一种有效且安全的动态群签名方案*

何业锋[1,2+], 张建中[2]

[1](西安邮电学院 信息与控制系,陕西 西安 710061)

[2](陕西师范大学 数学与信息科学学院,陕西 西安 710062)

An Efficient and Secure Dynamic Group Signature Scheme

HE Ye-Feng[1,2+], ZHANG Jian-Zhong[2]

[1](Department of Information and Control, Xi'an Institute of Posts & Telecoms, Xi'an 710061, China)

[2](College of Mathematics and Information Science, Shaanxi Normal University, Xi'an 710062, China)

+ Corresponding author: E-mail: ye_fenghe@yahoo.com.cn

Received 2003-09-23; Accepted 2003-12-08

He YF, Zhang JZ. An efficient and secure dynamic group signature scheme. *Journal of Software*, 2005,16(4): 609−615. DOI: 10.1360/jos160609

Abstract: To improve the efficiency of group signature, an efficient and secure dynamic group signature scheme is proposed based on the ELGamal encryption and signature of knowledge. It allows the group manager to add new members or delete old members freely. Furthermore, the length of the signature and the computational effort for signing and verifying are independent of the number of the group members and the deleted group members. So it may have many practical applications in *e*-commerce and military.

Key words: signature of knowledge; *e*-th root of discrete logarithm; group signature; dynamic group signature; coalition-resistance

摘 要: 为了提高群签名的效率,利用 ELGamal 加密和知识签名提出了一个有效且安全的动态群签名方案.该方案可以灵活地增加和删除群成员.并且签名长度以及签名与验证的工作量均独立于群成员与已删除群成员的人数.

关键词: 知识签名;离散对数的 e 次根;群签名;动态群签名;抵抗联合攻击

中图法分类号: TP309 **文献标识码**: A

Group signature is introduced by Chaum and Van Heyst[1]. A group signature scheme allows any member of a group to sign messages on behalf of the group. Signatures can be verified with respect to a single group public key,

* Supported by the National Natural Science Foundation of China under Grant No.10271069 (国家自然科学基金); the Natural Science Research Program of Shaanxi Province of China under Grant No.2002A03 (陕西省自然科学研究计划); the Open Subject for Computer Network and Information Security Key Laboratory of Ministry of Education of China (计算机网络与信息安全教育部重点实验室开放课题资助项目)

HE Ye-Feng was born in 1978. She is a master candidate at Shaanxi Normal University. Her current research areas are cryptology and information security. **ZHANG Jian-Zhong** was born in 1960. He is a professor and master supervisor at Shaanxi Normal University. His research areas are cryptology and security of computer communication network.

but they do not reveal the identity of the signer. Furthermore, it is not possible to decide whether two signatures have been issued by the same group member, However, there exists a designated group manager who can, in case of a later dispute, open signatures and reveal the identity of the signer.

The salient features of group signatures make them attractive for many specialized applications such as voting and bidding. More generally, group signatures can be used to conceal organizational structures.

After introduction of the concept of group signature, many group signature schemes[2−4] have been proposed. But all of them can not revoke a group member's ability to sign messages freely. A group signature which can allow the group manager to add new members or delete old members is defined as a dynamic group signature, which is introduced firstly by Chen and Pedersen[5]. They also give a corresponding scheme, however, in which the length of the secret keys of each member and the group manager is linear in the size of the group. Later people give some delete protocols[6,7] for some group signatures[3,4] so that they become dynamic. Unfortunately the delete protocols also increase the length of signatures and computational effort for signing and verifying. Furthermore, the length of a signature and / or the computational effort for signing and verifying depends on the number of the deleted group members. So all of them are not efficient.

In this paper, we present an efficient and secure dynamic group signature scheme, which allows the group manager to add new members or delete group members freely. The length of a signature and the computational effort for signing and verifying do not depend on the number of the deleted group members. So our new group signature scheme is more efficient than the schemes in Refs.[5−7]. It may have many practical applications in *e*-commerce as well as in military.

1 Dynamic Group Signatures

We begin with a definition of dynamic group signatures.

Definition 1. A dynamic group signature scheme is a digital signature scheme comprised of the following procedures:

SETUP: an algorithm for generating the initial group public key Y, a dynamic parameter, and a group manager's secret key S.

JOIN: a protocol between the group manager and a user, which makes the user become a new group member. The user's output is a membership certificate and a corresponding secret key.

