Further Study on Proxy Authorization and Its Scheme

Xuanwu Zhou, Yang Su, and Ping Wei

Key Lab of Network & Information Security of the APF,
Engineering College of the APF, 710086 Xi'an, P.R. China
{schwoodchow,wjxuanwu}@163.com

Abstract. Proxy authorization makes it possible to entrust the right of signing or making decisions to other parties. This paper analyzes the basic principles and security problems of proxy authorization schemes and presents three proxy authorization schemes based on elliptic curves cryptosystem. In the first multi-party proxy authorization scheme, a group of n members can cooperate to entrust their right, and the authorizing right can be supervised by secret sharing mechanism. In the second multicast proxy authorization scheme, the members can entrust their right in multicast mode. The multicasting design strategy prevents coalition attack, avoids the problem of generalized signature forgery. In the last conditionally anonymous scheme, the identity blinding algorithm enables the proxy signer to be anonymous and the anonymity can also be revoked if necessary. This design strategy avoids the misuse of proxy authorization and renders effective supervision on signature entrusting and proxy signing.

Keywords: Proxy authorization, multi-party authorization, multicast communication, forward security, BAN logic.

1 Introduction

Proxy authorization is essential in electronic commerce and other electronic transactions, it makes it possible for one to entrust his right of signing or making decisions to other parties. In electronic commerce and electronic government, there are many cases that proxy authorization is needed. For example, a member of the board of a company will have to sign a file in the name of the company board, or a secretary has to ratify an application for the manager. In electronic transactions, proxy authorization signature can well satisfy these application requirements. With the development electronic transactions, proxy authorization schemes with additional properties have been a new trend in the study of proxy authorization designed for specific requirements.

The prototype of proxy authorization signature is proxy signature, which is first put forward by Mambo, Usuda and Okamoto in 1996. Proxy signature can be defined as the following:

A and B are the two users of a signature system(M , S , K , $SIG(\cdot)$, $VER(\cdot)$), and (k_A , K_A), (k_B , K_B) are their secret and public key pairs, if the following conditions are satisfied:

J. Zhou (Ed.): Complex 2009, Part II, LNICST 5, pp. 1701 – 1718, 2009.

(1) A computes a number f with his secret key k_A, and sends it to B secretly;

(2) Any one (including B) will have no advantage trying to get k_A if he gets f ;

(3) B generates a new secret key for signature with k_B and f, and there exists a public verifying algorithm $VER_{AB}(\cdot)$ which satisfies $VER_{AB}(K_A, s, m) =$ True is equivalent to $s = SIG(f_{AB}, m)$;

(4) If anyone intends to get k_A, k_B, f or f_{AB} ,any signature $s = SIG(f_{AB}, m)$ will not be of any help.

Then $(M, S, K, SIG(\cdot), VER(\cdot))$ is a proxy signature system and A authorizes B with his right for generating signature , B gets the proxy authorization. A is original signer; B is proxy authorization signer, f is the entrusting secret key and f_{AB} is proxy signature key.

Presently, the security threats and weaknesses of different proxy authorization schemes can be summarized as the following:

(1) Lack of secrecy protection. Most proxy signature schemes lay emphasis on the protection of authorization rights of original signer, so the secret parameter f and private key k_A of original signer is well protected in the protocol. But the secrecy and security of proxy signer fall into neglect for the consideration of efficiency. Although the leakage of proxy signature key poses no threat to the original private key, yet it breaks the rules of fair transaction in e-commerce or e-government.

(2) Instable signing group. In present schemes, system parameters are related with the identity information of proxy signers. When accepting or deleting proxy signer , the system parameters essential for signature, authentication and revealing signer's anonymity will also change with the identity information. The original system parameters will naturally be invalid, so the group public key and private keys of other signers will also have to be changed. In addition, the resetting of the whole signature system will also be needed to reinforce the security of the renewed scheme. In such circumstances as with frequently changed proxy signers, present proxy authorization signature schemes prove to be impracticable too [1-8].

(3) Security threats from coalition attack and generalized forgery. A viable proxy authorization signature scheme is resistant to forgery attack both from outer adversaries and inner adversaries. Nevertheless, in present signature schemes, attackers from the proxy group have apparent advantages over outer adversaries for the original signer and proxy signer must interact to generate system parameters in the system initialization protocol. Therefore, present proxy authorization signature schemes are vulnerable to coalition attack and generalized signature forgery.

(4) Low efficiency. The security of present proxy authorization signature schemes mostly depends on the difficulty of discrete logarithm problem on finite field $GF(p)$ or large prime factor decomposing problem. What's more, proxy authorization signature is a complex authentication protocol on multi-levels, and many complex computations in the existing schemes (such as scalar multiplication, inverse transformation) have proved unnecessary and redundant. Therefore, present

signature schemes have such weaknesses as large secret key size, complex computation, and low efficiency for hardware and software application [9-14].

