Message Recovery via an Efficient Multi-Proxy Signature With Self-certified Keys

Manoj Kumar Chande[1], Cheng-Chi Lee[2,3], Chun-Ta Li[4]

(Corresponding author: Cheng-Chi Lee)

School of Studies in Mathematics, Pt. Ravishankar Shukla University[1]

Raipur, 492010, Chhattisgarh, India

E-mail: manojkumarchande@gmail.com

Department of Library and Information Science, Fu Jen Catholic University[2]

510 Jhongjheng Road, Taipei 24205, Taiwan, R.O.C.

E-mail: cclee@blue.lins.fju.edu.tw

Department of Photonics and Communication Engineering, Asia University[3]

Wufeng Shiang, Taichung, Taiwan 413, R.O.C.

Department of Information Management, Tainan University of Technology[4]

No. 529, Zhongzheng Road, Tainan City 71002, Taiwan, R.O.C.

E-mail: th0040@mail.tut.edu.tw

(Received Mar. 31, 2016; revised and accepted May 19 & June 10, 2016)

Abstract

Multi-proxy signature (MPS) scheme makes a very important branch of the proxy signature scheme family, as they are applicable in many practical situations. The MPS scheme enables the actual signer to pass on their signing authority to plural proxy signers, where each proxy/delegated signer should contribute together to create a genuine MPS to make the whole thing work. In this work, we shall present an efficient MPS scheme that apply self-certified key and the notion of message recovery. The major advantage of our scheme is that the verification of the public keys, the verification of MPS, and recovery of the message can be carried out simultaneously. This reduces the computation cost and communication load dramatically. The security analysis of the proposed scheme includes thorough discussions over the security of the secret keys, the legitimacy of the public key of the signer's, along with unforgeability of our MPS scheme (MPSS). The performance analysis of our MPSS, reflects that our scheme, has an edge regarding computational complexity, over the schemes given in Wu et al.'s and Xie et al.'s.

Keywords: Discrete logarithm problem, message recovery, multi-proxy signature, proxy signature, self-certified key

1 Introduction

What is a proxy signature scheme? By definition, this signature scheme enables the other person called proxy signer to sign in place of actual signer, with due permission [5, 10, 25]. Mambo et al. [15, 16] first brought the design of proxy signature from some authorized proxy person. Since then, enormous researches have focused on refining this specific signature itself and on making them applicable to as many real-life situations as possible [1]. Among the possibilities explored was the question of how to transfer the power of signing to plural proxy signer at a time, and in 2000, Huang and Shi [6] answered the question by offering their MPS scheme, as an extension of the fundamental proxy or delegated signature mechanism. After that, many researchers have developed and presented their own variants [2, 7, 13, 14, 19, 20, 24, 29, 30], of the MPSS (MPS scheme). Typically in a MPSS, commonly the following three entities are involved: the original/actual signer, two or more proxy signers, and recipient of signature. Please note that all the proxy signers have to jointly create the MPSS and this makes the major difference between a MPSS and a fundamental proxy signature scheme.

To an adversary, any form of digital transaction can be a target for attack. For example, with a forged public key, an attacker can try to forge as the original signer or a proxy signer. To prevent forgery attacks from taking effect, it is a good idea to authenticate the public key of all the entities involved before they participate in any part of the cryptographic processes. A common practice to do the job here is to use a certificate-based public key cryptosystem, where any legal user or verifier can confirm the public key authenticity and the verification of information regarding identity of the signer by checking the certificate issued to each signer by the certificate authority (CA) [8, 21]. However, certificate verification processes considerably increase both the computation cost and the communication load. In real time applications, in particular, when many users are trying to sign documents at the same time, it is extremely demanding for the system to handle the verification of multiple certificates simultaneously. To solve this problem, Shamir [22] presents a new cryptosystem based on the identity (ID-based) scheme. In such a system, the signer can be recognized through his public key. This way, certificates are no longer necessary, and therefore no certificate verification processes are needed. The shortcoming of this approach, however, is that the CA has knowledge of secret key of every signer, as the signer register himself. This may give the CA, a fair chance to pretend to be a genuine user. This is possible by creating a legitimate pair of keys for that user and no one identify that actually CA generates the pair of keys. In other words, public key verification remained a problem.

