



Cryptanalysis of and improvement on the Hwang–Chen multi-proxy multi-signature schemes

Yuh-Dauh Lyuu^{a,b,1}, Ming-Luen Wu^{b,c,*}

^a Department of Computer Science and Information Engineering, Department of Finance, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, Taiwan

^b Department of Computer Science and Information Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, Taiwan

^c Department of Information Management, Chung-Yu Institute of Technology, No. 40, Yi-7th Road, Keelung, Taiwan

Abstract

Hwang and Chen recently proposed new multi-proxy multi-signature schemes that allow a group of authorized proxy signers to sign messages on behalf of a group of original signers. This paper shows that their schemes are insecure because a malicious proxy signer can forge a signature for a message secretly while participating in the message signing process with the other proxy signers. This paper then proposes a method to remove this weakness with only small computational overheads and without impairing the security of the original schemes.

© 2004 Elsevier Inc. All rights reserved.

Keywords: Multi-signatures; Proxy signatures; Cryptanalysis; Primitive roots

* Corresponding author. Address: Department of Computer Science and Information Engineering, P.O. Box 12-20, Keelung 201, Taiwan, ROC.

E-mail addresses: lyuu@csie.ntu.edu.tw (Y.-D. Lyuu), d5526009@csie.ntu.edu.tw (M.-L. Wu).

¹ The author was supported in part by NSC grant 92-2213-E-002-016.

1. Introduction

A proxy signature scheme allows an authorized person called the proxy signer to sign messages on behalf of the original signer. The concept of proxy signatures is first introduced by Mambo et al. [11,12] and further studied in [8,9,17,20]. In addition to these proxy signature schemes, various group-oriented proxy signatures have been introduced [3–7,16,18,19,21,22]. In a (t, n) threshold proxy signature scheme, the original signer can authorize n proxy signers such that only the cooperation of t or more of them is able to generate proxy signatures [3,4,16,19,22]. A multi-proxy signature scheme is a threshold proxy signature scheme in which only the cooperation of all the proxy signers can generate proxy signatures on behalf of the original signer [7]. Finally a proxy multi-signature scheme allows the group of original signers to authorize one person as their proxy signer [5,18,21].

Recently, by combining the notions of multi-proxy signature and proxy multi-signature, Hwang and Chen proposed a new type of group-oriented proxy signature scheme called multi-proxy multi-signature scheme [6]. In this signature scheme, the group of original signers (called the original signer group) can authorize a group of persons (called the proxy signer group) as their proxy signers, who sign messages on behalf of the original signer group. A multi-proxy multi-signature scheme satisfies the following two requirements: (1) only the cooperation of all the members in the original signer group can authorize a proxy signer group and (2) only the cooperation of all the members in the proxy signer group can sign messages. In Hwang and Chen's schemes, the original signers and proxy signers all cooperate to create a proxy certificate. Afterwards, the proxy certificate enables the proxy signers to work together in generating the multi-proxy multi-signatures of any messages. Hwang and Chen claim that their schemes are unforgettable even from insider attacks [6,10].

This paper will present an insider attack on the Hwang–Chen schemes that leads to forged signatures. With our attack, a malicious proxy signer can forge a multi-proxy multi-signature for a message secretly while participating in a normal message signing process with the other proxy signers. The signature is valid as if the other proxy signers had cosigned. To thwart this type of attack, a modification of the Hwang–Chen schemes is proposed. In the modified schemes, the original schemes' security is not impaired and the computation overheads are small.

The rest of this paper is organized as follows. In Section 2, we review the Hwang–Chen schemes. Then we present an attack that compromises the security of their schemes in Section 3. In Section 4, a modification of their scheme is proposed and analyzed. Section 5 concludes.

2. Review of the Hwang–Chen schemes [6]

Hwang and Chen proposed two multi-proxy multi-signature schemes: one has the help of a clerk, whereas the other does not. Both schemes use the same calculations to generate the proxy certificate and signatures. But the scheme without a clerk is more flexible than the one with a clerk in that the proxy signers rather than the clerk produce the signatures. We therefore review the scheme without a clerk in this section. Our attack also works against the scheme with a clerk.

The scheme without a clerk has two types of participants: the original signers $\{U_1, U_2, \dots, U_n\}$ and the proxy signers $\{P_1, P_2, \dots, P_m\}$. The scheme can be divided into four phases: system set-up, proxy certificate generation, multi-proxy multi-signature generation, and multi-proxy multi-signature verification. We describe each phase in the following.

