonymity: Given a valid signature (r_1, r_2, r_3, c, j) , it is computationally hard to identify the actual signer because (r_1, r_2, r_3, c) were computed with random number. Therefore anyone except SEM cannot deduce identifier of the signature .

Unlink ability: Given any two group signatures (r_1, r_2, r_3, c, j) and $(r_1', r_2', r_3', c', j')$, they are generated by different random number. According difficulty of DL, it is computationally infeasible to decide whether the two signatures were generated by the same group member or by different group member.

Unforgeability: Suppose that a GM wants to generate a signature, he can choose a random number k ,but because he don't know $s_{i,j}$ or cannot deduce $s_{i,j}$ from $\sigma = ks_{i,j}$ in a feasible algorithm, so he can't generate a (y_i, r_1, σ, c, j) which can pass the verification of equation $r_1 = s^{-1}e_i{}^j\sigma$. Meanwhile, SEM can not forge a signature which can pass the verification of equation (2) in the reason he don't knows x_i .

Coalition-resistance: Our proposed scheme can resists coalition attack efficiently, there is unnecessary to check the relations between group member's public key like the schemes designed in [15] and [16]. Even if there are k group members their public keys satisfy relation $y_1 + y_2 + ... + y_k = y_i \mod q$, this k group members can compute x_i together as follow:

$$\begin{aligned}x_1 + x_2 + ... + x_k &= x^{-1}y_1 + x^{-1}y_2 + ... + x^{-1}y_k\\ &= x^{-1}(y_1 + y_2 + ... + y_k)\\ &= x^{-1}y_i\\ &= x_i\end{aligned}$$

But they cannot compute $s_{i,j}$, So they can only compute: $r_1 = ky_i,\ c = kH_2(m \| j)x_i$ ， cannot compute a valid σ which pass SEM's verification. Therefore these k group members can not forge a valid signature while there is a risk of leak out their own signing sub-key.

Revocability ： In our proposed scheme, every signature is published finally by SEM, SEM verify every signature message received from group members whether the signer's identity is valid, meanwhile verify whether the signing sub-key $s_{i,j}$ was used honestly by equation

$$r_1H_2(m \| j) = s^{-1}e_i{}^j\sigma ,$$

this verification can be used to prevent a dishonest group member forge a untraceable signature, so the r_3 generated by SEM can be regarded as the basis to open the signature.

Traceability ： With the analysis of revocability, the identity of signer can not be forged. Because only SEM knows e_i and (e_i, y_i) , therefore only SEM can open signature by $se_i^{-1}r_3 = r_1$.

Forward security ： While time period evolves from j to $j+1$, group member u_i 's signing key $(x_i, s_{i,j})$ be evolved to $(x_i, s_{i,j+1})$ where $s_{i,j+1} := e_i^{-1}s_{i,j}$,according to the difficulty of DL, besides SEM, every others can not derive $s_{i,j}$ from $s_{i,j+1}$,

Ahead signature prohibit: While u_i joins group in time period j , he receives the signing key $(x_i, s_{i,j})$ from GM and SEM, because he do not knows e_i ,he can not also derives $s_{i,j-1}$ from $s_{i,j}$,so he can not generates a valid signature (r_1, r_2, r_3, c, j') which satisfy $j' < j$.

Efficiency analysis ： By now, there is no group signature scheme with forward security based on bilinear pairing, therefore there is no necessary to provide a compare with other similar schemes. To enhance signing efficiency, SEM can computes s^{-1} and $e_i{}^j$ in advance. With the signing procedure, computing r_1, σ and c cost a point multiplication separately, SEM verify cost a point multiplications and the cost 4 times point multiplications to compute r_2 , r_3 and r_4 ,there are 9 times point multiplications in sum for generating a valid signature, 7 times computations of bilinear map and a hash compute for verify a signature. The length of a group signature is 5|P|.

To realize this scheme in environment: Pentium（R）4 CPU 2.4 GHz + 512M RAM + Windows XP + VC6.0+PBC library. A point multiplication cost 32ms and a cost 64ms. Numbers of group was set to 50. A group signature was generated in 410ms, verification procedure cost 440 ms, opening a group signature cost 1050ms.

III. Threshold Group Signature

A. Security Threshold Group Signature Scheme

A security threshold group signature scheme consists of following procedures.

（1） SETUP. A probabilistic procedure, on input a security parameter, outputs the system parameters, the group public key and the secret key for GM and SEM.

