A New Threshold Proxy Signature Scheme with Fast Revocation

Mohammad Beheshti-Atashgah, Mahmoud Gardeshi, and Majid Bayat

Abstract—In a (t, n) threshold proxy signature scheme, the original signer delegate his/her signing power to n proxy signers such that any t or more out of n proxy signers can sign messages on behalf of the original signer, but $t - 1$ or less of the proxy signers cannot generate a proxy signature. In a proxy signature scheme, when original signer wants to revoke the delegation earlier than his/her plan, he/she cannot do anything. Moreover, the revocation of delegated rights is an essential issue of the many proxy signature schemes. In this paper, we try to solve the fast revocation issue of the threshold proxy signature schemes and propose a new threshold proxy signature scheme with fast revocation that solve the fast revocation weakness of threshold proxy signature scheme.

Index Terms—Proxy signature scheme, threshold Proxy signature scheme, fast revocation.

I. INTRODUCTION

The concept of proxy signature schemes was first introduced by Mambo, Usuda and Okamoto in 1996 [1]. Mambo et al. in [2] proposed that according to the delegation, the proxy signature schemes are classified into three types: full delegation, partial delegation and delegation by warrant. In the full delegation, the original signer gives his/her private key to the proxy signer. In the partial delegation, the original signer produces a proxy key from his/her private key and gives it to the proxy signer. The proxy signer uses the proxy key to sign messages on behalf of the original signer. In the delegation by warrant, the original signer gives the proxy signer a warrant, which is introduced by the original signer and includes a specific time period, the identities of the original signer and the proxy signer, and other information. Then the proxy signer uses the warrant and corresponding private key to sign messages. Many proxy signature schemes have been proposed for each of these delegation types, as shown in Refs. [3–6].

However, most existing proxy signature schemes have following two weaknesses [7]. First, the declaration of a valid delegation in the warrant is useless because the proxy signer can still create a proxy signature and claim that his/her signing was issued during the delegation period. Second, even if the signer's key is compromised and the delegated

rights are abused; when the original signer wants to revoke the delegation earlier than his/her plan, he/she can do anything. Thus, the revocation of delegated rights is an essential issue of the proxy signature schemes.

To solve the above problems, some proxy signature schemes have been proposed. Sun showed a time-stamp proxy signature scheme and its improvement in [8]. But Sun's scheme could not solve the second problem. Seo et al. ([7]) proposed a mediated proxy signature scheme to solve the fast revocation problems. Their scheme uses a third special entity, called SEM (SEcurity Mediator) who is an on-line partially trusted server.

In 1997, Kim et al. [9] and Zhang et al. [10] independently proposed the first threshold proxy signature scheme, respectively. Threshold proxy signature schemes are based on proxy signature schemes and threshold cryptography, so these schemes have also the proxy fast revocation problems. For instance, the improvement of Hsu et al.'s scheme [11], Xie's scheme [12] and other threshold proxy signature schemes such as [13] have the same weakness.

Although the fast revocation weakness for proxy signatures was solved by Seo ([7]) and then Zhen-hua et al. [14] solved this weakness for proxy signature schemes in standard model, but the threshold proxy signature schemes still suffer from this weakness. So we want to solve the fast revocation weakness for the threshold proxy signature schemes. In our proposed scheme, for issuing a proxy signature on a message, the proxy signers must obtain a partial proxy token from the SEM and without this token, they can't create the proxy signature. So, if the original signer immediately wants to revoke the delegation, then he/she only instructs the SEM to stop issuing the token for the proxy signers.

The rest of this paper is organized as follows: In section 2, we first define the preliminaries and then will propose our new scheme. In section 3, the analysis of our scheme is presented as follows: the security requirements of our scheme are proposed in subsection 3.1 and then in subsection 3.2, we compare the computational complexities required between the proposed scheme and two threshold proxy signature schemes. And finally, our conclusions are given in section 4.

II. PROPOSED SCHEME

A. Preliminaries

In the proposed Scheme, the system has the three types of participants: the original signer O and the proxy signer group G which contains n proxy signer $\{P_1, P_2, \dots, P_n\}$ and the SEM. The system parameters and the notations are defined as follows.