Girault [3] introduced the self-certified public key concept. In Girault's design, the registered user gets to determine their own secret key, while the public key for each user is generated by CA. In comparison with the certificate-based approach, this system runs on a much lower computation cost, and the communication load is also lighter [12, 23]. The validity of a public key is checked when a user participates in signature schemes where self-certified public keys are used. If the signature or public key of the user fails in verification process than the user's access will be denied.

In 1994, Nyberg et al. [17] offered the first signature with the ability of message recovery. In Nyberg et al.'s scheme, the message is sent along with the signature and is then recovered by the verifier. Since no hashing of message is required, the consumption of storage space and communication bandwidth is low. The security of their scheme relies on the discrete logarithm problem (DLP). In this kind of schemes, only a legitimate signer can broadcast the authentic signature corresponds to the message to a signature's verifier, and the verifier can obtain the message and verify the authenticity of the signature. This way, the communication overhead can be effectively reduced.

Wu, Hsu, and Lin (WHL) [27] proposed couple of MPS scheme, and their security relies on DLP and the elliptic curve discrete logarithm problem (ECDLP) respectively. They combined the concept of message recovery and the self-certified public key. Later, in 2012, Xie [28] showed that WHL scheme [27] is vulnerable to a warrant attack by proxy signer via revision of original warrant. This attack through warrant revision can launched either by the proxy or the actual signer. To fix the problem, Xie presents a provably secure signature scheme resists a warrant attack and an adaptive chosen message attack under existential forgery.

Inspired by the brilliant earlier works, we have also developed an efficient MPS scheme, by applying self-certified public keys and our scheme provides message recovery as well. The remaining of our work is managed as follows: To begin with, the proposed scheme will be presented in detail in next section, followed by Section 3, in which the security analysis of our scheme is given. The performance analysis is given in Section 4. Finally, we conclude our work in last section.

2 The Proposed MPS Scheme

The details of our proposed MPS scheme is given in this part. Let's first define some notations and parameters in Table 1 that we are going to use throughout this paper.

The CA generates p, q, g, and β as system parameters and makes them public but keeps α secret. The CA also assists registered users to create their secret and public key pairs. The proposed MPS scheme has the following phases: (1) User Registration Phase, (2) Delegation Parameter Generation Phase, (3) Multi-Proxy Signature Generation Phase, and (4) Signature Verification and Message Recovery Phase. The details of the above phases are given below:

1) User Registration Phase.

   Suppose a user U_i with identity ID_i wishes to register with CA. To serve the purpose, he/she needs to present keys namely a secret key and an openly accessible public key paired up. Self-certified keys are generated as follows:

   a. Each user U_i selects a random number $a_i \in Z_q^*$ as their master key and computes

   $$v_i = g^{h(a_i \| ID_i)} \bmod p \quad (1)$$

   and then sends it to CA over a secure channel.

   b. Upon receiving (v_i, ID_i) from U_i, the CA chooses an integer $t_i \in Z_q^*$, which varies with time and computes the U_i's public key y_i and the witness w_i as follows:

   $$y_i = v_i \cdot g^{t_i} - h(ID_i) \bmod p \quad (2)$$
   $$w_i = t_i + \alpha \cdot \{y_i + h(ID_i)\} \bmod q \quad (3)$$

   for each U_i and sends (y_i, w_i) to them respectively.

   c. Upon receiving (y_i, w_i), each U_i computes his secret key

   $$x_i = w_i + h(a_i \| ID_i) \quad (4)$$

   and checks the validity of y_i, through the following equation

   $$\begin{aligned} g^{x_i} &= \{y_i + h(ID_i)\} \cdot \beta^{y_i + h(ID_i)} \bmod p \\ &= Y_i \bmod p. \end{aligned} \quad (5)$$

Table 1: Notations

Notation	Description
(p, q)	Large primes, with $q \mid p-1$.
g	Generator with order q, over $GF(p)$.
m_w	Message warrant.
$h(\cdot)$	One-way hash function [4, 9, 11].
(α, β)	The private and public key pair for CA, with $\beta = g^{\alpha} \bmod p$.
U_o	Denote the original/actual signer.
U_i	Denote the proxy/delegated signer, where $i = 1, 2, ...N$.
G	Group of proxy signers.
(x_i, y_i)	For signer the key pair of private and public key, where $i = 1, 2, ...N$.
ID_i	Represents identity of the signer, where $i = 0, 1, 2, ...N$.