（2） JOIN. When a user want to join the group, the group manager and the user execute a protocol interactively. The user receives the signing key and becomes a new group member.

（3） SIGN. Consist of two steps:

Sub-signature, a probabilistic algorithm, given input of a group member's signing key, a message m, this procedure outputs a sub-signature on message m.

Threshold signature synthesis, a valid algorithm, given input of t members' sub-signature on message m, this procedure outputs a threshold signature on message m.

（4） VERIFY. Given input of a group public key, a threshold group signature, a message m, this procedure verifies whether signature is a valid on m signed with a group member's signing key.

（5） OPEN. Given input of a valid threshold group signature on the message m, a GM's or SEM's secret key, this procedure determines the identity of all t signer for the threshold group signature.

A threshold group signature scheme should satisfy the security requirements as showed in section 2.1.

B. The Proposed Scheme

Setup: There are three kind roles in group, first is group administrator (GM), second is a set of group member and third is trusted on-line third party, called a security mediator (SEM). Firstly GM selects $x \in Z_q^*$ as his private key and compute

$$X = xP$$

as his public key, Defines two cryptographic hash functions:

$$H_1 : \{0,1\}^* \to Z_q^*$$

and

$$H_2 : G_1 \to Z_q ,$$

SEM constructs t-1 degree polynomial:

$$f(x) = a_0 + a_1 x + ... + a_{t-1} x^{t-1} \quad \bmod q$$

where each $a_k \in Z_q$ and computes a check vector:

$$V = (V_0, V_1, \ ... \ V_{t-1})$$

for each coefficient a_k as:

$$V_k = a_k P \quad \bmod q \qquad \mathrm{k} = \ 0,1,...., t\text{-}1 .$$

$\{G_1, G_2, \hat{e}q, P, Y, H_1, H_2, V\}$ be published as the group public messages. SEM selects $s \in Z_q^*$ as his private key and computes

$$S = sP$$

as his public key.

Join: Suppose that user u_i with identifier $ID_i \in Z_q^*$ is a user who wants to join the group. GM computes

$$y_i = f(ID_i)P$$

as u_i 's public key, and computes

$$x_i = xf(ID_i)P ,$$

sends x_i as a signing sub-key to u_i secretly. u_i can believes that the x_i received from GM is a signing sub-key after verified the correctness of x_i by equation:

$$\hat{e}(y_i, X) = \hat{e}(x_i, P)$$

and

$$y_i = \sum_{k=0}^{t-1} (ID_i)^k V_k \ .$$

Meanwhile computes

$$s_i = sy_i$$

then sends s_i as another signing sub-key to u_i secretly. u_i can verifies the correctness of s_i received from SEM by equation

$$\hat{e}(y_i, S) = \hat{e}(s_i, P) \ .$$

After the correctness of x_i and s_i are verified, user u_i become a group member and save the pair (x_i, s_i) as his signing key.

Revoke: In proposed scheme, there is a Certificate Revocation List (CRL) which records the information of revoked group members, the item of CRL is $(ID_i, \tau_{i1}, \tau_{i2})$ means that a group member with identifier ID_i was revoked in time τ_{i2} and all signatures signed by this user are invalid from time τ_{i1} .

Sign: Suppose there is a group $B = \{u_1, u_2, ...u_t\}$ needs to generate a group signature on message m ,every group member u_i where $(i = 1, 2, ...t)$ selects a random number $k \in Z_q^*$, computes:

$$r_i = \hat{e}(P, P)^{k_i} \ \text{,}$$

finally, t members can compute:

$$R = \prod_{i=1}^{t} r_i \ .$$

Member u_i can generates sub-signature:

$$q_i = k_i P + H_1(m \,||\, R) L_i x_i P$$

where

$$L_i = \prod_{j \in B, j \neq i} (0 - ID_j)/(ID_i - ID_j)$$

and sends (m, q_i, y_i) to SEM secretly. SEM checks whether signer is a valid group member by CRL, if the member was cancelled then refuse sub-signature immediately, otherwise SEM checks correctness of sub-signature q_i by equation:

$$\hat{e}(q_i, P) = r_i \cdot \hat{e}(H_1(m \,||\, R) \cdot L_i \cdot y_i, X) \ .$$

As soon as SEM gains t valid sub-signature on message m, Firstly SEM selects $k \in Z_q^*$ randomly and system time stamp T, computes

$$K = kX$$

and

$$Q_1 = \sum_{u_i \in B} q_i \ ;$$

Then SEM generates a identifier vector :

$$ID = (ID[1], ID[2], ..., ID[t])$$

with every signer's identifier, computes:

$$k_i^{'} = H_2(kQ_1 + iP)$$

and modifies ID by equation:

$$ID[i] = k_i^{'} \oplus ID[i] \ .$$

Finally SEM computes:

$$Q_2 = kQ_1 + H_1(ID \,||\, T)s^2P \ ,$$

the threshold signature $(m, R, K, Q_1, Q_2, T, ID)$ on message m be generated.