2.1. System set-up

The system parameters and the corresponding notations are defined as follows.

p	a large public prime such that $p - 1$ has a large prime factor;
q	a large public prime factor of $p - 1$;
g	a public integer with order q in \mathbb{Z}_p ;
h	a public one-way hash function;
ID_{u_i}	the unique ID of the original signer U_i ;
ID_{p_j}	the unique ID of the proxy signer P_j ;
$x_{u_i} \in \mathbb{Z}_q^*$	the secret key of the original signer U_i ;
$y_{u_i} = g^{x_{u_i}} \bmod p$	the certified public key of the original signer U_i ;
$x_{p_j} \in \mathbb{Z}_q^*$	the secret key of the proxy signer P_j ;
$y_{p_j} = g^{x_{p_j}} \bmod p$	the certified public key of the proxy signer P_j ;
w	the proxy warrant that specifies the public proxy details such as ID_{u_i} , ID_{p_j} , y_{u_i} , and y_{p_j} .

2.2. Proxy certificate generation

In this phase, all proxy signers P_1, P_2, \dots, P_m cooperate with all original signers U_1, U_2, \dots, U_n to generate the proxy certificate (K, V) as follows.

Step A.1: Each original signer U_i selects a random integer $k_{u_i} \in \mathbb{Z}_q^*$, computes $K_{u_i} = g^{k_{u_i}} \bmod p$, and broadcasts K_{u_i} to the other $n - 1$ original signers and all m proxy signers. Each proxy signer P_j selects a random integer $k_{p_j} \in \mathbb{Z}_q^*$, computes $K_{p_j} = g^{k_{p_j}} \bmod p$, and broadcasts K_{p_j} to all n original signers and the other $m - 1$ proxy signers.

Step A.2: Every original signer U_i and every proxy signer P_j compute

$$K = \left(\prod_{i=1}^n K_{u_i} \right) \left(\prod_{j=1}^m K_{p_j} \right) \text{mod } p.$$

Step A.3: Each original signer U_i computes $v_{u_i} = h(w)x_{u_i}y_{u_i} + k_{u_i}K \text{mod } q$ and sends v_{u_i} to the other $n - 1$ original signers and all m proxy signers. Each proxy signer P_j computes $v_{p_j} = h(w)x_{p_j}y_{p_j} + k_{p_j}K \text{mod } q$ and sends v_{p_j} to all n original signers and the other $m - 1$ proxy signers.

Step A.4: Each proxy signer verifies the correctness of v_{u_i} with the equations $g^{v_{u_i}} \equiv (y_{u_i}^{y_{u_i}})^{h(w)} K_{u_i}^K \text{ (mod } p)$, $i = 1, 2, \dots, n$. He also verifies the correctness of v_{p_j} with the equations $g^{v_{p_j}} \equiv (y_{p_j}^{y_{p_j}})^{h(w)} K_{p_j}^K \text{ (mod } p)$, $j = 1, 2, \dots, m$. If any of the equations are violated, the phase fails.

Step A.5: If all the above equations hold, each proxy signer computes

$$V = \left(\sum_{i=1}^n v_{u_i} + \sum_{j=1}^m v_{p_j} \right) \text{mod } q.$$

The proxy certificate available to all the proxy signers is (K, V) .

2.3. Multi-proxy multi-signature generation

When the proxy signer group wants to sign a message M on behalf of the original signer group, the following steps are carried out.

Step B.1: Each proxy signer P_j randomly selects an integer $t_j \in \mathbb{Z}_q^*$.

Step B.2: Each proxy signer P_j computes $r_j = g^{t_j} \text{mod } p$ and broadcasts r_j to the other $m - 1$ proxy signers.

Step B.3: Each proxy signer P_j computes R and s_j , where

$$R = \prod_{j=1}^m r_j \text{mod } p,$$

$$s_j = (V_j + x_{p_j}y_{p_j}Rh(M)) \text{mod } q.$$

Step B.4: Each P_j broadcasts s_j to the other $m - 1$ proxy signers.

Step B.5: Each proxy signer P_j checks the validity of (r_j, s_j) by testing $g^{s_j} \equiv r_j^V y_{p_j}^{Ry_{p_j}h(M)} \text{ (mod } p)$, $j = 1, 2, \dots, m$. If all of the equations hold, each proxy signer computes

$$S = \sum_{i=1}^m s_j \text{mod } q.$$

The multi-proxy multi-signature of message M is (w, K, V, M, R, S) .

