Available online at www.sciencedirect.com
SCIENCE DIRECT®

ELSEVIER

Future Generation Computer Systems 20 (2004) 887–893

FGCS
FUTURE GENERATION COMPUTER SYSTEMS
www.elsevier.com/locate/future

A nonrepudiable threshold multi-proxy multi-signature scheme with shared verification

Shiang-Feng Tzeng [a,*], Cheng-Ying Yang [b], Min-Shiang Hwang [c]

[a] *Department of Computer Science and Information Engineering, National Central University,*
No. 300, Jung-da Rd., Jung-li City, 320 Taoyuan, Taiwan, ROC
[b] *Graduate Institute of Networking and Communication Engineering, Chaoyang University of Technology,*
168 Gifeng E. Rd., Wufeng, 413 Taichung, Taiwan, ROC
[c] *Department of Management Information System, National Chung Hsing University,*
250 Kuo Kuang Road, 402 Taichung, Taiwan, ROC

Received 26 February 2003; received in revised form 5 January 2004; accepted 19 January 2004

Available online 18 March 2004

Abstract

In this paper, we shall propose a threshold multi-proxy multi-signature scheme with shared verification. In the scheme allows the group of original signers to delegate the signing capability to the designated group of proxy signers. Furthermore, a subset of verifiers in the designated verifier group can authenticate the proxy signature. A threshold multi-proxy multi-signature scheme with the nonrepudiation property is a scheme where the proxy group cannot deny signing for the message and the verifier group can identify the proxy group for a proxy signature.

Keywords: Digital signature; Proxy signature; Threshold proxy signature; Proxy multi-signature scheme; Multi-proxy multi-signature scheme; Threshold multi-proxy multi-signature scheme

1. Introduction

A new type of digital signature, proxy signature, was first proposed in 1996 [6,7]. The proxy signature allows a designated person, called a proxy signer, to sign on behalf of an original signer. So far, many proxy signature schemes were discussed [1,3–5,8,9,11–14].

They distinguish into several kinds of proxy signature schemes. The proxy multi-signature schemes were proposed in [13]. In the proxy multi-signature schemes, a proxy signer is allowed to generate a proxy multi-signature on behalf of two or more original signers. The threshold proxy signature schemes were also widely proposed [1,3–5,8,11,12,14]. In the (t, n) threshold proxy signature scheme, any t or more of the proxy singers can cooperatively sign messages on behalf of an original signer. Recently, a kind of proxy signature scheme, multi-proxy multi-signature scheme, was proposed [9]. The scheme allows the group of original signers to delegate the signing capability to the designated group of proxy signers.

In the schemes mentioned above, outsiders are allowed to play the role of verifiers. However, in most existing signature schemes, there can be only one legal signer and one legal verifier. To bridge this gap, Hsu and Wu [2] used the concept of (t, n) threshold signature to extend the capability of verification which is

[*] Corresponding author.
E-mail address: sftzeng@csie.ncu.edu.tw (S.-F. Tzeng).

0167-739X/$ – see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2004.01.002

addressed to one signer and a group of verifiers. Wang et al. [10] also extended the existing schemes to the group-oriented (t, n) threshold signature with a (k, l) shared verification scheme among groups. Similarly, in all existing proxy signature schemes, there can be only one legal verifier. In practical applications, there is usually a need to have some specified verifiers verify the proxy signature. Assume that several directors represent a directorate to delegate their signing capability to a group of managers. Then, several of these managers represent the company to sign a contract with another company through computer network. By using the proposed scheme, the validity of the business between the two companies can be guaranteed.

According to the above statement, the group of verifiers to verify the message has to be specified, and the message should also be able to be authenticated by the specified group of verifiers. In this paper, we shall propose a new scheme in which the original signer group, proxy signer group and verifier group are specified under the predefined proxy warrant.

In Section 2, we shall use the concept of proxy signature and predefine the proxy warrant to propose a threshold multi-proxy multi-signature scheme for multi-groups. The security of the proposed scheme will be discussed in Section 3. Finally, the conclusion will be given in Section 4.

2. The proposed scheme

In this section, a threshold proxy signature scheme among groups will be proposed. According to the proxy warrant, a subset of original signers allows a designated proxy group to sign on behalf of the original group. A message m has to be signed by a subset of proxy signers who can represent the proxy group. Then, the proxy signature is sent to the verifier group. A subset of verifiers in the verifier group can also represent the group to authenticate the proxy signature.