Verification: To verify the correctness of signature $(m, R, K, Q_1, Q_2, T, ID)$,verifier need to searches CRL firstly, if there is a revoke record $(ID_i, \tau_{i1}, \tau_{i2})$ which satisfies $\tau_{i1} \le T \le \tau_{i2}$,a service provided by SEM are required that SEM computes kQ_1 by

$$Q_2 = kQ_1 + H_1(ID \,||\, T)s^2P \ ,$$

then computes $k_i^{'} = H_2(kQ_1 + iP)$ and can gets signers list by $ID[i] = k_i^{'} \oplus ID[i]$. If there is a revoke record $(ID_i, \tau_{i1}, \tau_{i2})$ which satisfies $ID[j] = ID_i$,SEM return a message YES to verifier, verifier refuses threshold signature, otherwise verifier can verifies the correctness of $(m, R, K, Q_1, Q_2, T, ID)$ by equation:

$$\hat{e}(Q_1, P) = R \cdot \hat{e}(V_0, H_1(m \,||\, R)X) \qquad (3)$$

and

$$\hat{e}(Q_2, P) = \hat{e}(Q_1, K) \cdot \hat{e}(S, H_1(ID \,||\, T)S) \qquad (4)$$

Verifier accept threshold group signature while equation (3) and (4) are correctness, otherwise refuse threshold group signature.

Open: In the case of a dispute, SEM can open a signature $(m, R, K, Q_1, Q_2, T, ID)$ .Firstly SEM can computes kQ_1 with equation $Q_2 = kQ_1 + H_1(ID \,||\, T)s^2P$ by his secret key s ,then computes $k_i^{'} = H_2(kQ_1 + iP)$ where $(i = 1, 2, ...t)$, finally can determines signers list ID by $ID[i] = k_i^{'} \oplus ID[i]$.

C. Analysis of Security and Efficiency

In the following, we will show that our proposed scheme satisfies the all security requirements of a forward security threshold group signature.

Correctness： In proposed threshold group scheme, SEM confirms if signer is a valid group member by check the CRL while SEM received $(m, R, K, Q_1, Q_2, T, ID)$ from signer, then checks if the sub-signature was send from member u_i by equation : $\hat{e}(q_i, P) = r_i \cdot \hat{e}(H_1(m \,||\, R) \cdot L_i \cdot y_i, X)$. SEM generates threshold (m, R, K, Q_1, Q_2, T) when validation of t sub-signature be checked.

The signature $(m, R, K, Q_1, Q_2, T, ID)$ is a valid threshold group signature only if it can pass the verification of equation (3) and (4). We note correctness of equation (3) as follow:

$$\begin{aligned}
\hat{e}(Q_1, P) &= \hat{e}(\textstyle\sum_{u_i \in B} k_iP + H_1(m \,||\, R)L_ix_iP, P) \\
&= \hat{e}(\textstyle\sum_{u_i \in B} k_iP, P) \cdot \hat{e}(H_1(m \,||\, R)xa_0P, P) \\
&= \textstyle\prod_{u_i \in B} \hat{e}(k_iP, P) \cdot \hat{e}(a_0P, H_1(m \,||\, R)xP) \\
&= R \cdot \hat{e}(V_0, H_1(m \,||\, R)X)
\end{aligned}$$

Correctness of equation (3) be showed as follow:

$$\begin{aligned}
\hat{e}(Q_2, P) &= \hat{e}(kQ_1 + H_1(ID \,||\, T)s^2P, P) \\
&= \hat{e}(kQ_1, P)\,\hat{e}(H_1(ID \,||\, T)s^2P, P) \\
&= \hat{e}(Q_1, K) \cdot \hat{e}(S, H_1(ID \,||\, T)S)
\end{aligned}$$