2.4. Multi-proxy multi-signature verification

The multi-proxy multi-signature (w, K, V, M, R, S) is verified in two steps.

Step C.1: Verify the warrant w and the proxy certificate (K, V) by testing

$$g^V \stackrel{?}{\equiv} K^K \left(\prod_{i=1}^n y_{u_i}^{y_{u_i}} \right)^{h(w)} \left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{h(w)} \pmod{p}.$$

Step C.2: Check the correctness of (R, S) by testing

$$g^S \stackrel{?}{\equiv} R^V \left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{Rh(M)} \pmod{p}.$$

Accept the signature if both equations hold.

3. Our insider attack

We now present an attack on the Hwang–Chen schemes. Let the proxy signer P_1 be malicious throughout this section. We will show how P_1 can forge a multi-proxy multi-signature for a secret message M' while participating with the other proxy signers in signing another message M .

P_1 takes all the necessary Steps B.1–B.5 in the multi-proxy multi-signature generation phase. Let $a = h(M')h(M)^{-1} \pmod{q}$ and a^{-1} be the multiplicative inverse of a modulo q , i.e.,

$$a^{-1} = h(M')^{-1}h(M) \pmod{q}.$$

In Step B.1, P_1 randomly selects an integer $t_1 \in \mathbb{Z}_q^*$ as before. In Step B.2, P_1 waits for other proxy signers' r_2, r_3, \dots, r_m . He then privately computes

$$R' = g^{t_1} \prod_{j=2}^m r_j \pmod{p}$$

and solves for R such that

$$Rh(M) \equiv R'h(M') \pmod{q}. \tag{1}$$

Note that $a^{-1}R \equiv R' \pmod{q}$. P_1 now solves for r_1 satisfying

$$r_1 \prod_{j=2}^m r_j \equiv R \pmod{p},$$

and broadcasts this r_1 in Step B.2. Note that r_1 is no longer random as in the original scheme. In Step B.3, each proxy signer P_j computes $R = \prod_{j=1}^m r_j \pmod{p}$

and s_j . Then each proxy signer P_j except P_1 broadcasts s_j . After P_1 receives s_2, s_3, \dots, s_m from the other proxy signers, he computes

$$S = \sum_{j=1}^m s_j \text{ mod } q.$$

Now the forged signature (w, K, V, M', R', S) is completed. Note that P_1 never sends out his s_1 as required. He can attribute the failure to hardware or communications faults to diffuse suspicion.

This multi-proxy multi-signature (w, K, V, M', R', S) is valid because

$$g^V \equiv K^K \left(\prod_{i=1}^n y_{u_i}^{y_{u_i}} \prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{h(w)} \pmod{p},$$

and

$$\begin{aligned} g^S &\equiv g^{\sum_{j=1}^m s_j \text{ mod } q} \equiv g^{\sum_{j=1}^m (V_j + x_{p_j} y_{p_j} R h(M)) \text{ mod } q} \equiv \left(g^{t_1} \prod_{j=2}^m r_j \right)^V \left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{R h(M)} \\ &\equiv (R')^V \left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{R' h(M')} \pmod{p}. \end{aligned}$$

We remark that to forge a signature, P_1 must find the R such that

$$\left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{R h(M)} \equiv \left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{R' h(M')} \pmod{p},$$

i.e., P_1 must solve for the R satisfying Eq. (1).

4. Modifications to foil the attack

In this section we propose modifications of the Hwang–Chen multi-proxy multi-signature schemes to foil the attack and discuss the security and performance of the modified scheme.

4.1. The modified scheme

We will modify the Hwang and Chen scheme without a clerk. The same modifications can be applied to the scheme with a clerk and yield the same results. Hence we focus on the former case.

In our modified scheme, the participants and the notations are identical to those in Section 2. There are also four phases: system set-up, proxy certificate

generation, multi-proxy multi-signature generation, and multi-proxy multi-signature verification. The basic differences are the choice of moduli and the replacement of R with R^2 in the calculations.