To overcome the security threats and weaknesses in existing schemes, we present three proxy authorization schemes based on ECC (Elliptic Curve Cryptosystem) and provide analyses on security, feasibility and efficiency. As to the security proof of the schemes, we introduce BAN logic and Kailar logic into the analyzing of schemes. The one-way trapdoor function is based on ECDLP (Elliptic Curves Discrete Logarithm Problem), and the algorithms of the schemes take great advantage of the superiority of ECC, such as high efficiency, short key length and etc.

2 Multi-party Proxy Authorization Scheme

First, we present a multi-party proxy authorization scheme based on ECC, the scheme operates in a sequential mode.

2.1 System Parameter

Considering system security and implementing efficiency, the parameters of the scheme are as follows:

F_q denotes a finite field (q is a large prime number), an elliptic curve on this finite field is defined as $E: y^2 = x^3 + ax + b$ ($a,b \in F_q$, $4a^3 + 27b^2 \pmod{q} \neq 0$). $P \in E(F_q)$ is a base point whose order is large prime number of considerable scale. The order satisfies $ord(P) = l \geq 160$.

Then, as to any $k \in Z_l^*$, the computation of $K = kP$ via k and P is computationally feasible; but the computation of k via K and P is the ECDLP (Elliptic Curves Discrete Logarithm Problem), which is computationally infeasible [15-19].

ψ denotes a function which makes the conversion from a point $P = (x, y)$ on elliptic curve to x, and it's marked $(P)_x$. A is the message sender, C is the signature verifier, $B_1, B_2, \ldots B_n$ is a group of message signers. $k_i \in (1, 2 \ldots n-1)$ is the private key of B_i, and $K_i = k_i p$ is the corresponding public keys. $H(\cdot)$ is secure one-way hash function.

2.2 Signature Generating

A sends message m to the first proxy signer B_1, and the signature for this message is temporarily set as $s = 0$. After receiving signature message (m,(s_{i-1}, R_{i-1})), every signer B_i (i≥2) testifies the signature and executes the following protocols.

Step 1: B_i randomly selects $u_i \in Z_l^*$ and computes

$$m' = H(m), \tag{1}$$

$$R_i = u_i p \neq 0, \tag{2}$$

$$s_i = s_{i-1} + k_i (R_i)_x - m' u_i (\mathrm{mod}\, l). \tag{3}$$

Step 2: then he sends (m , (s_i , R_i)) to the next signer B_{i+1}, and sends R_i to other signers after B_i and also the signature verifier C .

2.3 Signature Verifying

Every proxy signer B_i (i≥2) should verify signature of B_1 , B_2 ,… B_{i-1} , and the signature verifier C should testify all the proxy signers. When $2 \le i \le n+1$, B_i testifies:

$$\sum_{j=1}^{i-1} (R_j)_x K_j \,?= m' \sum_{j=1}^{i-1} R_j + s_{i-1} p . \tag{4}$$

If the formula is correct, B_i accepts the proxy signature of $B_1, B_2, \ldots B_{i-1}$ as valid ones, if the proxy signatures precedent prove invalid, B_i will deny to generate a new signature.

As to the signature verifier C , he testifies the following:

$$\sum_{j=1}^{n} (R_j)_x K_j \,?= m' \sum_{j=1}^{n} R_j + s_n p . \tag{5}$$

If the formula is correct, C accepts the proxy signature of $B_1, B_2, \ldots B_n$ as valid ones, if the formula proves incorrect , C will terminate the protocol.

2.4 Analysis of Multi-party Scheme

According to formula (3) $s_i = s_{i-1} + k_i (R_i)_x - m' u_i (\mathrm{mod}\, l)$

$$\Rightarrow s_{i-1} = s_{i-2} + k_{i-1} (R_{i-1})_x - m' u_{i-1} (\mathrm{mod}\, l)$$

$$= \sum_{j=1}^{i-1} (k_j (R_j)_x - m' u_j) (\mathrm{mod}\, l)$$

According to formula (5) $\sum_{j=1}^{n}(R_j)_x K_j \,?= m'\sum_{j=1}^{n} R_j + s_n p$

$$\Rightarrow m'\sum_{j=1}^{i-1} R_j + s_{i-1} p = m'\sum_{j=1}^{i-1} u_j p + \sum_{j=1}^{i-1}(k_j (R_j)_x - m' u_j)(\mathrm{mod}\, l)p$$

$$= \sum_{j=1}^{i-1}(m' u_j + k_j (R_j)_x - m' u_j)(\mathrm{mod}\, l)p$$

$$= \sum_{j=1}^{i-1} k_j (R_j)_x (\mathrm{mod}\, l)p = \sum_{j=1}^{i-1}(R_j)_x K_j$$

Thus the multi-party proxy authorization scheme proves correct.

3 Proxy Authorization for Multicast Communication

In this part, we present a proxy authorization scheme for multicast communication environment. The scheme operate in a multicast mode, the scheme is composed of n members that intend to entrust their right and a signature collector.

3.1 System Parameter

F_q denotes a finite field (q is a large prime number), an elliptic curve on this finite field is defined as $E: y^2 = x^3 + ax + b$ ($a,b \in F_q$, $4a^3 + 27b^2 (\mathrm{mod} q) \neq 0$). $P \in E(F_q)$ is a base point whose order is large prime number of considerable scale. The order satisfies $ord(P) = l \geq 160$.