Anonymity: Given a valid threshold group signature $(m, R, K, Q_1, Q_2, T, ID)$, because

$$\begin{aligned}
Q_1 &= \textstyle\sum_{u_i \in B} (k_iP + H_1(m \,||\, R)L_ixP) \\
&= \textstyle\sum_{u_i \in B} k_iP + \sum_{u_i \in B} H_1(m \,||\, R)L_ixP \\
&= \textstyle\sum_{u_i \in B} k_iP + xa_0H_1(m \,||\, R)P
\end{aligned}$$

There are no information of any member in Q_1 ; meanwhile for the other parts of signature, R and ID consist of t random number separately; K and Q_2 were computed by random number and hash function, therefore $(m, R, K, Q_1, Q_2, T, ID)$ is anonymity, anyone except SEM cannot deduce identifier of the signature .

Unlink ability: Given two group threshold signatures $(m, R, K, Q_1, Q_2, T, ID)$ and $(m', R', K', Q_1', Q_2', T', ID')$, they are generated by different random number. According difficulty of DL, it is computationally infeasible to deduce the signers list from a threshold signature, meanwhile it is also in vain to compare same part of $(m, R, K, Q_1, Q_2, T, ID)$ and $(m', R', K', Q_1', Q_2', T', ID')$. Therefore anyone besides SEM cannot decide whether the two different signatures were generated by the same group of members or by different group of members.

Traceability: In proposed threshold group scheme, because threshold group signature be generates by SEM and identifiers of all signers are stored in *ID*, therefore SEM can computes signers list by open algorithm using his secret key. Any others cannot open a threshold group signature.

Coalition-resistance: Our proposed scheme can resists coalition attack efficiently. when t members want to forge a threshold group signature. Although they can compute xa_0P by

$$\begin{aligned}
\textstyle\sum_{u_i \in B} x_iL_i &= \textstyle\sum_{u_i \in B} xf(ID_i)L_iP \\
&= xf(0)P \\
&= xa_0P
\end{aligned}$$

and further forge R, but they cannot compute s^2P ,and cannot deduce s from SEM's public key S and equation $Q_2 = kQ_1 + H_1(ID \,||\, T)s^2P$ due to the difficulty of DL and k is a random number, therefore they cannot forge a valid signature which is untraceable.

Forward security： A member u_i 's information $(ID_i, \tau_{i1}, \tau_{i2})$ would be recorded in CRL if u_i was cancelled, u_i sub-signatures cannot be accepted by SEM no longer and the signatures he participated that signing time after τ_{i1} can pass verification no longer. But all signatures he participated that signing time before τ_{i1} are still valid.

Efficiency analysis： With the threshold signing procedure, generating a sub-signature cost 2 times point multiplications, synthesizing t sub-signatures to a threshold group signature cost 2t times computations of bilinear map and SEM computes K, Q_2 in 3 times point multiplications. In sum there are $2t+3$ times point multiplications and $2t$ times computations of bilinear map for generate a valid threshold signature, 5 times computations of bilinear map for verify a threshold signature. The length of a signature is 5|P|.

To realize this scheme in environment: Pentium （R） 4 CPU 2.4 GHz + 512M RAM + Windows XP + VC6.0+PBC library. The numbers of group was set to 50 and threshold value was set to 5. A valid threshold signature was generated in 1400ms, verification procedure cost 340 ms, opening a threshold signature cost 46ms.

IV. Conclusion

In this paper, A group signature scheme and a threshold group signature scheme based on bilinear paring were proposed. The group signature scheme has forward security while group member's signing key can be evolved with unlimited time periods. The threshold group signature scheme has forward security without necessary to maintain time periods in first time. Two schemes have also the function to prevent ahead signature, a group member can verify the signing key received from group GM and SEM. Anyone Even if GM or SEM cannot forge any valid group member's signature. These two schemes support the group member revocation efficiently and have traceability.

References

[1] D. Chaum, E. van Heyst, Group signatures. In Advances in Cryptology-EuroCrypt'91 , Lecture Notes in Computer Science 547, Springer-Verlag, 1991,pp, 257–265.

[2] J. CAMENISH, M. STADLER, Efficient group signatures for large groups, Proceedings of CRYPTO'97, Lecture Notes in Computer Science 1296, Springer-Verlag, 1997, pp.410–424.