DELETE: a protocol between the group manager and group members that results in a group member's ability to sign messages being revoked.

SIGN: an algorithm that, on the input of a group public key, a membership certificate, a secret key, and a message m, outputs a group signature of m.

VERIFY: an algorithm for establishing the validity of an alleged group signature of a message with respect to a group public key.

OPEN: an algorithm that, given a message, a valid group signature on it, a group public key, and a group manager's secret key, determines the identity of the signer.

A dynamic group signature scheme must satisfy the following security properties:

Unforgeability: Only group members are able to sign messages on behalf of the group.

Anonymity: Given a group signature, identifying the actual signer is computationally hard for everyone but the group manager.

Unlinkability: Deciding whether two different group signatures are computed by the same group member is computationally hard.

Exculpability: Neither a group member nor the group manager can sign on behalf of other group members.

Traceability: The group manager is always able to open a valid group signature and identify the actual signer, moreover, a signer can not prevent the opening of a valid group signature.

Coalition-resistance: A colluding subset of group members can not generate valid group signatures that can not be traced.

Revocability: A group signature produced using SIGN by a deleted group member must be rejected using a VERIFY.

2 Preliminaries and Techniques

After giving the notation, this section introduces the building blocks necessary in the subsequent design of our group signature scheme.

2.1 Notations

The symbol || denotes the concatenation of two (binary) strings (or of binary representations of integers and group elements) and ‘ ’ denotes the empty string. If A is a set, $a\in_R A$ means that a is chosen at random from A according to the uniform distribution. Finally, we assume a collision resistant hash function H: $\{0,1\}\rightarrow\{0,1\}^k$ ($k\approx160$).

2.2 Signature of knowledge

Let $G=\langle g\rangle$ be a cyclic group of order n. The discrete logarithm of $y\in G$ to the base g is the smallest positive integer x satisfying $g^x=y$.

An e-th root of the discrete logarithm of $y\in G$ to the base g is an integer x satisfying $g^{(x^e)}=y$ if such an x exists.

The concept of signature of knowledge is introduced by Camenisch[2]. Informally, the signature of knowledge can be seen as an non-interactive proof of knowledge.

The first primitive we define is a signature of the knowledge of the discrete logarithm of y to the base g. It is basically Schnorr Signatureon[8], a message m of the entity knowing the discrete logarithm of y.

Definition 2. A pair $(c,s)\in\{0,1\}^k\times Z_n^*$ satisfying $c=H(m\|y\|g\|g^s y^c)$ is a signature of knowledge of the discrete logarithm of the element $y\in G$ to the base g on the message m. It is denoted by $SPK\{\alpha: y=g^\alpha\}(m)$ (see Ref.[3]).

Such a signature can be computed if the secret key $x=\log_g y$ is known, by choosing r at random from Z_n^* and computing c and S according to $c=H(m\|y\|g\|g^r)$ and $s=(r-cx)(\mathrm{mod}\,n)$.

Definition 3. A pair $(c,s)\in\{0,1\}^k\times Z_n^*$ satisfying $c=H(m\|z\|y\|h\|g\|h^s z^c\|g^s y^c)$ is a signature of equality of the discrete logarithm of the group element z with respect to the base h and the discrete logarithm of the group element y with respect to the base g for the message m. It is denoted by $SPK\{\alpha: z=h^\alpha\wedge y=g^\alpha\}(m)$ (see Ref.[3]).

Such a signature can also be computed if the secret key $x=\log_g y=\log_h z$ is known, by choosing r at random from Z_n^* and computing c and S according to $c=H(m\|z\|y\|h\|g\|h^r\|g^r)$ and $s=(r-cx)\pmod n$.

Definition 4. A 3-tuple $(c,s_1,s_2)\in\{0,1\}^k\times Z_n^{*2}$ satisfying $c=H(m\|y\|z_1\|z_2\|g\|h\|h_1\|g_1\|y^c g^{s_1}h^{s_2}\|z_1^c h_1^{s_2}\|z_2^c g_1^{s_1})$ is a signature of the knowledge on the message m. The knowledge includes the h-part of the representation of the element y to the bases g and h equaling the discrete logarithm of the element z_1 to the base h_1, and the g-part of the representation of the element y to the bases g and h equaling the discrete logarithm of the element z_2 to the bases g_1. It is denoted by $SPK\{(\alpha,\beta): y=g^\alpha h^\beta\wedge z_1=h_1^\beta\wedge z_2=g_1^\alpha\}(m)$ (see Ref.[3]).

