Repairing Efficient Threshold Group Signature Scheme

Zuhua Shao
Department of Computer and Electronic Engineering, Zhejiang University of Science and Technology
No. 318, LiuHe Road, Hangzhou, Zhejiang, 310023, P. R. of China
(Email: zhshao_98@yahoo.com)
(Received July 15, 2006; revised and accepted Apr. 21, 2007)

Abstract

To enhance the efficiency of threshold group signature schemes, Yu and Chen, recently, proposed an efficient threshold group signature scheme. By using elliptic curves, the proposed scheme can use short secret key and reduces the load of signature verification. However, in this paper we find that there are many ambiguities in the proposed scheme. The verifiers cannot verify valid signatures, while adversaries can not only easily forge the signatures of individual members, but also forge group signatures without the knowledge of secret keys. Though we can modify it to withstand this forgery attack, the modified scheme cannot withstand a general coalition attack inherent in many threshold signature schemes.

Keywords: Coalition attack, forgery attack, threshold proxy signature, threshold secret share scheme

1 Introduction

Globalization of Internet has accelerated the exchange of electronic information on both the personal and business levels. E-government, online tax filing and electronic banking are important areas for developments. For electronic political and commercial applications, evidence of possession of documents is especially important. A digital signature is analogous to an ordinary hand-written signature and establishes both of sender authenticity and data integrity assurance. At present, there are two most popular public-key algorithms which can provide digital signatures: One is the RSA-type signature scheme [8], the security of which is based on factoring; the other is the ElGamal-type signature scheme [3], the security of which is based on the discrete logarithm problem over the finite field $GF(p)$. However, the length of the RSA signature is too long and the verification of the ElGamal signature requires too more computations.

However, elliptic curve cryptosystems ECC [1, 5, 6, 7], which has a smaller secret key and similar level of security to other cryptosystems, are very appealing. Hence, ECC is already becoming a hot research area.

The concept of group-oriented cryptography, introduced by Desmedt [2] in 1987, plays an important role in the modern society. A group-oriented signature scheme is a method which allows a group to decide its signing policy in such a way that only the authorized subsets of this group can cooperate to sign messages. If the authorized subsets are any set of t members of this group, then it is called a threshold signature scheme. That is, a (t, n) threshold signature scheme allows any t or more signers of the group to cooperatively sign messages on behalf of the group, but $t-1$ or fewer signers cannot [4, 12].

Recently, Yu and Chen [11], by using elliptic curves, proposed a (t, n) threshold group scheme that can use short secret key and reduces the load of signature verification. However, in this paper we find that there are many ambiguities in the proposed scheme. The verifiers cannot verify valid signatures, while adversaries can not only easily forge the signatures of individual members, but also forge group signatures without the knowledge of secret keys.

We also modify the Yu-Chen threshold signature scheme against this forgery attack. However, the modified scheme cannot withstand a general coalition attack inherent in many threshold signature schemes.

This paper is organized as follows: Section 2 briefly reviews the Yu-Chen threshold group signature scheme, Section 3 discusses the security of the signature scheme, and Section 4 proposes a modification. We conclude this paper in Section 5.

2 Brief Review of the Yu-Chen Threshold Signature Scheme

The Yu-Chen (t, n) threshold signature scheme is over an elliptic curve. For the sake of brevity, we omit the introduction of the elliptic curve.

The proposed scheme consists of three phases: the key generation phase, the signature generation phase and the signature verification phase.

The CA (Center Authority) is responsible for generating the system parameters, a secretary authenticates the

signature of each member and publicizes the group signatures, and any recipient can verify the group signatures.

2.1 Key Generation

The CA generates and publicizes system parameters, group public key, individual public keys, and retains the threshold function. The system parameter generation process can be divided into the following steps:

Step 1. The CA generates and publicizes the following system parameters:

E**:** $y^2 = x^3 + ax + b(\text{mod } p)$ represents an elliptic curve, where $a, b \in Z_p, 4a^3 + 27b^2 \neq 0(\text{mod } p)$.

p**:** A large prime number, such that $GF(p) = \{0, \ldots, p-1\}$.

N**:** A large prime number which is the order of the elliptic curve cryptosystem, where $\#E(GF(p))$ lies between $p + 1 - 2\sqrt{p}$ and $p + 1 + 2\sqrt{p}$.