4.1.1. System set-up

The system parameters and the corresponding notations are defined as follows:

- $N = p_1p_2$ a public odd integer where p_i are large primes such that each $p_i - 1$ has a large prime factor q_i ;
- $Q = q_1q_2$ a public integer;
- g a public integer with order Q in \mathbb{Z}_N^* ;
- h a public one-way hash function;
- ID_{u_i} the unique ID of the original signer U_i ;
- ID_{p_j} the unique ID of the proxy signer P_j ;
- $x_{u_i} \in \mathbb{Z}_Q^*$ the secret key of the original signer U_i ;
- $y_{u_i} = g^{x_{u_i}} \text{ mod } N$ the certified public key of the original signer U_i ;
- $x_{p_j} \in \mathbb{Z}_Q^*$ the secret key of the proxy signer P_j ;
- $y_{p_j} = g^{x_{p_j}} \text{ mod } N$ the certified public key of the proxy signer P_j ;
- w the proxy warrant that specifies the public proxy details such as ID_{u_i} , ID_{p_j} , y_{u_i} , and y_{p_j} .

The requirements for p_i are identical to those for p under the original Hwang–Chen schemes. N should be chosen such that factoring N and Q and solving the discrete logarithm problem in \mathbb{Z}_N^* are intractable.

We next show that obtaining a g with order $Q = q_1q_2$ is computationally easy. But let’s review some notations first. Let $\phi(N)$ denote Euler’s phi function, which gives the number of positive integers $j \in \{1, 2, \dots, N - 1\}$ that are relatively prime to N . The order of g modulo N is denoted by ord_{Ng} or simply $\text{ord}(g)$ if N is understood. If g and p are relatively prime integers with $p > 0$ and if $\text{ord}_p g = \phi(p)$, then g is called a primitive root modulo p . A universal exponent of N is a positive integer u such that $g^u \equiv 1 \text{ mod } N$ for all g relatively prime to N . The minimal universal exponent of N is denoted by $\lambda(N)$. The following known facts are needed for our purpose [1,15].

Fact 4.1. Let N be an odd positive integer with prime factorization $N = p_1p_2$. Then the following hold.

1. $\lambda(N) = \text{lcm}(\phi(p_1), \phi(p_2))$.
2. Let r_i be a primitive root modulo p_i , $i = 1, 2$. The solution of the simultaneous congruences $x \equiv r_i \text{ (mod } p_i)$, $i = 1, 2$, is an integer with order $\lambda(N)$ modulo N .

Fact 4.2. Let $G = \langle x \rangle$ be a cyclic group generated by x . If $\text{ord}(x) = d$ and if ℓ is a positive integer, then

$$\text{ord}(x^\ell) = \frac{d}{\text{gcd}(d, \ell)}.$$

We find primitive roots modulo $p_i, i = 1, 2$, using, e.g., the efficient Algorithm 4.80 of [13]. Suppose $p_i = a_i q_i + 1$. Then $\text{lcm}(\phi(p_1), \phi(p_2)) = \text{lcm}(a_1 q_1, a_2 q_2) = \ell q_1 q_2$ for some integer ℓ . By Fact 4.1, we can use the Chinese remainder algorithm to compute a g_0 with order $\lambda(N) = \text{lcm}(\phi(p_1), \phi(p_2)) = \ell q_1 q_2$. By Fact 4.2, $g_0^\ell \text{ mod } N$ has order $\frac{\ell q_1 q_2}{\text{gcd}(\ell q_1 q_2, \ell)} = q_1 q_2$. We will take $g = g_0^\ell \text{ mod } N$.

4.1.2. Proxy certificate generation

This phase is the same as that of the Hwang–Chen schemes except that N replaces p and Q replaces q .

4.1.3. Multi-proxy multi-signature generation

When the proxy signer group wants to sign a message M on behalf of the original signer group, the following steps are carried out.

- Step B.1: Each proxy signer P_j randomly selects an integer $t_j \in \mathbb{Z}_Q^*$.
- Step B.2: Each proxy signer P_j computes $r_j = g^{t_j} \text{ mod } N$ and broadcasts r_j to the other $m - 1$ proxy signers.
- Step B.3: Each proxy signer P_j computes numbers R and s_j , where

$$R = \prod_{j=1}^m r_j \text{ mod } N,$$

$$s_j = (V_j + x_{p_j} y_{p_j} R^2 h(M)) \text{ mod } Q.$$

- Step B.4: Each P_j broadcasts s_j to the other $m - 1$ proxy signers.
- Step B.5: Each proxy signer P_j checks the validity of (r_j, s_j) by testing $g^{s_j} \equiv r_j^V y_{p_j}^{R^2 y_{p_j} h(M)} \pmod{N}, j = 1, 2, \dots, m$. If all the equations hold, each proxy signer computes

$$S = \sum_{i=1}^m s_i \text{ mod } Q.$$

The multi-proxy multi-signature of message M is (w, K, V, M, R, S) .