This verification can be done as follows:

$$\begin{aligned} g^{x_i} &= g^{t_i + \alpha\{y_i + h(ID_i)\} + h(a_i \| ID_i)} \bmod p \\ &= g^{t_i} \cdot g^{\alpha\{y_i + h(ID_i)\}} \cdot g^{h(a_i \| ID_i)} \bmod p \\ &= v_i \cdot g^{t_i} \cdot \beta^{y_i + h(ID_i)} \bmod p \\ &= \{y_i + h(ID_i)\} \cdot \beta^{y_i + h(ID_i)} \bmod p \\ &= Y_i \bmod p. \end{aligned}$$

2) Delegation Parameter Generation Phase.

Now U_o wishes to transfer his authority of signing to N proxy signers G $= \{U_1, U_2, ...U_N\}$. U_o and U_i take the following steps to do the job:

a. U_o chooses a random integer $k_i \in Z_q^*$ and calculates

$$K_i = g^{k_i} \bmod p \tag{6}$$

$$K = \prod_{i=1}^{N} K_i \bmod p \tag{7}$$

$$H = h\big(\beta^{\sum_{i=0}^{N}(y_i + h(ID_i))} \cdot \prod_{i=0}^{N}(y_i + h(ID_i)) \| m_w \| K\big) \tag{8}$$

$$\sigma_i = x_o \cdot N^{-1} \cdot H + k_i \bmod q \tag{9}$$

b. U_o transmits (σ_i, m_w) to each $U_i \in$ G and broadcasts (K_i, K, H).

c. After getting (σ_i, m_w) from U_o, each $U_i \in$ G verifies its authenticity through the equation

$$g^{\sigma_i} = \big(Y_o\big)^{N^{-1} \cdot H} \cdot K_i \bmod p$$

If this equation checks out, then only U_i agrees to his proxy share.

3) Multi-Proxy Signature Generation Phase

To generate a signature for message M, as an alternative of U_o, each $U_i \in G$ carries out the following calculations:

a. Each $U_i \in G$ selects a random integer value $b_i \in Z_q^*$ and evaluates

$$c_i = g^{b_i} \bmod p, \tag{10}$$

then transmits c_i to other users in group G.

b. Each U_i computes

$$c = \{M \| h(M)\} \cdot \prod_{j=1}^{N} c_j \bmod p$$

$$\rho_i = b_i + (\sigma_i + x_i \cdot H) \cdot h(m_w \| c \| K) \bmod q \tag{11}$$

and sends ρ_i to other members in G.

c. Now each $U_i \in$ G has a collection of (c_j, ρ_j) received from all the other members of G. U_i checks the validity by computing

$$c_j \cdot \Big[\big(Y_o\big)^{N^{-1} \cdot H} \cdot \big(Y_j\big)^{H} \cdot K_j\Big]^{h(m_w \| c \| K)} = g^{\rho_j} \bmod p$$

if the above equation checks out, then U_i computes

$$\rho = \sum_{j=1}^{N} \rho_j \bmod q$$

Now the multi-proxy signature (K, c, ρ, m_w, H) is completed.

4) Signature Verification and Message Recovery Phase.

The verifier confirms the authenticity of the generated signature, through the equation

$$M \| h(M) = c \cdot g^{-\rho} \cdot \Big[\prod_{i=0}^{n} \big(Y_i\big)^{H} \cdot K\Big]^{h(m_w \| c \| K)} \bmod p. \tag{12}$$

Now with this recovered message M and its hash value, the verifier can ensure the authenticity of both M and the generated signature. The verification equation involves the public key's of both the proxy and actual signers, which can be automatically verified. This way, all three tasks, namely verification of public key, verification of signature, and recovery of message, can be completed in one stroke.