ψ denotes a function which makes the conversion from a point $P = (x, y)$ on elliptic curve to x , and it's marked $(P)_x$. A is the message sender, C is the signature verifier, $B_1, B_2, \ldots B_n$ is a group of message signers. $k_i \in (1, 2 \ldots n-1)$ is the private key of B_i , and $K_i = k_i p$ is the corresponding public keys. $H(\cdot)$ is secure one-way hash function. B_c is the message collector in the multicast proxy authorization scheme.

3.2 Signature Generating

A sends message m to every signer B_i (i =1,2,…, n) and the signature collector B_c , the signature for the time being is defined as s =0. After getting the message, every signer B_i and the signature collector B_c will execute the following:

Step 1: B_i randomly selects $u_i \in Z_l^*$ and computes

$$R_i = u_i p \neq 0 . \tag{6}$$

Then he sends R_i to the signature collector.

Step 2: After getting all the signature piece R_i (i =1,2,…, n), B_c computes

$$R = \sum_{i=1}^{n} (R_i)_x R_i . \tag{7}$$

Then B_c sends R to every signer B_i (i =1,2,…, n)

Step 3: As to message m, every signer B_i computes

$$m' = H(m), \tag{8}$$

$$s_i = (m' + (R)_x)k_i - (R_i)_x u_i (\mathrm{mod}\, l) . \tag{9}$$

s_i is the signature piece of message m generated by B_i ,then B_i sends(m , s_i) to the signature collector B_c.

Step 4: After getting all the signature pieces (m , s_i) (i =1,2,…, n), B_c computes

$$s = \sum_{j=1}^{n} s_j (\mathrm{mod}\, l) . \tag{10}$$

Then (m , s, R) is the final signature for message m, and it will be sent to signature verifier C.

3.3 Signature Verification

After receiving signature message (m , s, R), signature verifier C will compute the following formula to testify the signature.

$$m' = H(m), \tag{11}$$

$$\sum_{i=1}^{n} ((R)_x + m')K_i \,?= R + sp . \tag{12}$$

If the formula is correct, then (m , s, R) is a valid proxy signature for message m generated by B_i (i =1, 2,…, n), or else it will prove invalid.

3.4 Multi-cast Scheme Analysis

According to formula $s_i = (m' + (R)_x)k_i - (R_i)_x u_i (\mathrm{mod}\, l)$

$$\Rightarrow s = \sum_{i=1}^{n} s_i (\mathrm{mod}\, l)$$

$$= \sum_{i=1}^{n} [(m' + (R)_x)k_i - (R_i)_x u_i] (\mathrm{mod}\, l)$$

According to formula $R_i = u_i p \neq 0$

$$\Rightarrow R + sp = \sum_{i=1}^{n} (R_i)_x R_i + \sum_{i=1}^{n} [(m' + (R)_x)K_i - (R_i)_x R_i]$$

$$= \sum_{i=1}^{n} [(R_i)_x R_i + (m' + (R)_x)K_i - (R_i)_x R_i]$$

$$= \sum_{i=1}^{n} (m' + (R)_x)K_i = \sum_{i=1}^{n} ((R)_x + m')K_i$$

Thus the proxy authorization scheme for multicast communication proves correct.

4 Anonymous Proxy Authorization Signature Based on ECC

In this part, we will present a proxy authorization scheme with conditional anonymity, the conditional anonymity can protect the privacy of the proxy signer and renders effective supervision on the proxy entrusting and proxy signing.

4.1 System Parameters

The parameters of the scheme are as follows: F_q denotes a finite field (q is a prime number of n bits, $n \geq 190$), an elliptic curve on this finite field is defined as follows: $E: y^2 = x^3 + ax + b$ ($a, b \in F_q$, $4a^3 + 27b^2 (\mathrm{mod} q) \neq 0$). $P \in E(F_q)$ is a base point whose order is a large prime number l (which satisfies $l \geq 160$ bits). $\# E(F_q)$ denotes the order of the elliptic curve which has a factor of large prime number larger than 160 bits. $k_A, k_B \in Z_l^*$ are the private key of original signer A and a proxy signer B ($K_A = k_A P$, $K_B = k_B P$ is the public key), $ID_B \in E(F_q)$ is the identity information of B, $w \in Z_l^*$ f is the entrusting information, $h(\cdot)$ is a secure one-way hash function.

4.2 Proxy Entrusting

Step 1: A randomly selects $u \in Z_l^*$, and computes

$$ID'_B = h(u \| (ID_B)_x)P. \tag{13}$$

Step 2: A sends ID'_B to proxy signer B secretly, ID_C is the blinded identity of B. Later, B generates and issues signatures with the blinded identity ID'_B.

Step 3: Similarly, A performs interactive protocols with other group members and then keeps record of three-element triple (ID_B , ID'_B , u) of each member so that he can trace the group member when necessary.

Step 4: With randomly selects $u \in Z_l^*$ A computes

$$U = uP \neq 0, \tag{14}$$

$$s_A = k_a h(w \| (U)_x) + u(\mathrm{mod} l). \tag{15}$$

Step 5: A sends (U , s_A , w) to B as the entrusting private key .