4.1.4. Multi-proxy multi-signature verification

The multi-proxy multi-signature (w, K, V, M, R, S) is verified in two steps.

Step C.1: Verify the warrant w and the proxy certificate (K, V) by testing

$$g^V \stackrel{?}{\equiv} K^K \left(\prod_{i=1}^n y_{u_i}^{y_{u_i}} \right)^{h(w)} \left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{h(w)} \pmod{N}.$$

Step C.2: Check the correctness of (R, S) by testing

$$g^S \stackrel{?}{\equiv} R^V \left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{R^2 h(M)} \pmod{N}.$$

Accept the signature if both equations hold.

4.2. Discussions

We first discuss the security of our modified scheme. The security of the modified scheme is based on the following intractability assumptions:

1. The discrete logarithm problem is hard.
2. Solving for x in the equation $x^x \equiv a \pmod{N}$ for a constant a is hard [2].
3. The factoring problem is hard [14].

Assumptions 1 and 2 are necessary by Hwang–Chen’s analysis. Assumption 3 allows our modified scheme to resist the proposed attack. The reason is as follows. In the attack, $a = h(M')h(M)^{-1} \pmod{Q}$ and $R' = g^{t_1} \prod_{j=2}^m r_j \pmod{N}$. To forge a signature with the attack, a malicious proxy signer must find an R such that

$$\left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{R^2 h(M)} \equiv \left(\prod_{j=1}^m y_{p_j}^{y_{p_j}} \right)^{R'^2 h(M')} \pmod{N}.$$

That is, the malicious proxy signer must solve for R satisfying $R'^2 h(M') \equiv R^2 h(M) \pmod{Q}$. As $h(M') \equiv ah(M) \pmod{Q}$, the malicious proxy signer must compute a square root of $aR'^2 \pmod{Q}$. Because factoring Q is infeasible, computing a square root of $aR'^2 \pmod{Q}$ is infeasible. As a result, forging a signature with the attack is hard.

Now we briefly discuss the performance of the modified scheme. Compared with the Hwang–Chen scheme without a clerk, our modified scheme uses different moduli and group order. In addition, our modified scheme and the Hwang–Chen scheme differ slightly in Steps B.3, B.5, and C.2. In our scheme, each proxy signer P_j in Steps B.3 and B.5 uses R^2 instead of R , so does the verifier in Step C.2. To express the computation and communication costs more clearly, some symbols are defined in Table 1. The computation costs are listed in Table 2 and the communication costs in Table 3. For comparison, we also

Table 1
The definitions of the symbols

Symbol	Definition
T_m	Time to execute one modular multiplication
T_e	Time to execute one modular exponentiation
T_h	Time to execute the one-way hash function h
$ I $	Size of integer I

Table 2
Computation costs

Phases	Hwang–Chen scheme without a clerk	Our modified scheme
Proxy certificate generation	$(3m^2 + 3n^2 + 6mn - 2n - 2m)T_e + 2(n + m)^2T_m + (n + m)T_h$	$(3m^2 + 3n^2 + 6mn - 2n - 2m)T_e + 2(n + m)^2T_m + (n + m)T_h$
Multi-proxy multi-signature generation	$(3m^2 - 2m)T_e + 3m^2T_m + mT_h$	$(3m^2 - 2m)T_e + (3m^2 + m)T_m + mT_h$
Multi-proxy multi-signature verification	$6T_e + 3T_m + 2T_h$	$6T_e + 4T_m + 2T_h$

Table 3
Communication costs

Phases	Hwang–Chen scheme without a clerk	Our modified scheme
Proxy certificate generation	$(n + m - 1)(n + m)(p + q)$	$(n + m - 1)(n + m)(N + Q)$
Multi-proxy multi-signature generation	$(m^2 - m + 2)(p + q) + w + M $	$(m^2 - m + 2)(N + Q) + w + M $

list the computation and communication costs of the Hwang–Chen scheme in the same tables. As in Hwang–Chen’s paper, in Table 2 we do not count the computation costs of modular addition and modular subtraction because their computation times are much less than those of T_m or T_e defined in Table 1. Also we do not count the costs of the following calculations: $x_{u_i}y_{u_i}$, $x_{p_j}y_{p_j}$, $y_{u_i}^{y_{p_j}}$, $y_{p_j}^{y_{u_i}}$, $\prod_{i=1}^n y_{u_i}^{y_{p_j}}$, and $\prod_{j=1}^m y_{p_j}^{y_{u_i}}$. This is because they are computed once and for all.