$H(\cdot)$**:** A one-way hash function.

G**:** Base point with order n, representing a base point $G \in E(GF(p))$ on the elliptic curve cryptosystem.

x_i**:** The public identity of group members U_i.

Step 2. The CA generates and retains the following system parameters:

$f(x)$**:** (t, n) threshold function. $f(x) = a_{t-1}x^{t-1} + \ldots + a_1x + a_0(\text{mod } n)$. a_i is a random integer between 1 and $n-1, i = 0, \ldots, t-1$.

$f(0) = a_0$**:** the group secret key.

$f(x_i)$**:** Secret key of individual group member U_i.

Step 3. The CA calculates and publicizes group public key N:

$$Y = f(0)G,$$
$$N = -Y.$$

Step 4. The CA calculates and publicizes individual public key N_i:

$$Y_i = f(x_i)G,$$
$$N_i = -Y_i.$$

2.2 Threshold Digital Signature Generation

Suppose that there is a group that needs to sign a message, the members $U_1, U_2, \ldots, U_t$ can represent the group by signing message m. This stage requires the generation of the individual digital signatures, verification of individual signatures and generation of (t, n) threshold signature. The stages are as follows:

Step 1. Each member U_i uses their secret key $f(x_i)$ and a random integer $k_i, 1 \leq k_i \leq n-1$ to calculate their signature (r_i, s_i) for message m.

$$R_i = (x_{R_i}, y_{R_i}) = k_iG, \text{ publicize } R_i,$$
$$r_i = x_{R_i}(\text{mod } n),$$
$$r_is_i = k_i + f(x_i)h(m)[\prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}](\text{mod } n).$$

$h(m)$ represents the message calculation using a one-way hash function that improves the system security. Member U_i sends his individual digital signature (r_i, s_i) to the secretary.

Step 2. On receiving the digital signature (r_i, s_i) of all members U_i, the secretary employs the following equation to confirm the validity of the signature.

$$D_i = (x_{D_i}, y_{D_i})$$
$$= r_is_iG + h(m)[\prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}]N_i,$$
$$d_i = x_{D_i}(\text{mod } n).$$

Check whether $d_i = r_i$ is satisfied. If yes, then (r_i, s_i) on message m is a valid signature of U_i. Otherwise, the signature is invalid.

Step 3. On receiving the digital signature (r_i, s_i) of all members U_i, the secretary calculates and publicizes the group signature (r, s) on message m: The secretary first obtains the public $R_i = (x_{R_i}, y_{R_i})$ of all members and then calculates R.

$$R = \sum_{i=1}^{t} R_i = (x_R, y_R),$$
$$r = x_R(\text{mod } n),$$
$$s = \sum_{i=1}^{t} r_is_i(\text{mod } n).$$

2.3 Threshold Digital Signature Verification

Any recipient of (r, s) can verify the authenticity of the group signature on message m.

Step 1. The receiver first calculates the following equation:

$$S = \sum_{i=1}^{t} r_is_i(\text{mod } n),$$

to determine whether $S = s$ is satisfied. If yes, Step 2 is performed. Otherwise, the signature is invalid.

Step 2. The following equation is calculated next:

$$\begin{aligned} Q &= (x_Q, y_Q) = sG + h(m)N, \\ q &= x_Q(\text{mod } n). \end{aligned}$$

Determine whether $q = r$ is satisfied. If yes, (r, s) is an authentic group signature for message m, the signature is invalid.

3 Comment on the Yu-Chen Threshold Group Signature Scheme

3.1 Ambiguities

In the review of the Yu-Chen threshold signature scheme, we have corrected some typos in their paper. However, there are many ambiguities:

1) The order N of the elliptic curve cryptosystem is the same as the group public key N.
2) The order n of the base point G is the same as the threshold (t, n).
3) Secret key of individual group member U_i should be sent to the member U_i in a secure channel.
4) The elliptic scalar multiplication $h(m)N_i[\prod_{j=1,j\neq i}^{t} \frac{-x_j}{x_i-x_j}]$ should be denoted as $h(m)[\prod_{j=1,j\neq i}^{t} \frac{-x_j}{x_i-x_j}]N_i$.
5) When the receiver verifies the signature (r, s), he is required to compute:

$$S = \sum_{i=1}^{t} r_i s_i (\text{mod } n).$$

However, he does not know $(r_1, s_1, \ldots, r_t, s_t)$. Hence the group signature must be $(r, s, r_1, s_1, \ldots, r_t, s_t)$. This modification would reduce the efficiency of the group signature scheme.