Step 6: B computes the following formula to testify the validity of s_A

$$s_A P ? = h(w \| (U)_x)K_a + U. \tag{16}$$

Step 7: If testifying formula (4) is incorrect, the protocol will be aborted, otherwise, B generates a pair of new keys (k_{bc} , $K_{bc} = k_{bc}P$) with the help of A, and then declares his new public key K_{bc} with identity ID'_B

Step 8: B computes

$$k_c = (s_A + k_{bc})(\mathrm{mod} l), \tag{17}$$

$$I = ID_C + ID_B + k_{bc}K_a. \tag{18}$$

Then k_c is the private key for proxy authorization signature, and $K_c = k_c P$ is the corresponding public key for verifying proxy signature.

4.3 Proxy Authorization Signature Generating

Step 1: As to message m, B randomly selects $v \in Z_l^*$ and computes

$$V = vP \neq 0, \tag{19}$$

$$s = k_c^{-1}(v + m(V)_x)(\mathrm{mod} l). \tag{20}$$

Step 2: Then B sends (m , s , V , w , K_{bc}) to verifier as the proxy authorization signature for message m.

4.4 Signature Verification

Step 1: The verifier first testifies

$$h(w\|(U)_x)K_a + U + K_{bc} = K_c, \tag{21}$$

$$sK_c \,?= V + m(V)_x P. \tag{22}$$

Step 2: If the formula is correct, he affirms that the signature (m, s, V, w, K_{bc}) is signed by a valid proxy signer, or else the signature will not be accepted.

4.5 Anonymous Identity Tracing

When disputes or any doubt arise, the anonymity of a proxy signer has to be revoked for notarization.

Step 1: The signature verifier first testifies proxy signature (m, s, V, w, K_{bc}) with the formula in verification protocol.

Step 2: Then A computes

$$E = I - k_a K_{bc}, \tag{23}$$

$$F = I - ID_C - k_a K_{bc}. \tag{24}$$

Step 3: A searches for the three-element triple (ID_B, ID'_B, u) in which $ID_B + ID_C = E$、$ID_B = F$ from all the triples that he has kept. Then ID_B in the triple is just the real identity information of the specific group member.

4.6 Analysis of the Anonymous Scheme

As to formula $h(w\|(U)_x)K_a + U + K_{bc} = K_c$ in signature verifying,

$$h(w\|(U)_x)K_a + U + K_{bc} = h(w\|(U)_x)k_a P + uP + k_{bc}P$$

$$= (h(w\|(U)_x)k_a + u + k_{bc})P$$

According to formula $s_A = k_a h(w\|(U)_x) + u (\mathrm{mod} l)$

$$\Rightarrow (h(w\|(U)_x)k_a + u + k_{bc}) = (s_A + k_{bc})(\mathrm{mod} l)$$

$$\Rightarrow (h(w\|(U)_x)k_a + u + k_{bc})P = (s_A + k_{bc})P$$

According to formula $k_c = (s_A + k_{bc})(\mathrm{mod} l)$

$$\Rightarrow (s_A + k_{bc})P = k_c P = K_c$$

As to formula $V + m(V)_x P = vP + m(V)_x P = (v + m(V)_x)P$

According to $s = k_c^{-1}(v + m(V)_x)$

$$\Rightarrow sk_c = (v + m(V)_x)$$

$$\Rightarrow (v + m(V)_x)P = sk_c P = sK_c$$

Thus the signature protocol of the anonymous proxy authorization is correct.

As to identity tracing formula, according to formula $E = I - k_a K_{bc}$ and formula

$$I = ID_C + ID_B + k_{bc} K_a$$

$$\Rightarrow E = ID_C + ID_B + k_{bc} K_a - k_a K_{bc} = ID_C + ID_B$$

According to formula $F = I - ID_C - k_a K_{bc}$ and formula $I = ID_C + ID_B + k_{bc} K_a$

$$\Rightarrow F = ID_C + ID_B + k_{bc} K_a - ID_C - k_a K_{bc} = ID_B$$

Thus the notarization protocol of the anonymous scheme also proves correct.

5 Analyses of the Scheme with BAN Logic and Kailar Logic

As to the above schemes, we present formalizing analyses of their privacy, integrity and other properties with BAN logic and Kailar logic.

5.1 Formalizing Analyses of Protocol and BAN Logic

Methods for cryptographic protocol analyzing can be categorized as natural language, symbolization method and formalization method. The former two methods have long been put into practice; in 1978 Needham and Schroeder put forward the method of formalization analyzing to find the flaws in protocols. Formalization method is mathematical method to describe the system properties, and it aims to find the inconsistence and incompleteness in a system. Formalization methods focus on protocol properties and are independent of specific cryptographic algorithms. Formalization methods have two forms: methods based on algebraic analyses and methods based on logic inferring. The former one includes Dolve-Yao model which was put forward by Dolve and Yao in 1983, Dolve-Yao model is a algebraic model for cryptographic protocol modeling. The other is based on logic methods, among which BAN logic is the most important one. BAN logic is cryptographic protocol analyzing method based on logic inferring put forward by Burrows, Abadi and Needham in 1989.It focuses on entity trust and trust infering, and makes great improvement on SPA(Security Protocol Analysis)problem.The following is the basic sybolms in BAN logic.