[3] J.CAMENISH, M.A.MICHELS, group signature scheme with improved efficiency, Proceedings of ASIACRYPT'98, Lecture Notes in Computer Science 1541, Springer-Verlag, 1998 , pp,160–174.

[4] G. ATENIESE, G. TSUDIK, A coalition-resistant group signature[EB\OL]. http : //www.isi.edu/∼gts/pubs.html.

[5] G. ATENIESE, G. TSUDIK. Some open issues and new directions in group signatures, Financial Cryptography (FC'99), LNCS 1648, Springer-Verlag, 1999, pp,196–211.

[6] R. Anderson, Invited lecture. In : Proceedings of the 4th ACM Conference on Computer and Communications Security , Zurich , Switzerland , 1997 ,pp, 1–7.

[7] D. Boneh, B. Lynn, H. Shacham, Short signatures from the weil pairing, C Boyd (Ed.), In Asiacrypt'01, Gold Coast , Australia :Springer-Verlag , 2001, pp, 514–532.

[8] D. Boneh, X. Boyen, Short signatures without random oracles, Christian Cachin, Jan Camenisch (Eds.), Eurocrypt'04, Interlaken, Switzerland, Springer-Verlag, 2004, pp,56–73.

[9] D.L. Vo, K. Kim, Yet Another Forward Secure Signature from Bilinear Pairings. ICISC 2005, LNCS 3935, Springer-Verlag, Berlin Heidelberg , 2006, pp.441–455.

[10] Z. Wang, H.Y. Chen, A practical Identity-Based Signature Scheme from Bilinear. EUC Workshops 2007, LNCS 4809, 2007, pp, 704–715.

[11] D.X. Song, Practical forward secure group signature schemes, Proc of the 8th ACM Conf on Computer and Communications Security (CCS 2001), New York : ACM Press , 2001, pp,225–234.

[12] S.Z. Chen, D.X. Li, An efficient revocable group signature schemes with forward security [J]. Chinese Journal of Computers , 2006 , 29 (6) :pp, 998–1003.

[13] R.P.Li, J. Yu and G.W. Li, and D.X. Li, Forward Secure Group Signature Schemes with Efficient Revocation, Journal of Computer Research and Development, 2007, 44 (7) : pp,1219–1226.

[14] X.M.Wang, F.W.Fu and G.Z.Zhan, A Forward Secure Multisignature Scheme, Chinese Journal of Computers , 2004 , 27 (9) , pp,1177–1191.

[15] X. Cheng, H. Zhu, Y. Qiu, and X. Wang, Efficient Group signatures from Bilinear Pairing, CISC'05 LNCS 3822, Springer-Verlag, 2005, pp.128–139.

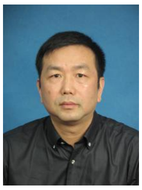
[16] H. Park, H.Kim, K. Chun, and J. Lee, Untraceability of Group Signature Scheme based on Bilinear Pairing and their Improvement, International Conference on Information Technology(ITNG'07) IEEE, 2007, pp,103–109.

[17] Y. Desmedt, Y. Frankel. Threshold cryptosystems. In:Brassard G ed. A dvances in Cryptology, CRYPTO'89 Proceedings. LNCS 435, Berlin: Springer-Verlag, 1990, 307–315.

[18] C.B.Ma,D.K.He, A New Chameleon Threshold Signature Based on Bilinear Pairing. Journal of Computer Research and Developmnt, 2005,42(8): 1427–1430.

[19] F.Xiao,F. G.Chen, D.Y.Zhang. New ID-based Threshhold Signature Scheme from Bilinear Pairings. INDOCRYPT 2004,LNCS 3348, 371–383.

[20] H.X.Peng,D.G.Feng. A Forward Secure Threshold Signature Scheme from Bilinear Pairing. Journal of Computer Research and Development,2007,44(4): 574–580.



Jianhua Zhu, born in 1962. PH.D. in information security in 2009, vice professor and postgraduate supervisor in Department Information Security, College of Computer Science & Technology, HuaZhong University of Science & Technology. His main research interests are cryptography and network security.

Guohua Cui, born in 1946. professor and Ph.D. supervisor in Department Information Security, College of Computer Science & Technology, HuaZhong University of Science & Technology. His main research interests are cryptography and network security.

Shiyang Zhou, born in 1960. vice professor and postgraduate supervisor in Department Science, College of Computer Science & Technology, HuaZhong University of Science & Technology. His main research interests are algorithm design and network security.