- O : The original signer.

Manuscript received September 2, 2012; revised October 1, 2012. This work is supported by the Education & Research Institute for ICT (ERICT in Iran) and the authors would like to thank for their support.

Mohammad Beheshti-Atashgah is with the Research Center of Intelligent Signal Processing, Tehran, Iran (e-mail: M.Beheshti.A@gmail.com).

Mahmoud Gardeshi is with Department of Electrical and Computer Engineering, Imam Hossein University, Tehran, Iran (e-mail: mgardeshi@ihu.ac.ir).

Majid Bayat is with Department of mathematics and Computer Science, Tarbiat Moallem University, Tehran, Iran (e-mail: Bayat@tmu.ac.ir).

- G : The original signers group includes n signers.
- SEM: A security mediator, the on-line partially trusted server.
- p, q : Two large prime numbers such that $q|p-1$.
- g : A generator of Z_p^* from order q .
- The original signer has secret key $x_o \in Z_q^*$ and corresponding public key $y_o = g^{x_o} \bmod p$. Each proxy signer P_i in proxy signers group G has secret key $x_i \in Z_q^*$ and public key $y_i = g^{x_i} \bmod p$. Similarly, $x_s \in Z_q^*$ is the secret key and $y_s = g^{x_s} \bmod p$ is the corresponding public key of the SEM.
- $h(\cdot)$: A collision resistant one-way hash function.
- $APSID$: denotes the identities of the actual proxy signers.
- ω : A warrant which specifies original signer's ID, proxy signers' identities, SEM's ID, period of delegation and etc.

All of the public keys are certificated by a Certificate Authority or CA. of course, when any one of signers request CA to change his/her public key (in the case of public-key substitute attack), CA uses Zero-Knowledge Proof Test to insure the correspondence between public and private keys. In this way, the public-key substitute attack will be prevented.

The proposed scheme comprises four phases: secret share generation phase, proxy share generation phase, Proxy signature issuing phase and proxy signature verification phase.

1) Secret share generation phase

For secret share generating, all P_i in proxy signers group $G = \{P_1, P_2, \dots, P_n\}$ cooperate to generate the secret share by executing Pedersen's VSS (Verifiable Secret Sharing) scheme [15].

Each P_i in the proxy signers group G randomly chooses a $(t-1)$ -degree polynomial $f_i(x)$ over $Z_q[x]$.

$$f_i(x) = x_i + a_{i0}A_0 + a_{i1}x + a_{i2}x^2 + \dots + a_{i,t-1}x^{t-1} \bmod q, \quad (1)$$

where $A_0 = g^{\sum_{i=1}^n a_{i0}} \bmod p$

P_i publishes

$$A_{il} = g^{a_{il}} \bmod p, l = 1, 2, \dots, t-1 \quad (2)$$

Each P_j computes and sends $f_j(i)$ to P_i via a secure channel that $1 \leq i, j \leq n, i \neq j$.

P_i can check the validity of the shared value $f_j(i)$ by the equality $g^{f_j(i)} = y_j^{y_j} A_{j0}^{A_0} \prod_{l=1}^{t-1} A_{jl}^{A_{il}} \bmod p$, if all the shared values $f_j(i)$ from the other proxy signers are verified then P_i computes the public value A_l ($l = 1, 2, \dots, t-1$) and his/her secret share s_i as follows:

$$A_l = \prod_{i=1}^n A_{il} \bmod p, \quad (3)$$

$$s_i = \sum_{j=1}^n f_j(i) = f(i) = \sum_{i=1}^n x_i + a_0 A_0 + a_{i1}i + a_{i2}i^2 + \dots + a_{i,t-1}i^{t-1} \bmod q \quad (4)$$

2) Proxy share generation phase

The original signer O chooses $k_{P_i}, k_S \in_R Z_q$ at random

and then computes $K_O = g^{k_O} \bmod p$ and $K_P = g^{\sum_{i=1}^n k_{P_i}} \bmod p$, where $k_O = \sum_{i=1}^n k_{P_i} + k_S$.