S $|\equiv$ M: denotes S believes M;

S $\triangleleft$ M: denotes S sees M;

S $|\approx$ M: denotes S said M, that is S sent message M;

$S \mid\Rightarrow M$: denotes S controls M, that is S has the judging right to message M;

$\#(M)$: denotes fresh(M), that is message M is new;

$\mid\xrightarrow{K} S$: denotes S owns public key K, and the relevant private key is secure;

$\{M\}_{S_S}$: denotes generating a signature for message M with the private key (for signing) of S;

$\{M\}_{S_S K_G}$: denotes the joint signature with private key of one party and public key of the other.

The following is the rules of BAN logic that we refer to in our analyzing.

R1 Message Meaning Rule

$$\frac{A \mid\equiv A \xleftrightarrow{K} B, A \lhd \{M\}_K}{A \mid\equiv (B \mid\approx M)}, \frac{A \mid\equiv \mid\xrightarrow{K} B, A \lhd \{M\}_{K^{-1}}}{A \mid\equiv (B \mid\approx M)}. \tag{25}$$

R2 One-time Random Number Testing Rule

$$\frac{A \mid\equiv fresh(M), A \mid\equiv B \mid\approx M}{A \mid\equiv (B \mid\equiv M)}. \tag{26}$$

R3 Adjudication Rule

$$\frac{A \mid\equiv B \Rightarrow M, A \mid\equiv B \mid\equiv M}{A \mid\equiv M}. \tag{27}$$

R5 Trust Projecting Rule

$$\frac{A \mid\equiv (X, Y)}{A \mid\equiv X}. \tag{28}$$

R7 Once Said Projecting Rule

$$\frac{A \mid\equiv B \mid\approx (X, Y)}{A \mid\equiv B \mid\approx X, A \mid\equiv B \mid\approx Y}. \tag{29}$$

R10 Seeing Rules

$$\frac{A \mid\equiv \mid\xrightarrow{K} B, A \lhd \{M\}_{K^{-1}}}{A \lhd M}, \frac{A \mid\equiv \mid\xrightarrow{K} A, A \lhd \{M\}_K}{A \lhd M}. \tag{30}$$

5.2 Scheme Analyzing with BAN Logic

S (Signature Generator), G and C (Signature Verifier) are the protocol parties in the scheme, $K_c = k_c P$, $K_g = k_g P$, $K_s = k_s P$ are the relevant public keys, T is time sign. In our scheme $\{ T_C, M \}_{S_C K_G}$ is equivalent to $\{ T_C, M \}_{K_{cg}}$.

The scheme can be described in Needham symbol as the following.

1) S→C: S, G 2) C→S: $\{ T_C, R,M,G, \{ T_C ,R,M,S \}_{S_C K_G} \}_{S_C K_S}$

3) S→G: $\{ T_C ,R,M,S \}_{S_C K_G}$, $\{ T_S ,S \}_{S_S K_G}$ 4) G→S: $\{ T_S+1 \}_{S_G K_S}$

And can be further processed as the following.

C→S: $\{ T_C, M, \{ T_C, M \}_{S_C K_G} \}_{S_C K_S}$

S→G: $\{ T_C, M \}_{S_C K_G}$, $\{ T_S \}_{S_S K_G}$ from S

G→S: $\{ T_S+1 \}_{S_G K_S}$ from G

And then we can get the basic hypothesis for the protocols.

(1) $S \mid\equiv \mid \xrightarrow{K_C} C$ (2) $S \mid\equiv \mid \xrightarrow{K_G} G$ (3) $G \mid\equiv \mid \xrightarrow{K_S} S$

(4) $C \mid\equiv \mid \xrightarrow{K_S} S$ (5) $C \mid\equiv G \mid\equiv \mid \xrightarrow{K_S} S$ (6) $G \mid\equiv C \mid\Rightarrow M$

(7) $S \mid\equiv (C \mid\Rightarrow M)$ (8) $G \mid\equiv \#(T_C)$ (9) $S \mid\equiv \#(T_C)$ (10) $G \mid\equiv \#(T_S)$