Such a signature can also be computed if the secret key (x_1,x_2) satisfying $y=g^{x_1}h^{x_2}, z_1=h_1^{x_2}$ and $z_2=g_1^{x_1}$ is known, by choosing $r_1,r_2 \in_R Z_n^*$ and computing c and s_i according to $c=H(m\|y\|z_1\|z_2\|g\|h\|h_1\|g_1\|g^{r_1}h^{r_2}\|h_1^{r_2}\|g_1^{r_1})$ and $s_i=(r_i-cx_i)(\mathrm{mod}\, n)\ 1\le i\le 2$.

Definition 5. Let $l\le k$ be a security parameter. An $(l+1)$-tuple $(c,s_1,...,s_l)\in\{0,1\}^k\times Z_n^{*l}$ satisfying the equation.

$$c=H(m\|y\|g\|e\|t_1\|...\|t_l) \quad \text{with} \quad t_i=\begin{cases} g^{(s_i^e)} & \text{if } c[i]=0 \\ y^{(s_i^e)} & \text{otherwise} \end{cases}$$

is a signature of knowledge of an *e*-th root of the discrete logarithm of *y* to the base *g*, and is denoted by $SKROOTLOG\{\alpha: y=g^{\alpha^e}\}(m)$ (see Ref.[3]).

Such a signature can be computed if the *e*-th root *x* of the discrete logarithm of *y* to the base *g* is known. One first computes the values $t_i^*=g^{(r_i^e)}$ for $i=1,2,...,l$ by randomly choosing $r_i\in Z_n^*$. Then, c is set to $H(m\|y\|g\|e\|t_1^*\|...\|t_l^*)$, and finally

$$s_i=\begin{cases} r_i & \text{if } c[i]=0 \\ r_i/x(\mathrm{mod}\, n) & \text{otherwise} \end{cases}$$

for $i=1,...,l$. It can easily be seen that the resulting tuple $(c,s_1,...,s_l)$ satisfies the verification equation.

Definition 6. An efficient signature of knowledge of the *e*-th root of the *g*-part of a representation of *y* to the bases *h* and *g*, denoted by E-SKROOTREP $\{(\alpha,\beta): y=h^\alpha g^{\beta^e}\}(m)$, consists of an $(e-1)$-tuple $(y_1,...,y_{e-1})\in G^{e-1}$ and of a signature of knowledge

$$U=SPK\{(\gamma_1,...,\gamma_e,\delta): y_1=h^{\gamma_1}g^{\delta}\wedge y_2=h^{\gamma_2}y_1^{\delta}\wedge...\wedge y_{e-1}=h^{\gamma_{e-1}}y_{e-2}^{\delta}\wedge\ y=h^{\gamma_e}y_{e-1}^{\delta}\}(m).$$

The signature of knowledge can be verified by checking the correctness of *U* (see Ref.[3]).

The following equation explains why a verifier will be convinced of the prover's knowledge of (α,β):

$$y=h^{\gamma_e}(h^{\gamma_{e-1}}(...h^{\gamma_2}(h^{\gamma_1}g^{\delta})^{\delta}...)^{\delta})^{\delta}=h^{\gamma_e+\gamma_{e-1}\delta+...+\gamma_2\delta^{e-2}+\gamma_1\delta^{e-1}}g^{\delta^e}=h^{\alpha}g^{\beta^e}.$$

Such a signature can be computed if values *r* and *x* in Z_n are known for which $y=h^r g^{x^e}$: one first computes the values $y_i=h^{r_i}g^{x^i}$ for $i=1,...,e-1$ with randomly chosen $r_i\in Z_n$, then the signature of knowledge *U* is computed.

3 Construction of the Dynamic Group Signature Scheme

Based on the above signatures of knowledge, we propose an efficient and secure dynamic group signature scheme. The following is the details of construction.

SETUP

The group manager computes the following values:

a RSA module $n=p_1q_1$, p_1 and q_1 are two large primes. Z_p is a field and $n\,|\,p-1$.

an integer $e>1$, e is a relative prime to $\phi(n)$.

two integers $a_1,a_2>1$ whose *e*-th roots can not be computed without knowing the factorization of *n*.

a cyclic group $G=\langle g\rangle$ of order *n* in which computing discrete logarithms is infeasible.

a public key $y_R=h^\rho$ for a randomly chosen value $\rho\in Z_n$.

a dynamic parameter $a_T\in_R Z_n^*$, for time *T*.