The original signer also computes the following values

$$\sigma_{P_i} = k_{P_i} + x_o h(\omega, K_O) \bmod q \quad (5)$$

$$\sigma_S = k_S + x_o h(\omega, K_O) \bmod q \quad (6)$$

Then original signer O sends $(\omega, K_O, \sigma_{P_i})$ to each proxy signer P_i and sends (ω, K_O, σ_S) to SEM. To confirm the validity of $(\omega, K_O, \sigma_{P_i})$, each P_i computes $R_{P_i} = g^{\sigma_{P_i}} \bmod p$ and sends (ω, R_{P_i}) to the SEM. After P_i receives $R_S = g^{\sigma_S} \bmod p$ from the SEM, P_i verifies whether or not the following equation holds: $R_{P_i} \cdot R_S = K_O \cdot y_o^{2h(\omega, K_O)} \bmod p$. Similarly, SEM verifies this equation by using R_{P_i} .

If the verification is successful, each P_i and SEM compute their shares $\sigma_{P_i}^*$ and σ_{P_S} , respectively.

$$\begin{aligned} P_i: \sigma_{P_i}^* &= \sigma_{P_i} + x_i h(\omega, K_O) \bmod q \\ \text{SEM: } \sigma_{P_S} &= \sigma_S + x_S h(\omega, K_O) \bmod q \end{aligned} \quad (7)$$

In the end of phase (1.) and (2.), each P_i computes his/her proxy share as follows.

$$\sigma_{P_i}' = \sigma_{P_i}^* + s_i h(\omega, K_O) \bmod q \quad (8)$$

3) Proxy signature issuing phase

Without loss of generality, we suppose that the proxy group $\{P_1, P_2, \dots, P_t\}$ is the actual proxy group with an identity collection $APSID$ which will sign message m .

Each proxy signer P_i randomly chooses $l_{P_i} \in_R Z_q$ and computes $L_{l_{P_i}} = g^{l_{P_i}} \bmod p$ and then sends $(\omega, m, K_O, R_{P_i}, L_{P_i})$ to SEM. First, the SEM check the correctness of $(\omega, m, K_O, R_{P_i}, L_{P_i})$ by comparing it with the previous received $(\omega, K_O, \sigma_{P_i})$ in phase 2. In continue, the SEM must ascertain the following conditions, before he generates partial proxy signature on the m :

- The period of proxy delegation specified in ω should be valid.
- The K_O should not be in the public revocation list (the revocation list maintained by the SEM). If the K_O is in the public revocation list, it means that the delegation has been revoked.

If the validation is finished correctly, then SEM will issue the token (partial proxy signature) as follows:

SEM chooses $l_S \in_R Z_q$ at random and computes

$$L_S = g^{l_S} \bmod p, L_O = g^{l_S} \cdot g^{\sum_{i=1}^n k_{P_i}} \bmod p \quad (9)$$

SEM generates a partial proxy signature on the m as follows:

$$S_{SEM} = l_S + \sigma_{P_S} h(m, L_O) \bmod p \quad (10)$$

Then SEM sends (L_O, S_{SEM}, L_S) to the proxy signers and verifier. The proxy signers validate the received threesome through the following equation

$$g^{S_{SEM}} = L_S \cdot \left(R_S \cdot y_S^{h(\omega, K_O)} \right)^{h(m, L_O)} \mod p \quad (11)$$

Moreover, each P_i randomly chooses $k_i \in_R \mathbb{Z}_q^*$. Then P_i computes and publishes $r_i = g^{k_i} \mod p$.

Finally, each P_i generates his/her individual signature as follows. For simplicity, we will use the notation $H = h(R, A_0, APSID, m, K_O)$ in continue.

$$R = \prod_{i=1}^t r_i \mod p \quad (12)$$

$$\gamma_i = (k_i R + S_{SEM}) + \sigma_{P_i}' H \mod q \quad (13)$$

Then, all of the proxy signers send their individual signatures γ_i to a designated clerk. The designated clerk can be any proxy signers in $APSID$. The clerk validates proxy sub-signature γ_i by checking the equation

$$g^{\gamma_i} = r_i^R \left[L_S \cdot \left(R_S \cdot y_S^{h(\omega, K_O)} \right)^{h(m, L_O)} \right] \times \left[\left(g^{k_{P_i}} (y_O y_i)^{h(\omega, K_O)} \right)^H \cdot \left(y_i A_{i0}^{A_0} \prod_{i=1}^{t-1} A_i^{j_i} \right)^{h(\omega, K_O)} \right] \mod p \quad (14)$$

If all the individual proxy signatures are valid, the clerk computes the final signature as follows

$$T = \sum_{i=1}^t \gamma_i \mod q \quad (15)$$

Then, $(\omega, T, K_O, R, A_0, APSID)$ is the proxy signature on message m .