3.2 Forgery Attack Against the Group Signature Scheme

3.2.1 Forge Signature of Group Member U_i

Suppose that an adversary wants to forge an individual signature of a group member U_i for any message m.

Without the knowledge of the secret key $f(x_i)$, he does the following steps:

Step 1. Choose a random integer $k_i, 1 \leq k_i \leq n-1$ to calculate D_i and r_i as follows:

$$\begin{aligned} D_i &= (x_{D_i}, y_{D_i}) = k_i G + h(m)[\prod_{j=1,j\neq i}^{t} \frac{-x_j}{x_i - x_j}]N_i, \\ r_i &= x_{D_i}(\text{mod } n). \end{aligned}$$

Step 2. Compute $s_i = k_i/r_i(\text{mod } n)$. Then the adversary sends the forged individual digital signature (r_i, s_i) of the member U_i to the secretary.

Obviously, the secretary cannot find this forgery.

3.2.2 Forge Group Signature

Suppose that an adversary wants to forge a group signature for any message m. Without the knowledge of the secret key $f(0)$, he does the following steps:

Step 1. Choose a random integer $s, 1 \leq s \leq n-1$ to calculate Q and r as follows:

$$\begin{aligned} Q &= (x_Q, y_Q) = sG + h(m)N, \\ r &= x_Q(\text{mod } n). \end{aligned}$$

Step 2. Choose $(r_1, s_1, \ldots, r_t, s_t)$ such that

$$s = \sum_{i=1}^{t} r_i s_i (\text{mod } n).$$

Obviously, the any recipient cannot find this forgery.

This kind of simple forgery attacks comes from the flaw that $h(m)$ does not contain the partial signature R.

Therefore, $h(m)$ must be replaced by $h(m, R)$.

4 Further Modification

In this section, we first present a modification to improve the Yu-Chen threshold group signature scheme, and then discuss a general coalition attack against threshold signature schemes.

4.1 The Modified Group Signature Scheme

We would like to improve the Yu-Chen threshold signature scheme against the forgery attacks. The key generation is the same as that of the Yu-Chen threshold group signature scheme.

4.1.1 Threshold Signature Generation

Suppose that there is a group that needs to sign a message, the members $U_1, U_2, \ldots, U_t$ can represent the group by signing message m.

Step 1. Each member U_i uses their secret key $f(x_i)$ and a random integer $k_i, 1 \leq k_i \leq n-1$ to calculate their individual signature (R_i, s_i) for message m.

$$R_i = (x_{R_i}, y_{R_i}) = k_i G, \text{ publicize } R_i.$$

Step 2. After receiving R_j from all other member U_i, each member U_i calculates R, e, and s_i as follows:

$$\begin{aligned} R &= \sum_{i=1}^{t} R_i, \\ e &= h(m, R), \\ s_i &= k_i + f(x_i)e[\prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}](\text{mod } n). \end{aligned}$$

Member U_i sends (R_i, s_i) to the secretary.

Step 3. On receiving the digital signature (R_i, s_i) of all members U_i, the secretary employs the following equations to confirm the validity of the signature:

$$\begin{aligned} R &= \sum_{i=1}^{t} R_i, \\ e &= h(m, R), \quad \text{and checks} \\ R_i &= s_i G + e[\prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}]N_i. \end{aligned}$$

If yes, then (R_i, s_i) on message m is a valid signature of U_i. Otherwise, the signature is invalid.

Step 4. If all (R_i, s_i) are valid, the secretary calculates

$$s = \sum_{i=1}^{t} s_i (\text{mod } n).$$

The group signature for the message m is (e, s).

4.1.2 Threshold Group Signature Verification

Any recipient of threshold group signature (e, s) can verify the authenticity of the group signature on message m by checking the following equation:

$$e = h(m, sG + eN).$$

This verification equation is the elliptic curve version of the Schnorr signature scheme [9], which is secure so far.

Because

$$s_i = k_i + f(x_i)e[\prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}](\text{mod } n)$$

implies

$$R_i = s_i G + e[\prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}]N_i,$$

the secretary is always to accept all individual signature (Ri, si). Thus

$$\sum_{i=1}^{t} R_i = (\sum_{i=1}^{t} S_i)G + e\sum_{i=1}^{t}(\prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j})N_i.$$

Hence, $R = sG + eN$ implies $e = h(m, sG + eN)$.