3 Security Analysis

This section serves to check the security aspects of our MPS scheme. The security of our scheme can be divided into three parts: safety of private keys, legitimacy of signers' public keys, and unforgeability of signatures.

1) Safety of private keys.

   a. Safety of private key (α) of CA.
   Suppose an adversary is looking to obtain CA's secret key α, which lies under the protection of DLP [18, 26]. To get α from Equation (3), the adversary faces great difficulty because of the lack of knowledge of the time variant secret t_i, which is only known to CA. It can be seen from Equation (2) that t_i is secure under DLP.

   b. Safety of secret key (x_i) of signer i.
   The secret key x_i of signer i is generated through the conduction of Equation (4), which depends on the hash value $h(a_i \| ID_i)$. It can be clearly from Equation (1) that the hash value is secure under the protection of DLP.
   Let an adversary or some delegated signers attempt to get the secret key x_o of actual signer U_o from Equation (9). However, it is not feasible for them due to unknown value k_i from Equation (6) and this k_i is secure because of DLP.

   c. Infeasible to obtain secret keys from public keys.
   It is not possible for an adversary to derive secret key of the actual signer U_o or any delegated signer U_i through intercepted data (c_i, ρ_i) or from a genuine multi-proxy signature (K, c, ρ, m_w, H). As we can see, with the value of σ_i (see Equation (9)) substituted into Equation (11), we come to

   $$\rho_i = b_i + \{(x_o \cdot N^{-1} \cdot H + k_i) + x_i \cdot H\} \cdot h(m_w \| c \| K) \bmod q,$$

   where there are still two unknowns parameters k_i and b_i securely under the protection of DLP (see Equations (6) and (10)). Therefore, there is no way an adversary can derive any secret key x_o or x_i from public data.

2) Legitimacy of signers' public keys.

   The secret key x_i, identity ID_i, and public key y_i must satisfy the verification Equation (5). In other words, for any fake secret key x'_i, fake identity ID'_i, and fake public key y'_i to take effect, all three must pass the test of Equation (5). An adversary can create a fake value ID'_i and randomly chooses private key x'_i at will, but to come by a public key y'_i to make the trio work is extremely difficult due to the obstruction of DLP. Alternatively, if the adversary tries to fix the public key y'_i and identity ID'_i, then again DLP will get in the way and nullify the adversary's attempt to derive an effective secret key x'_i.

   Lastly, if the adversary tries another route to come by a valid identity ID'_i with the made-up duo of fixed keys x'_i, y'_i, the attempt will still fail because of the unbreakable reversal of OWHF [4, 9, 11].

3) Unforgeability of signatures.

   Suppose an adversary is looking to reuse a genuine multi-proxy signature (K, c, ρ, m_w, H) to illegally sign the message M'. To do the job, the adversary has to find an effective ρ, which is difficult due to the obstruction of DLP (see Equation (12)).

   On the other hand, in case an adversary attempted to obtain message M by using (K, c, ρ, m_w, H), then the adversary would have to overcome the reversal of OWHF.

Then, in the following passages, we shall demonstrate that how our MPSS fulfil all fundamental security properties including (1) Identifiability, (2) Prevention of misuse, (3) Unforgeability, (4) Undeniability, and (5) Verifiability.

1) Identifiablity.

   The multi-proxy signature (K, c, ρ, m_w, H) contains the message warrant m_w, by which the verifier can identify the proxy signer and actual signer.

2) Prevention of misuse.

   The warrant m_w carries a lot of information with it including type of delegation, delegation duration, as well as indication of which message is assigned to the proxy signers for signing. Therefore, the proxy signers cannot mistakenly sign a message they are not authorized by the actual signer to sign.

3) Unforgeability.

   The actual signer U_o is not able generate a valid MPS, because there is no way for U_o to collect the private keys of all the delegated signers. On the other hand, any delegated signer or any other person cannot counterfeit a MPS either due to the lack of the actual signer's private key, which is protected due to intractability of DLP.