As to step 2 in the scheme, according to explanatory rule of BAN logic,

$$\Rightarrow S \lhd \{ T_C, M, \{ T_C, M \}_{K_{cg}} \}_{K_{sc}}$$

According to hypothesis (1) and R1, R10

$$\Rightarrow S \mid\equiv C \mid\approx \{ T_C ,M, \{ T_C ,M \}_{K_{cg}} \}$$

With rule R7 $\Rightarrow S \mid\equiv C \mid\approx \{ T_C, M \}$

According to hypothesis (9), R2 and the conclusion above

$$\Rightarrow S \mid\equiv C \mid\equiv \{ T_C, M \}$$

Then with rule R5, we can infer

$$\Rightarrow S \mid\equiv C \mid\equiv M$$

According to hypothesis (7), R3 and the conclusion above

$$\Rightarrow S \mid\equiv M$$

As to step 3 in the scheme, according to explanatory rule of BAN logic,

$$\Rightarrow G \lhd \{ T_C, M \}_{S_C K_G}, \{ T_S \}_{S_S K_G}$$

According to hypothesis (1), R1, R10 and the conclusion above

$$\Rightarrow G \mid\equiv C \mid\sim \{ T_C, M, \{ T_C, M \}_{K_{cg}} \}$$

With rule R7 $\Rightarrow G \mid\equiv C \mid\sim \{ T_C, M \}$

According to hypothesis (9), R2 and the conclusion above

$$\Rightarrow G \mid\equiv C \mid\equiv \{ T_C, M \}$$

And with rule R5

$$\Rightarrow G \mid\equiv C \mid\equiv M$$

According to hypothesis (7), R3 and the conclusion above

$$\Rightarrow G \mid\equiv M$$

With the latter message in step 3

$$\Rightarrow G \mid\equiv S \mid\sim \{T_S, S, \{ T_C, R, M, S \}_{S_C K_G} \}$$

And with rule R7

$$\Rightarrow G \mid\equiv S \mid\sim \{ \{ T_C, R, M, S \}_{S_C K_G} \}$$

According to hypothesis (9), (5), rule R2 and the conclusion above

$$\Rightarrow G \mid\equiv S \mid\equiv M$$

And with step 4 $\Rightarrow S \mid\equiv G \mid\sim \{T_S + 1\}$

According to hypothesis (9), (10), rule R2 and the conclusion above

$$\Rightarrow S \mid\equiv G \mid\equiv M$$

Thus, we prove that the schemes satisfy the objectives of BAN logic:

1. $G \mid\equiv M$ 2. $S \mid\equiv M$ 3. $G \mid\equiv S \mid\equiv M$ 4. $S \mid\equiv G \mid\equiv M$

And they also satisfy the objectives of VO logic (Van Oorschot):

1. $G \mid\equiv S \mid\sim M$ 2. $G \mid\equiv C \mid\sim M$ 3. $G \mid\equiv \#(M)$

5.3 Scheme Analyzing with Kailar Logic

First, we introduce the basic rules and symbols in Kailar logic that will be referred to in our analyzing.

$K_a \mid\equiv$ S: denotes K_a Authenticates S, that is K_a can testify the validity of signature from S

$S \lhd \{M\}_{S_S}$: denotes someone has sent message M to S with signature generated with S_s

$S \mid\approx M$: denotes S says M, that is S declares M and is responsible for M

$S \xrightarrow{m}$: denotes S has sent message m , that he sends message m to others

$\xrightarrow{m} S$: denotes S has received message m , that is S receives message m

$S \mid\Rightarrow M$: denotes S can prove M, that is S can prove M to a third party without disclosing any secret

$X \in M$: denotes X is an explanation of M or it is combination of several domains

$\mid\xrightarrow{m} S$: denotes S is trusted for the correctness of message m that he has declared.

The following are the relevant rules in Kailar logic.

R1 Comprising Rule

$$\frac{S \mid\Rightarrow x; x \Rightarrow y}{S \mid\Rightarrow (x, y)}. \tag{31}$$

R2 Signature Rule

$$\frac{S \lhd \{m\}_{k^{-1}}; x \in m; S \mid\Rightarrow (k \mid\equiv B)}{S \mid\Rightarrow (B \mid\approx x)}. \tag{32}$$

R3 Trust Rule

$$\frac{S \mid\Rightarrow (B \mid\approx x); S \mid\Rightarrow (\mid\xrightarrow{x} B)}{S \mid\Rightarrow x}. \tag{33}$$

R4 Connection Rule

$$\frac{S \mid\Rightarrow x; S \mid\Rightarrow y}{S \mid\Rightarrow (y \mid x)}. \tag{34}$$

In our scheme, $K_c = k_c P$, $K_a = k_a P$, $K_b = k_b P$ are the public keys.

The communication protocol in the scheme is equivalent to a protocol (described in Needham symbol).

1) A→C: $\{ m,A,B \}_{K_{AC}}$ 2) C→A: $\{ k_a , K_b \}_{K_{AC}}$ 3)A→B:h(m), $\{ \{ m \}_{k_a} \}_{K_b}$

4) B→C: $\{ h(m),A,B, \{ \{ m \}_{k_a} \}_{K_b} \}_{K_{BC}}$ 5) C→B: $\{ k_b , K_a \}_{K_{BC}}$

And then we can get the basic hypothesis for the protocols.