4) Proxy signature verification phase

The verifier can check the validity of the proxy signature through the equation:

$$g^T = R^R \left[L_S \cdot \left(R_S \cdot y_S^{h(\omega, K_O)} \right)^{h(m, L_O)} \right] \times \left[K_P A_0^{A_0 h(\omega, K_O)} y_0^{h(\omega, K_O)} \cdot \left(\prod_{i=1}^t y_i \right)^{2h(\omega, K_O)} \right]^H \mod p. \quad (16)$$

III. ANALYSIS OF THE PROPOSED SCHEME

A. Security Requirements

- 1) **Verifiability:** In our scheme, the proxy signature is consist of $(\omega, T, K_O, R, A_0, APSID)$. So, from ω and $APSID$, any verifier can identify the identities of the original signer, the proxy signers and the SEM. Since the original signer's public key is needed to verify the signature, the verifier can be convinced of the original signer's agreement on the signed message.

- 2) **Strong Unforgeability:** Suppose that the dishonest original signer try to forge the proxy signers' proxy signature. But he/she don't know the value of share s_i and from other hand; the original signer cannot obtain the proxy signers' private keys (because it is equivalent to solving the DLP). So the original signer cannot generates or the obtain proxy signers' shares σ_{P_i} and thus he/she will not be able to forge a proxy signature by name of $APSID$. Therefore, only the proxy signers can create a valid proxy signature. **Resistance against known forgery attacks:** collusion attack and Public-key substitute attack. In collusion attack, a malicious proxy group of t or more malicious proxy signers collude and can obtain any proxy/secret share by using the Lagrange interpolating polynomial. Finally, they can easily forge a valid proxy signature on their arbitrarily message by name of $APSID$. With use of the SEM and a random value k_i in proxy signature generation phase, even if the malicious proxy signers can obtain the all of shares s_i , they can't forge a valid proxy signature by name of $APSID$. Therefore our scheme is resistance against collusion attack. In Public-key substitute attack, any malicious proxy signer can obtain a forged public key which satisfying the proxy signature verification equation. Then he/she requests CA to change his/her forged public key with real public key. The CA uses Zero-knowledge Proof Test to insure the correspondence between public and private keys and will not certificate the forged public key. In proposed scheme, we use the Zero-knowledge Proof CA and therefore our scheme is resistance against Public-key substitute attack.
- 3) **Strong Identifiability:** In our scheme, identity information of the proxy signers is included explicitly in a valid proxy signature and $(\omega, APSID)$ as a form of public key y_i . So, anyone can determine the identity of the proxy signers from the proxy signature created by them, and confirm the identity of the proxy signers from $\omega, APSID$.
- 4) **Strong Undeniability:** Due to the difficulty of DLP, anyone cannot know the proxy signers' private keys and only each proxy signer can know his/her private key. Therefore, when the proxy signature is created by the proxy signers, they cannot repudiate it.
- 5) **Prevention of misuse:** only the proxy signers group can generate a valid proxy signature because their proxy validation is verified by SEM and only they have the SEM's token. Moreover, the original signer or the malicious attacker's misuse is also prevented, because they cannot compute a valid proxy key pairs.

B. Performance Evaluation

In the Table I, we compare the computational complexities between our proposed scheme and the Hwang et al. scheme ([15]) and the Xie scheme. It can be seen that the computational costs of our scheme in the secret share generation and proxy signature verification phases, are a few more than the Hwang et al. scheme but our scheme still has less computational amount than the Xie scheme. Our proposed scheme has much computational than the other two

schemes in the proxy share generation phase. In the proxy generation phase, our scheme has less computational from both of the Hwang et al. and the Xie schemes.