4) Undeniability.

   The components c and ρ of the proposed signature (K, c, ρ, m_w, H) are collectively completed by all the proxy signers, and therefore no $U_i \in G$ can deny his signature.

5) Verifiability.

   With the correctness of the verification confirmed, the verifier can authenticate the signature and identify, whether the signed message corresponds to the proxy warrant.

Table 2: Computational complexity comparison

Phases	WHL [27]	Xie's [28]	Our scheme
Registration	$4nT_e + 5nT_m + 5nT_h + nT_i$	$4nT_e + 5nT_m + 5nT_h + nT_i$	$4nT_e + 3nT_m + 2nT_h$
Proxy Key Generation	$5nT_e + 5nT_m + (3n+1)T_h + (n+1)T_i$	$(4n+1)T_e + (7n+3)T_m + (4n+1)T_h + (n+1)T_i$	$(4n+1)T_e + (5n+2)T_m + (2n+1)T_h + (n+1)T_i$
Multi Proxy Sign Gen	$(5n^2 - 3n)T_e + (6n^2 - 4n)T_m + 2n^2T_h$	$(4n^2 - 3n)T_e + (6n^2 - 3n)T_m + 2n^2T_h$	$(4n^2 - 3n)T_e + (5n^2 - 2n)T_m + (n^2+1)T_h$
Signature Verification	$4T_e + (2n+5)T_m + (2n+5)T_h$	$4T_e + (2n+5)T_m + (2n+4)T_h + T_i$	$4T_e + (n+4)T_m + (n+2)T_h + T_i$
Total Cost	$(5n^2 + 6n + 4)T_e + (6n^2 + 8n + 5)T_m + (2n^2 + 10n + 6)T_h + (2n+1)T_i$	$(4n^2 + 5n + 4)T_e + (6n^2 + 11n + 8)T_m + (2n^2 + 11n + 8)T_h + (2n+1)T_i$	$(4n^2 + 5n + 5)T_e + (5n^2 + 6n + 6)T_m + (n^2 + 5n + 4)T_h + (n+2)T_i$

Table 3: Communication cost comparison

Phase	WHL [27]	Xie's [28]	Our scheme
Proxy Key Generation	$(n+1) \cdot \lvert p \rvert + 2n \cdot \lvert q \rvert$	$(n+1) \cdot \lvert p \rvert + (2n+1) \cdot \lvert q \rvert$	$(n+1) \cdot \lvert p \rvert + (2n+1) \cdot \lvert q \rvert$
Multi-proxy Sign Gen	$n \cdot (\lvert p \rvert + \lvert q \rvert)$	$n \cdot (\lvert p \rvert + \lvert q \rvert)$	$n \cdot (\lvert p \rvert + \lvert q \rvert)$
Signature Verification	$2 \cdot \lvert p \rvert + 3 \cdot \lvert q \rvert$	$2 \cdot (\lvert p \rvert + \lvert q \rvert)$	$2 \cdot (\lvert p \rvert + \lvert q \rvert)$
Total	$(2n+3) \cdot \lvert p \rvert + (3n+3) \cdot \lvert q \rvert$	$(2n+3) \cdot \lvert p \rvert + (3n+3) \cdot \lvert q \rvert$	$(2n+3) \cdot \lvert p \rvert + (3n+3) \cdot \lvert q \rvert$

4 Performance Analysis

Now we shall see comparison of the complexity of the proposed MPSS with [27] and [28]. We do not consider the complexity of addition and subtraction operations as they are negligible.

As Table 2 shows, the proposed scheme is obviously superior to the other two schemes as far as computational complexity is concerned.

As Table 3 shows, the three schemes have the same total communication cost and therefore are equally efficient in this matter.

5 Conclusion

In this paper, we present a new MPSS using self-certified public keys. The security analysis has established the security of the secret keys, the genuineness of the public key of signers, as well as the unforgeability of the proposed scheme. Furthermore, the performance analysis has proven that the new scheme has an edge over the WHL scheme and Xie's scheme with respect to the computational load.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions. In addition, this research was partially supported by the Ministry of Science and Technology, Taiwan, R.O.C., under contract no.: MOST 104-2221-E-165-004 and MOST 104-2221-E-030-002.