Therefore, the group signature (e, s) is always to satisfy the verification equation.

4.2 A General Coalition Attack Against Threshold Signature Schemes

Though our modification can withstand the forgery attack suffered by the Yu-Chen threshold group signature scheme, there is a general coalition attack against threshold signature schemes. In the ordinary threshold signature scheme, the group's secret key is $f(0)$, and each member U_i has the secret share $f(x_i)$. If t or more malicious members pool their secret shares together, they can recover $f(0)$ by applying Lagrange interpolating polynomial. Then each one of them can alone compute valid signatures for new messages on behalf of the group afterwards without the cooperation of other signers and without being detected by verifiers. Obviously, this violates the group's signing policy. Otherwise, if such coalition is permissive, other signers would follow this kind of dishonesty. Thus, each user can also alone compute valid group signatures after one coalition. It's terrible for threshold signature schemes.

This coalition attack is inherent in many threshold signature schemes using threshold secret share scheme, as long as the secret key can be recovered from secret shares.

The other paper of mine [10] provided approach to withstand this kind of coalition attack. Though it is easy to transpose it into elliptic curves, resulting scheme is perhaps not applicable for smart cards since it requires some more communication and computation.

5 Conclusions

We have pointed out that there are many ambiguities in the Yu-Chen threshold signature scheme. The verifiers cannot verify the signature, while adversaries can not only easily forge the signatures of individual members, but also forge group signatures without the knowledge of secret keys. Though we can modify it to withstand the forgery attack, the modified scheme cannot withstand the coalition attack inherent in many threshold signature schemes.

Acknowledgements

This material is based upon work funded by Science and Technology Department of Zhejiang Province of China under Grant No.2007C31G2130023.

References

[1] W. J. Caelli, E. P. Dawson, and S. A. Rea, "Pki, elliptic curve cryptography, and digital signatures," *Computers & Security*, vol. 18, no. 1, pp. 47-66, 1999.

[2] Y. Desmedt, "Society and group oriented cryptography: A new concept, Advances in Cryptology," *Proceeding Of Crypto'87*, LNCS 293, pp. 120-127, Springer-Verlag, Berlin, 1987.

[3] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. IT-31, pp. 469-472, 1985.
[4] L. Harn, "Group-oriented (t, n) threshold digital signature scheme and digital multisignature," *IEE Proceedings-Computer Digital Techniques*, vol. 141, no. 5, pp. 307-313, 1994.
[5] K. Koblitz, "Elliptic curve cryptosystems," *Mathenmatics of Computation*, vol. 48, no. 177, pp. 203-209, 1987.
[6] A. Menezes and S. Vanstone, "Elliptic curve systems," *Proposed IEEE P1363 Standard*, pp. 1-42, 1995.
[7] V. S. Miller, "Use of elliptic curves in cryptography," Cryptology, *CRYPTO'85*, LNCS 218, pp. 417-426, Springer-Verlag, Berlin, 1985.
[8] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, vol. 21, pp. 120-126, 1978.
[9] C. P. Schnorr, "Efficient signature generation by smart cards," *Journal of Cryptology*, vol. 3, no. 3, pp. 161-174, 1991.
[10] Z. Shao, "Improvement of threshold proxy signature scheme," *Computer Standard and Interface*, vol. 27, no. 1, pp. 53-59, 2005.
[11] Y. L. Yu, and T. S. Chen, "An efficient threshold group signature scheme," *Applied Mathematics and Computation*, vol. 167, no. 1, pp. 362-371, 2005.
[12] K. Zhang, "Threshold proxy signature schemes," *Information Security Workshop*, pp. 191-197, Japan, Sep. 1997.

Zuhua Shao was born in Shanghai, People's Republic of China, on 30 April 1948. He received B.S. degree in mathematics and M.S. in algebra from the Northeastern Normal University, People's Republic of China in 1976 and 1981 respectively. Since 1990 he has taught computer science as an associated professor in the Hangzhou Institute of Financial Managers, The Industrial and Commerce Bank of China. Now he is a professor at the Zhejiang University of Science and Technology. His current research interests are cryptography and financial data security.