5. Conclusions

In this paper, we present an attack that exposes a weakness of Hwang and Chen’s schemes [6]. In addition, we propose improvements of their schemes to overcome this weakness without compromising the original schemes’ security. The extra computation overheads are minimal.

References

- [1] G.H. Hardy, E.M. Wright, *An Introduction to the Theory of Numbers*, fifth ed., Oxford University Press, New York, 1979.
- [2] L. Harn, Group-oriented (t, n) threshold digital signature scheme and digital multisignature, *IEE Proceedings—Computers and Digital Techniques* 141 (5) (1994) 307–313.
- [3] C.-L. Hsu, T.-S. Wu, T.-C. Wu, New repudiable threshold signature scheme with known signers, *The Journal of Systems and Software* 58 (2001) 119–124.
- [4] M.-S. Hwang, I.-C. Lin, E.J.-L. Lu, A secure nonrepudiable threshold proxy signature scheme with known signers, *Informatica* 11 (2) (2000) 137–144.
- [5] S.-J. Hwang, C.-C. Chen, A new proxy multi-signature scheme, in: *International Workshop on Cryptology and Network Security*, Taiwan, 2001, pp. 199–204.
- [6] S.-J. Hwang, C.-C. Chen, New multi-proxy multi-signature schemes, *Applied Mathematics and Computation* 147 (2004) 57–67.
- [7] S.-J. Hwang, C.-H. Shi, A simple multi-proxy signature scheme, in: *Proceedings of the Tenth National Conference on Information Security*, Taiwan, 2000, pp. 134–138.
- [8] S. Kim, S. Park, D. Won, Proxy signatures, revisited In *Information security and cryptology—ICISC'97LNCS*, vol. 1334, Springer-Verlag, Berlin, 1997, pp. 223–232.
- [9] N.-Y. Lee, T. Hwang, and C.-H. Wang, On Zhang's nonrepudiable proxy signature schemes, in: *Third Australasian Conference, ACISP'98*, 1998, pp. 415–422.
- [10] Z.-C. Li, L.C.K. Hui, K.P. Chow, C.F. Chong, W.W. Tsang, H.W. Chan, Cryptanalysis of Harn digital multisignature scheme with distinguished signing authorities, *Electronics Letters* 36 (4) (2000) 314–315.
- [11] M. Mambo, K. Usuda, E. Okamoto, Proxy signatures: delegation of the power to sign message, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E79-A* (9) (1996) 1338–1353.
- [12] M. Mambo, K. Usuda, E. Okamoto, Proxy signatures for delegation signing operation, in: *CCS '96, Proceedings of the 3rd ACM Conference on Computer and Communications Security*, 1996, pp. 48–57.
- [13] A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, FL, 1997.
- [14] M.O. Rabin, Digital signatures and public-key functions as intractable as factorization. Technical Report MIT/LCS/TR-212, MIT Laboratory for Computer Science, January 1979.
- [15] K.H. Rosen, *Elementary Number Theory and Its Applications*, third ed., Addison Wesley, Reading, MA, 1993.
- [16] H.-M. Sun, An efficient nonrepudiable threshold proxy signature scheme with known signers, *Computer Communications* 22 (1999) 717–722.
- [17] H.-M. Sun, Design of time-stamped proxy signatures with traceable receivers, *IEE Proceedings—Computers and Digital Techniques* 147 (6) (2000) 462–466.
- [18] H.-M. Sun, On proxy (multi-)signature schemes, in: *2000 International Computer Symposium*, Taiwan, 2000, pp. 65–72.
- [19] H.-M. Sun, N.-Y. Lee, T. Hwang, Threshold proxy signatures, *IEE Proceedings—Computers and Digital Techniques* 146 (5) (1999) 259–263.
- [20] S.-M. Yen, C.-P. Hung, and Y.-Y. Lee, Remarks on some proxy signature scheme, in: *2000 International Computer Symposium*, Taiwan, 2000, pp. 54–59.
- [21] L. Yi, G. Bai, G. Xiao, Proxy multi-signature scheme: A new type of proxy signature scheme, *Electronics Letters* 36 (6) (2000) 527–528.
- [22] K. Zhang, Threshold proxy signature schemes, *1997 Information Security Workshop*, Japan, 1997, pp. 191–197.