The group's public key is $Y=(n,e,a_1,a_2,G,g,y_R)$ and the group manager's secret key is $S=(p_1,q_1,\rho)$. The group public key *Y* and the dynamic parameter (a_T,T) are put in the public notation board which can only be modified or renewed by the group manager.

JOIN

To become a group member, Alice computes her membership certificate as follows:

1) Alice chooses $x \in_R Z_n^*$ as her secret key, and computes $y = g^x \pmod p$ as her public key.

2) To prevent the group manager from learning x, this certificate must be issued using the blind RSA-signature scheme of Chaum[8].

(1) Alice computes

$\tilde{x} = r^e(a_1x^2 + a_2)(\mathrm{mod}n), r \in_R Z_n^*$

$\tilde{y} = g^{x^2} = y^x(\mathrm{mod}p)$

$U = SPK\{\alpha : y = g^\alpha \wedge \tilde{y} = y^\alpha\}$ (‘’)

$V = SKROOTLOG\{\beta : g^{\tilde{x}} = (\tilde{y}^{a_1}g^{a_2})^{\beta^e}\}$ (‘’)

then Alice sends $\tilde{x}, \tilde{y}, U$ and V to the manager.

(2) Checking the correctness of the signatures of knowledge U and V, if they are correct, the manager computes:

$\tilde{v} = \tilde{x}^{1/e} \pmod n$

$(a_T)^{1/e} \pmod n$

$\tilde{v}_T = \tilde{v} \cdot (a_T)^{1/e} = (a_T\tilde{x})^{1/e} \pmod n$

and sends $\tilde{v}_T$ to Alice.

(3) Alice unbinds and thereby obtains her membership certificate:

$v_T = \tilde{v}_T / r = [a_T(a_1x^2 + a_2)]^{1/e} \pmod n$.

Let us now explain what the signatures of knowledge U and V actually mean. The signature U shows that Alice knows the discrete logarithm x of y to the base g, and the discrete logarithm of $\tilde{y}$ to the base y is equal to x. The signature assures that $\tilde{x} = r^e(a_1x^2 + a_2) \pmod n$ holds for the coming r Alice knows, and therefore the group manager can conclude that $\tilde{x}$ is correctly blinded in the secret key.

DELETE

If the manager wants to revoke Bob’s ability to sign messages at time T', he chooses $a_{T'}$ randomly as a dynamic parameter after time T' and puts $(a_{T'}, T')$in the public notation board. Thus the manager can compute the blind certificates for all the members of the group except Bob. For example, he computes the blind certificate for Alice.

$\tilde{v}_{T'} = \tilde{v}(a_{T'})^{1/e} = (a_{T'}\tilde{x})^{1/e} \pmod n$.

He sends $\tilde{v}_{T'}$ to Alice, but he sends nothing to Bob.

Similarly, Alice can compute her membership certificate:

$v_{T'} = \tilde{v}_{T'} / r = [a_{T'}(a_1x^2 + a_2)]^{1/e} \pmod n$.

So Bob’s ability to sign messages is revoked because he has not a corresponding membership certificate after time T'.

SIGN

To sign a message m, Alice computes the following values:

$A = yy_R^z = g^x y_R^z \pmod p$, $z \in_R Z_n^*$

$B = h^z \pmod p$

$C = g^{x^2} y_R^{xz} \pmod p$

$V_1 = SPK\{(\alpha, \beta) : A = g^\alpha y_R^\beta \wedge B = h^\beta \wedge C = A^\alpha\}(m)$

$V_2 = E - SKROOTLOG\{(\gamma, \delta) : (C^{a_1}g^{a_2})^{a_{T'}} = y_R^\gamma g^{\delta^e}\}(m)$

The resulting signature on the message consists of (A, B, C, V_1, V_2).

VERIFY

The signature (A, B, C, V_1, V_2) is valid if the two signatures of knowledge V_1 and V_2 are correct.