(1) $A,B \mid\Rightarrow (K_{AC} \mid\equiv C , K_{BC} \mid\equiv C)$ (2) $A,C \mid\Rightarrow (k_b \mid\equiv B)$

(3) $B,C \mid\Rightarrow (k_a \mid\equiv A)$ (4) $A,B \mid\Rightarrow \mid\xrightarrow{m} C$

(5) $A \mid\approx m \Rightarrow A \xrightarrow{m}$ (6) $B \mid\approx h(m) \Rightarrow \xrightarrow{m} B$

As to step 4 in the scheme, according to Kailar logic,

$$\Rightarrow C \mid\Rightarrow B \mid\approx h(m), C \mid\Rightarrow B \mid\approx \{ \{ m \}_{k_a} \}_{K_b}$$

And with connection rule R4

$$\Rightarrow C \mid\Rightarrow (B \mid\approx h(m), B \mid\approx \{ \{ m \}_{k_a} \}_{K_b})$$

According to hypothesis (2), rule R2 and the conclusion above

$$\Rightarrow C \mid\Rightarrow (\xrightarrow{h(m)} B , \xrightarrow{\{\{m\}_{k_a}\}_{k_b}} B)$$

With hypothesis (4), rule R3

$$\Rightarrow A \mid\Rightarrow (\xrightarrow{h(m)} B , \xrightarrow{\{\{m\}_{k_a}\}_{k_b}} B)$$

According to hypothesis (2), signature rule R2

$$\Rightarrow A \mid\Rightarrow (B \mid\approx h(m))$$

According to hypothesis (6), comprising rule R1

$$\Rightarrow A \mid\Rightarrow (\xrightarrow{m} B)$$

As to step 1,2 and4, According to hypothesis (1), trust rule R3

$$\Rightarrow C \mid\Rightarrow (A \xrightarrow{h(m)}) C \mid\Rightarrow (A \xrightarrow{\{\{m\}_{k_a}\}_{k_b}})$$

Then with connection rule R4

$$\Rightarrow C \mid\Rightarrow (A \xrightarrow{h(m)} , A \xrightarrow{\{\{m\}_{k_a}\}_{k_b}})$$

According to hypothesis (3), rule R2, R1and the conclusion above

$$\Rightarrow C \mid\Rightarrow (A \mid\approx m)$$

According to hypothesis (1), trust rule R3

$$\Rightarrow \mathrm{B}\left|\Rightarrow(\mathrm{A}\right|\approx \mathrm{m})$$

According to hypothesis (5), R1and the conclusion above

$$\Rightarrow \mathrm{B}\left|\Rightarrow(\mathrm{A}\xrightarrow{m}\right)$$

Thus, we prove that the schemes satisfy the objectives of Kailar logic:

$$1.\ \mathrm{B}\left|\Rightarrow(\mathrm{A}\xrightarrow{m}\right) \quad 2.\ \mathrm{A}\left|\Rightarrow(\xrightarrow{m}\mathrm{B}\right)$$

6 Additional Properties of the Scheme

Additional properties make proxy authorization more applicable for real-life application; furthermore, many additional properties come from the special requirements of different circumstances. In addition to the advantages of present proxy authorization signature schemes, the proposed schemes have many additional properties, such as conditional anonymity, forward security and multi-party supervision. These properties make our scheme more applicable to such circumstances as with restricted computation ability and integrated space, circumstances with limited bandwidth yet requiring for high-speed operation.

(1) Conditional anonymity. In the scheme, the attack of ID_B via ID'_B is equivalent to ECDLP and attacking the one-way property of hash function $h(\cdot)$. In the verification protocol, the verifier first testifies whether identity certificate is from a valid proxy signer by formula $h(w\|(U)_x)K_a + U + K_{bc} = K_c$,then checks the validity of signature with the formula $sK_c \,?= V + h(m)(V)_x P$, the proxy signer who signs can not deny that he has generate a proxy signature of message m with private key k_c and random parameter v. If disputes or any question arise, the identity information ID_B of the group member can be recovered with $E = I - k_a K_{bc}$ and formula $F = I - ID_C - k_a K_{bc}$ in anonymity tracing protocol, thus the anonymity is conditionally traceable.

Furthermore, the anonymity entrusting and revocation protocol can be executed on secret sharing mode, that is, the secret can be divided into n pieces for n members, only a group of no less than t members can cooperate to recover the secret. With secret sharing, the scheme provides effective supervision on authorization and signature generating.

(2) Forward security. The key pair (K_c , k_c) is irrelevant with the real identity ID_B. No one but the proxy signer who generates and issues the signature can figure out whether two different proxy signatures are generated by the same signer. If a signature is successfully attacked or the anonymity tracing protocol is executed,

other proxy signature key pairs and identity are still secure and anonymous, so the group signature scheme proves to be forward secured.

(3) Resistance to coalition attack. If proxy signers or the original singer collude to forge proxy authorization signature of B, they will also have to attack k_c and the random number v according to the analyses outlined above. Since these secret parameters are generated randomly and independently for a specific signature process, other proxy signers and the original singer know nothing but their own secret parameters, so the coalition attack of original signer and valid proxy signers possesses no superiorities over that of outer adversaries and single attacker.

7 Conclusion and Future Work

In the paper, we analyze the basic principles and potential application of proxy authorization. Considering the security problems and system flaws in other schemes, we present three improved proxy authorization schemes. The first two schemes can achieve multiple authorizations for multicast communication environment, and the schemes can be reinforced by secret sharing mechanism. The third scheme can provide conditional anonymity for proxy signers, if a signature is successfully attacked or the identity tracing protocol is executed, other signatures generated by the same proxy signer and signatures generated by other signers are still secure and anonymous, so the scheme proves to be forward secured.