References

[1] A. Boldyreva, A. Palacio, and B. Warinschi, "Secure proxy signature schemes for delegation of signing rights", *Journal of Cryptology*, vol. 25, no. 1, pp. 57–115, 2012.

[2] F. Cao, and Z. Cao, "A secure identity-based multi-proxy signature scheme", *Computers & Electrical Engineering*, vol. 35, no. 1, pp. 86–95, 2009.

[3] M. Girault, "Self-certified public keys", *Advances in Cryptology-EUROCRYPT'91*, pp. 490-497, 1991.

[4] M. S. Hwang, C. C. Lee, and T. H. Sun, "Data error locations reported by public auditing in cloud storage service", *Automated Software Engineering*, vol. 21, vol. 3, pp. 373-390, 2014.

[5] M. S. Hwang, C. C. Lee, and S. F. Tzeng, "A new proxy signature scheme for a specified group of verifiers", *Information Sciences*, vol. 227, pp. 102-115, 2013.

[6] S. J. Hwang and C. H. Shi, "A simple multi-proxy signature scheme", *In: Proc.* 10^{th} *National Conf. on Information Security*, Hualien, Taiwan, ROC, pp. 134–138, 2000.

[7] J. H. Ji, D. Li, and M. Wang, "New proxy multi-signature, multi-proxy signature and multi-proxy multi-signature schemes from bilinear pairings", *Chinese Journal of Computers-Chinese Edition*, vol. 27, no. 10, pp. 1429–1435, 2004.

[8] A. V. N. Krishna, A. H. Narayana, K. M. Vani, "Window method based cubic spline curve public key cryptography," *International Journal of Electronics and Information Engineering*, vol. 4, no. 2, pp. 94–102, 2016.

[9] C. C. Lee and Y. M. Lai, "Toward a secure single sign-on mechanism for distributed computer networks", *The Computer Journal*, vol. 58, no. 4, pp. 934-943, 2015.

[10] C. C. Lee, T. C. Lin, S. F. Tzeng, and M. S. Hwang, "Generalization of proxy signature based on factorization", *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 3, pp. 1039-1054, 2011.

[11] C. T. Li, C. Y. Weng, C. C. Lee, and C. C. Wang, "A hash based remote user authentication and authenticated key agreement scheme for the integrated EPR information systems", *Journal of Medical Systems*, vol. 39, no. 11, pp. 1-11, 2015.

[12] J. Li and S. Wang, "New efficient proxy blind signature scheme using verifiable self-certified public key", *International Journal of Network Security*, vol. 4, no. 2, pp. 193–200, 2007.

[13] X. Li, K. Chen, and S. Li, "Multi-proxy signature and proxy multi-signature schemes from bilinear pairings", *Parallel and Distributed Computing: Applications and Technologies*, Springer Berlin Heidelberg, pp. 591–595, 2004.

[14] C. Y. Lin, T. C. Wu, and J. J. Hwang, "Multi-proxy signature schemes for partial delegation with cheater identification", *Proceeding of IWAP 2*, 2002.

[15] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures: Delegation of the power to sign messages", *IEICE Trans. Fundamentals*, vol. E79-A, no. 9, pp. 1338–1354, 1996.

[16] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation", *In: Proc. 3^{rd} ACM Conf. on Computer and Communications Security*, pp. 48–57, 1996.

[17] K. Nyberg and R. Rueppel, "Message recovery for signature schemes based on the discrete logarithm problem", *Designs, Codes and Cryptography*, vol. 7, no. 1, pp. 61-81, 1996.

[18] R. Padmavathy and C. Bhagvati, "A new method for computing DLP based on extending smooth numbers to finite field for ephemeral key recovery", *International Journal of Network Security*, vol. 17, no. 3, pp. 251–262, 2015.