The following explains briefly why such a signature convinces a verifier that the signer knows a membership certificate and the corresponding secret key. Consider the signature V_1: it 'Proves' that the signer knows her secret key x and the pair (A,B) is an ELGamal encryption[8] of g^x encrypted under the group manager's public key y_R. It also 'proves' that the signer's secret key x is equal to the discrete logarithm of C to the base A. Similarly, the signature V_2 guarantees that the signer knows her membership certificate $v_{T'}$ because of

$$(C^{a_1}g^{a_2})^{a_{T'}}=[(g^{x^2}y_R^{xz})^{a_1}g^{a_2}]^{a_{T'}}=y_R^{(a_{T'}\cdot a_1xz)}\cdot g^{a_{T'}(a_1x^2+a_2)}=y_R^{(a_{T'}\cdot a_1xz)}\cdot g^{v_{T'}^e} \pmod p .$$

OPEN

In the case of a dispute, the manager can recover the signer's public key $y=A/B^{\rho} \pmod p$ and may also give the signature of knowledge $\overline{U}=SPK\{\alpha : A/y=B^{\alpha}\wedge y_R=h^{\alpha}\}$ (' ') as the proof.

4 Security of the Proposed Scheme

Our scheme is secure and coalition-resistant under the RSA and decisional Diffie-Hellman assumptions[4]. The security of the non-interactive variant, i.e. the group signature scheme, relies additionally on the random oracle model.

Unforgeability: Only group members are able to sign messages on behalf of the group because the group signature scheme is a statistical zero-knowledge (honest-verifier) proof of knowledge of the membership certificate and corresponding secret key, if we assume that hash function H behaves as a random function.

Anonymity: We know that the signer's public key is encrypted and the underlying signatures of knowledge are statistically zero-knowledge, and no information about y is statistically revealed by (A,B,C,V_1,V_2) in the random oracle model. So it is computationally hard for everyone except the group manager to identify the actual signer.

Unlinkability: Deciding if two signatures (A,B,C,V_1,V_2) and (A',B',C',V_1',V_2') are computed by the same group member is computationally hard because the signer's public key is encrypted randomly. Similarly as for anonymity, the problem of linking two signatures reduces to decide whether the discrete logarithms $\log_{y_R}(A/A')$ and $\log_h(B/B')$ are equal. This is, however, impossible under the decisional Diffie-Hellman assumption[4].

Exculpability: Neither a group member nor the group manager can sign on behalf of other group members, because all of them don't know other group member's secret key due to the security of blind RSA-signature scheme of Chaum and ELGamal encryption.

Traceability: From the "open" algorithm, we know that the group manager is able to open any valid group signature and provably identify the actual signer.

Coalition-Resistance: A colluding subset of group members can not generate a valid signature that can not be traced because only the group manager can issue the group member's membership certificates.

Revocability: A deleted group member can not issue a valid group signature again because he has not a corresponding membership certificate after time T'.

Furthermore, the signatures issued by a deleted group member before he is deleted are still anonymous and unlinkable because the DELETE protocol reveals no information about the deleted member.

5 Efficiency of the Proposed Scheme

Our group signature scheme allows the group manager to add new members or delete group members freely. Furthermore, the length of a signature and the computational effort for signing and verifying do not depend on the number of the deleted group members. So our scheme is more efficient.

6 Conclusions

This paper presents a dynamic group signature scheme which is more efficient than the previous dynamic group signature schemes.

References:

[1] Chaum D, Van Heyst E. Group signatures. In: Davies DW, ed. Advances in Cryptology—EUROCRYPT'91. New York: Springer-Verlag, 1997. 257−265.

[2] Camenisch J. Efficient and generalized group signatures. In: Fumy W, ed. Advances in Cryptology—EUROCRYPT'97. New York: Springer-Verlag, 1997. 465−479.

[3] Camenisch J, Stadler M. Efficient group signature schemes for large groups. In: Kaliski B, ed. Advances in Cryptology—CRYPTO'97. New York: Springer-Verlag, 1997. 410−424.

[4] Atenitse G, Clannish J, Joye M, Tsudik G. A practical and provably secure coalition-resistant group signature scheme. In: Bellaire M, ed. Advances in Cryptology—CRYPTO'00. New York: Springer-Verlag, 2000. 255−270.

[5] Chen L, Pedersen TP. Group signatures: unconditional security for members [Ph.D. Thesis]. Denmark: Aarhus University, 1996.

[6] Bresson E, Steren J. Efficient revocation in group signatures. In: Kim K, ed. Public Key Cryptology (PKC2001). New York: Springer-Verlag, 2001. 190−206.

[7] Song D. Practical forward-secure group signature schemes, 2001-11-05. http://www.ece.cmu.edu/~dawnsong/papers/grpsig.pdf.

[8] Schneier B. Applied Cryptography: Protocol, Algorithm, and Source Code in C. 2nd ed., New York: John Wiley and Sons, 1994. 334−340.