The proposed schemes can achieve the same security with less storing space, smaller communication band-width and less overheads of the system and thus well embody the design principles of low communication costs and system overheads.

Acknowledgments. This work was supported in part by the Nature Science Foundation of China under Grant No.60842006 and Grant No. 60573032.

And the authors should also thank the anonymous reviewers for their constructive advice and comments to the paper, with which we can be able to improve our work clerically and academically.

References

1. Nakanishi, T., Tao, M.: A Group Signature Scheme Committing the Group. In: Deng, R.H., Qing, S., Bao, F., Zhou, J. (eds.) ICICS 2002. LNCS, vol. 2513, pp. 73–84. Springer, Heidelberg (2002)
2. Huang, Z.: Research on Digital Signature with Additional Properties. Xidian University, Xi'an (2005)
3. Zhou, X.: Dynamic Group Signature with Forward Security and Its Application. In: Proceeding of the Sixth International Conference on Grid and Cooperative Computing GCC 2007, pp. 473–480. IEEE Press, Piscataway (2007)
4. Avanzi, R.M.: Aspects of Hyper-elliptic Curves over Large Prime Fields in Software Implementations. In: International Association for Cryptology Research 2004, pp. 148–162. Springer, Heidelberg (2004)

5. Hui-Xian, L., Chun-tian, C.: A New (t, n)-threshold Multi-secret Sharing Scheme. In: Hao, Y., Liu, J., Wang, Y.-P., Cheung, Y.-m., Yin, H., Jiao, L., Ma, J., Jiao, Y.-C. (eds.) CIS 2005. LNCS, vol. 3802, pp. 421–426. Springer, Heidelberg (2005)
6. Park, H.-U., Lee, I.-Y.: A digital nominative proxy signature scheme for mobile communication. In: Qing, S., Okamoto, T., Zhou, J. (eds.) ICICS 2001. LNCS, vol. 2229, pp. 451–455. Springer, Heidelberg (2001)
7. Zdzislaws, H., Knap, M.M.: Research on Pre-processing and Post-processing of Data in the Process of Creation Quasi-optimal Decision Trees. Intelligence Methods (11), 13–15 (2002)
8. Ting-Yi, C., Chou-Chen, Y., Min-Shiang, H.: A threshold signature scheme for group communications without a shared distribution center. Future Generation Computer Systems 20(6), 1013–1021 (2004)
9. Abe, M., Ohkubo, M., Suzuki, K.: 1 out of n Signature from a Variety of Keys. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 415–423. Springer, Heidelberg (2002)
10. Hwa-Ching, H., Tung-Shou, C., Yu-Hsuen, L.: The ringed shadow image technology of visual cryptography by applying diverse rotating angles to bide the secret sharing. In: IEEE International Conference on Networking, Sensing and Control, 2004, vol. (2), pp. 996–1001. IEEE Press, Piscataway (2004)
11. Tochikubo, K., Uyematsu, T., Matsumoto, R.: Efficient Secret Sharing Schemes Based on Authorized Subsets. IEICE Transactions Special Section on Cryptography and Information Security E88-A(1), 322–326 (2005)
12. Hwang, M.S., Lin, E.J., Lin, I.C.: A practical (t,n) threshold proxy signature scheme based on the RSA cryptosystem. IEEE Transactions on Knowledge and Data Engering 15(5), 1552–1560 (2003)
13. Bellare, M., Micciancio, D., Warinschi, B.: Foundations of Group Signatures: Formal Denitions, Simpli¯ed Requirements, and a Construction based on General Assumptions. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656. Springer, Heidelberg (2003)
14. Malkin, T., Obana, S., Yung, M.: The Hierarchy of Key Evolving Signatures and a Characterization of Proxy Signatures. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 306–322. Springer, Heidelberg (2004)
15. Kobara, K., Imai, H.: On the channel capacityof narrow-band subliminal channels. In: Varadharajan, V., Mu, Y. (eds.) ICICS 1999. LNCS, vol. 1726, pp. 309–323. Springer, Heidelberg (1999)
16. Park, H.U., Lee, I.Y.: A Digital Nominative Proxy Signature Scheme for Mobile Communications. In: Qing, S., Okamoto, T., Zhou, J. (eds.) ICICS 2001. LNCS, vol. 2229, pp. 451–455. Springer, Heidelberg (2001)
17. Chang, T.-Y., Yang, C.-C., Hwang, M.-S.: Cryptanalysis of publicly veriableauthenticated encryption. IEICE Transactions on Fundamental E87-A(6), 1645–1646 (2004)
18. Zhang, F.G., Safavi-Naini, R., Susilo, W.: An Efficient Signature Scheme from Bilinear Pairings and Its Applications. In: Bao, F., Deng, R., Zhou, J. (eds.) PKC 2004. LNCS, vol. 2947, pp. 277–290. Springer, Heidelberg (2004)
19. Fan, H., Feng, D.: Theory and Method of Secure Protocols. Science Press (2003)