[19] C. Pan, S. Li, Q. Zhu, C. Wang, and M. Zhange, "Notes on proxy signcryption and multi-proxy signature schemes", *International Journal of Network Security*, vol. 17, no. 1, pp. 29–33, 2015.

[20] R. A. Sahu, and S. Padhye, "Provable secure identity-based multi-proxy signature scheme", *International Journal of Communication Systems*, vol. 28, no. 3, pp. 497–512, 2015.

[21] K. R. Santosh, C. Narasimham, and P. Shetty, "Cryptanalysis of multi-prime RSA with two decryption exponents," *International Journal of Electronics and Information Engineering*, vol. 4, no. 1, pp. 40–44, 2016.

[22] A. Shamir, "Identity-based cryptosystems and signature schemes", *Advances in cryptology*, pp. 47-53, 1984.

[23] Z. Shao, "Improvement of threshold signature using self-certified public keys", *International Journal of Network Security*, vol. 1, no. 1, pp. 24–31, 2005.

[24] N. Tiwari and S. Padhye, "Provable secure multi-proxy signature scheme without bilinear maps", *International Journal of Network Security*, vol. 17, no. 6, pp. 736–742, 2015.

[25] S. F. Tzeng, C. C. Lee, and M. S. Hwang, "A batch verification for multiple proxy signature", *Parallel Processing Letters*, vol. 21, no. 1, pp. 77–84, 2011.

[26] C. Wu, X. Du, and Z. Jiang, "Linear complexity of a family of pseudorandom discrete logarithm threshold sequences", *International Journal of Network Security*, vol. 18, no. 3, pp. 487–492, 2016.

[27] T. S. Wu, C. L. Hsu, and H. Y. Lin, "Self-certified multi-proxy signature schemes with message recovery", *Journal of Zhejiang University SCIENCE A*, vol. 10, no. 2, pp. 290-300, 2009.

[28] Q. Xie, "Provably Secure Self-certified Multi-proxy Signature with Message Recovery", *Journal of Networks*, vol. 7, no. 10, pp. 1616-1623, 2012.

[29] Q. Xue and Z. Cao, "Improvement of multi-proxy signature scheme", *Proceedings of The Fourth International Conference on Computer and Information Technology*, 2004.

[30] Y. Yu, Y. Sun, and B. Yang, "Multi-proxy signature without random oracles", *Chinese Journal of Electronics*, vol. 17, no. 3, pp. 475–480, 2008.

M. K. Chande received the B. S. and M. S. degrees in mathematics from Pt. Ravishankar Shukla University, Raipur (C.G.), India, in the year 1997 and 1999 respectively. He is currently serving as the capacity of an assistant professor in the Department of Applied Mathematics, Shri Shankaracharya Institute of Professional Management and Technology, Raipur (C.G.), India. He is life member of Cryptology Research Society of India (CRSI). Currently he is doing his Ph. D. degree from School of Studies in Mathematics, Pt. Ravishankar Shukla University, Raipur (C.G.), India. His research interest includes cryptography, analysis, design and applications of digital signatures.

C. C. Lee received the Ph.D. degree in Computer Science from National Chung Hsing University (NCHU), Taiwan, in 2007. He is currently an Associate Professor with the Department of Library and Information Science at Fu Jen Catholic University. Dr. Lee is currently as an editorial board member of International Journal of Network Security, Journal of Computer Science, Cryptography, and International Journal of Internet Technology and Secured Transactions. He also served as a reviewer in many SCI-index journals, other journals, other conferences. His current research interests include

data security, cryptography, network security, mobile communications and computing, wireless communications. Dr. Lee had published over 100+ articles on the above research fields in international journals. He is a member of IEEE, the Chinese Cryptology and Information Security Association (CCISA), the Library Association of The Republic of China, and the ROC Phi Tau Phi Scholastic Honor Society.

C. T. Li received the Ph.D. degree in Computer Science and Engineering from National Chung Hsing University, Taiwan, in 2008. He is currently an Assistant Professor of the Department of Information Management, Tainan University of Technology, Tainan, Taiwan. His research interests include information security, wireless sensor networks, mobile computing, and security protocols for ad hoc networks.