With attendance to the security of our scheme, the computational complexities of the proposed scheme are acceptable. Note that both of the Hwang scheme and the Xie scheme are vulnerable against the collusion attacks and they also have not the fast revocation capability. We use the

following notation in the Table I:

- T_e : The time for executing a modular exponentiation computation.
- T_m : The time for executing a modular multiplication computation.
- T_h : The time for executing a one-way hash function h .

TABLE I: THE COMPARISON OF COMPUTATIONAL COMPLEXITIES.

	Hwang et.al's scheme [16]	Xie's scheme [12]	Proposed scheme
Secret Share Gen	$(n^2t)T_e + (3n^2t - 2nt - 3n^2 + 3n - t)T_m$	$(n^2t + 2n)T_e + (3n^2t - 2nt - 3n^2 + 6n - t + 1)T_m$	$(n^2t + n)T_e + (3n^2t - 2nt - 3n^2 + 4n - t + 1)T_m$
Proxy Share Gen	O: $tT_e + (nt - n + 1)T_m + T_h$ each P_i: $(t + 1)T_e + (2t - 1)T_m + T_h$	O: $tT_e + (nt - n + 2)T_m + T_h$ each P_i: $(t + 1)T_e + (2t - 1)T_m + T_h$	O: $(n + 1)T_m + T_h$ each P_i: $(2n)T_e + (3n + 2)T_m$
Proxy Signature Gen	Individual signature: $(t^2)T_e + (4t^2 - 7t + 5)T_m + T_h$ Proxy signature: $(3t^2 - t - 1)T_e + (6t^2 - 6t)T_m + (t^2 - t) + T_h$	Individual signature: $(t - 1)T_e + (t^2 - t + 4)T_m + T_h$ Proxy signature: $(nt + 5t^2 - 4t + 4)T_e + (nt + 4t^2 - 4t + 8)T_m + 2T_h$	Individual signature: $(t + 4)T_e + (t + 6)T_m + 2T_h$ Proxy signature: $(t + 8)T_e + (2t + 6)T_m + 3T_h$
Proxy Verification	$4T_e + (t + 4)T_m + 2T_h$	$(n + t + 7)T_e + (n + t + 5)T_m + 2T_h$	$7T_e + (2t + 3)T_m + 3T_h$
Success Attacks	Collusion attack [17]	Collusion attack [18]	-
Fast Revocation	No	No	Yes

IV. CONCLUSIONS

The fast revocation of delegated rights is an essential issue of the proxy signature schemes and their improvements.

So far, many ways have been proposed to solve this weakness. Seo et al. in ([7]) showed a mediated proxy signature scheme which can perform the fast revocation. They used a special entity called a SEM in their scheme. In this paper, we used the Seo et al.'s technique and proposed the first threshold proxy signature scheme with fast revocation.

Our proposed threshold proxy signature scheme has acceptable efficiency and provides all security requirements for threshold proxy signature.

REFERENCES

- [1] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signature: delegation of the power to sign messages," *IEICE Transactions on Fundamentals*, 1996, vol. 9, pp. 1338-1353.
- [2] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signature for delegating signing operation," in *Proceedings of the 3rd ACM Conference on Computer and Communications Security*, 1996, New Delhi, India. New York, NY, USA: ACM, 1996, pp. 48-56.
- [3] A. Boldyreva, A. Palacio, and B. Warinschi, "Secure proxy signature scheme for delegation of signing rights,"

- [4] X. Y. Huang, W. Susilo, Y. Mu, et al., "Proxy signature without random oracles," in *Proceedings of the 2nd International Conference on Mobile Ad-hoc and Sensor Networks*, Dec 13-15, 2006, Berlin, German: Springer-Verlag, 2006, pp. 473-484.
- [5] Y. Yu, Y. Sun, B. Yang, et al., "Multi-proxy signature without random oracles". *Chinese Journal of Electronics*, 2008, vol. 17, no. 3, pp. 475-480.
- [6] Z. H. Liu, Y. P. Hu, and H. Ma, "Secure proxy multi-signature scheme in the standard model," in *Proceedings of the 2nd International Conference on Provable Security (ProvSec'08)*, Oct 30 Nov 1, Shanghai, China. LNCS 5324. Berlin, German: Springer-Verlag, 2008, pp. 127-140.
- [7] S. H. Seo, K. A. Shim, and S. H. Lee, "A mediated proxy signature scheme with fast revocation for electronic transactions," in *Proceedings of the 2nd International Conference on Trust, Privacy and Security in Digital Business*, Aug 22-26, 2005, Copenhagen, Denmark. LNCS 3592. Berlin, German: Springer-Verlag, 2005, pp. 216-225.
- [8] M. H. Sun, "Design of time-stamped proxy signatures with traceable receivers," *IEE Proceedings: Computers and Digital Techniques*, 2000, vol. 147, no. 6, pp. 462-466.
- [9] S. J. Kim, S. J. Park, and D. H. Won, "Proxy Signatures, revisited," *ICICS'97*, LNCS 1334, Springer-Verlag, pp. 223-232, 1997.
- [10] K. Zhang, "Threshold proxy signature schemes," *Information Security Workshop*, Japan, 1997, pp. 191-197.
- [11] Z. Tan, "Improvement on C.-L. Hsu et al's threshold proxy signature scheme with known signers," *international Conference on Convergence Information Technology*, 2007, pp. 1463-1467.
- [12] Q. Xie, "Improvement of Tzeng et al.'s nonrepudiable threshold proxy signature scheme with known signers," *Applied Mathematics and Computation*, vol. 168, 2005, pp. 776-782.

- [13] J. Hu and J. Zhang, "Cryptanalysis & improvement of a threshold proxy signature scheme," *Computer Standards and Interfaces*, 2009, pp.169-173.
- [14] L. Z. Hua, H. Y. Pu, Z. X. Song, and M. Hua, "secure proxy signature scheme with fast revocation in the standard model," *Journal of China Universities of posts and Telecommunications*, August 2009, vol. 16, no. 4, pp. 116-124.
- [15] T. P. Pedersen, "Distributed proves with Applications to Undeniable Signatures," in *Proc. Advance in Cryptology*, LNCS 547, Springer-Verlag, 1991, pp. 221-242.
- [16] M. S. Hwang, I. C. lin, and E. J. Lu, "A secure nonrepudiable Threshold proxy signature scheme with known signer," *Informatica*, vol. 11, no. 2, 2000, pp.137-144.
- [17] C. L. Hsu and T. S. Wu, "Efficient nonrepudiable Threshold proxy signature scheme with known signers against the collusion attack," *Applied Mathematics and Computation*, Science Direct, pp. 305-319.
- [18] Z. Liu and Z. Tan, "A New Type of Collusion Attack against Threshold proxy signature schemes," *International Conference on Convergence Information Technology*, 2007, pp. 279-283.

Schemes, Identity-Based Cryptography, Lattice-Based Cryptography and Network Security.

Mohammad Beheshti-Atashgah was born in Tehran, Iran on December 1984. He received his B.Sc. degree in Electrical Engineering in 2008 and his M.Sc. in Communication Engineering from Imam Hossein University, Tehran, Iran in 2011. Until now, he has published more than 14 papers in the National and International journals, Conferences and workshops. His research interests include: Cryptographic Protocols, Provable Security of Digital Signature Schemes, Identity-Based Cryptography, Lattice-Based Cryptography and Network Security.

Mahmoud Gardeshi received his B.Sc. degree in applied mathematics from Shiraz University in 1989 and M.Sc. degree in applied mathematics from Tabriz University in 1991. He also received his M.Phil degree from Amir Kabir University, Iran in 1999. Now, He is a researcher at the Imam Hossein University (I.H.U), Tehran, Iran. His research interest includes: Public key Cryptography, Lattice-Based Cryptography, Digital Signatures and Cryptographic Protocols.

Majid Bayat is a Ph.D. candidate in the Department of Mathematics and Computer Sciences at Kharazmi University (Tarbiat Moallem University) in Tehran, Iran. He is presently a Research Assistant of Tarbiat Moallem University and Information Systems and Security Lab (ISSL) of Sharif University in Tehran, Iran. His research interests include: Public Key Cryptography, Key Agreement Protocols and Provable Security.