In other words, some threshold values will be given to indicate the number of persons to represent a group to authorize the signing capability or to sign a message or to verify the proxy signature. The proposed scheme requires a share distribution center (SDC) which is responsible for setting some parameters and initializing the scheme. Moreover, the SDC is trusty and does not collaborate with the conspiring party. Therefore, the proposed scheme involves four parties: the share distribution center, the original group, the proxy group and the verifier group. The procedure of the proposed scheme contains four phases: the secret share generation phase, the proxy share generation phase, the proxy signature generation phase and the proxy signature verification phase. The reference configurations and flow are illustrated in Fig. 1. For the initial step, the SDC first selects and publishes the following parameters:

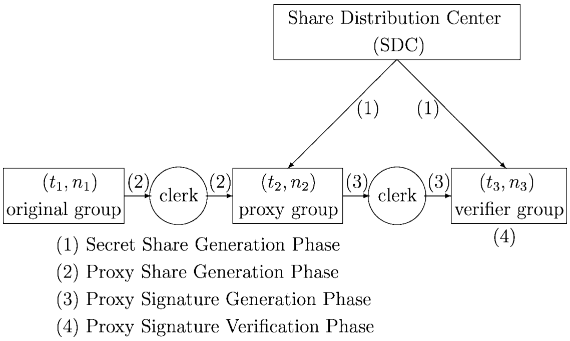
Share Distribution Center (SDC)
(1)
(1)
(t_1, n_1) original group
(2)
clerk
(2)
(t_2, n_2) proxy group
(3)
clerk
(3)
(t_3, n_3) verifier group
(4)
(1) Secret Share Generation Phase
(2) Proxy Share Generation Phase
(3) Proxy Signature Generation Phase
(4) Proxy Signature Verification Phase

Fig. 1. Threshold multi-proxy multi-signature scheme.

- p: a large prime;
- q: a large prime factor of $p - 1$;
- g: a generator in Galois field $\mathrm{GF}(p)$ of order q;
- $h(\cdot)$: a one-way hash function;
- m_{W}: a warrant which records the identities of the original signers in the original group, the proxy signers in the proxy group and the verifiers in the verifier group, the parameters (t_1, n_1), (t_2, n_2) and (t_3, n_3) and the valid delegation time, etc.;
- AOSID (actual original signers' ID): the identities of the actual original signers;
- APSID (actual proxy signers' ID): the identities of the actual proxy signers.

Each original signer U_{O_i} owns a secret key $x_{\mathrm{O}_i} \in Z_q^*$ and a public key $y_{\mathrm{O}_i} = g^{x_{\mathrm{O}_i}} \bmod p$ which is certified by a certificate authority (CA). Similarly, each proxy signer U_{P_i} or verifier U_{V_i} also owns a secret key $x_{\mathrm{P}_i} \in Z_q^*$ or $x_{\mathrm{V}_i} \in Z_q^*$ and a public key $y_{\mathrm{P}_i} = g^{x_{\mathrm{P}_i}} \bmod p$ or $y_{\mathrm{V}_i} = g^{x_{\mathrm{V}_i}} \bmod p$ which is also certified by the same or different the CA.

Then, the SDC accepts the registration of a group of proxy signers or a group of verifiers. For notations, the parameters will be expressed by a subscript G_O, G_S or G_V which is used by the original signer group, the proxy signer group or the verifier group. Let $G_\mathrm{O} = \{U_{\mathrm{O}_1}, U_{\mathrm{O}_2}, \dots, U_{\mathrm{O}_{n_1}}\}$, $G_\mathrm{P} = \{U_{\mathrm{P}_1}, U_{\mathrm{P}_2}, \dots, U_{\mathrm{P}_{n_2}}\}$ and $G_\mathrm{V} = \{U_{\mathrm{V}_1}, U_{\mathrm{V}_2}, \dots, U_{\mathrm{P}_{n_3}}\}$ be groups of n_1 original signers, n_2 proxy signers and n_3 verifiers, respectively. According to the definition of a threshold multi-proxy multi-signature scheme, any t_1 out of n_1 original signers ($1 \le t_1 \le n_1$) can represent the original signer group to delegate signing capability. Any t_2 out of n_2 proxy signers ($1 \le t_2 \le n_2$) can represent the proxy group to sign a message on behalf of the original group. Similarly, any t_3 out of n_3 verifiers ($1 \le t_3 \le n_3$) can represent the verifier group to verify the proxy signature.

2.1. Secret share generation phase

In this phase, three parties are involved, including the share distribution center, the proxy group and the verifier group. The SDC first chooses the proxy group secret key $X_\mathrm{P} \in Z_q^*$ and computes the proxy group public key $Y_\mathrm{P} = g^{X_\mathrm{P}} \bmod p$ which is certified by the CA. Similarly, the SDC also selects a verifier group secret key $X_\mathrm{V} \in Z_q^*$ and computes the verifier group public key $Y_\mathrm{V} = g^{X_\mathrm{V}} \bmod p$ which is also certified by the CA. For the proxy group G_P and the verifier group G_V, the SDC randomly generates two secret polynomial functions

$$f_\mathrm{P}(x) = X_\mathrm{P} + P_1 x + \cdots + P_{t_2-1} x^{t_2-1} \bmod q$$

and

$$f_\mathrm{V}(x) = X_\mathrm{V} + V_1 x + \cdots + V_{t_3-1} x^{t_3-1} \bmod q,$$

and the degrees are $t_2 - 1$ and $t_3 - 1$, respectively.

Furthermore, the SDC obtains a public key y_{P_i} for each proxy signer in the proxy signer group and computes her/his secret shadow $f_\mathrm{P}(y_{\mathrm{P}_i})$, for $i = 1, 2, \dots, n_2$. The corresponding public key is also determined by $y_{f\mathrm{P}_i} = g^{f_\mathrm{P}(y_{\mathrm{P}_i})} \bmod p$, for $i = 1, 2, \dots, n_2$. Similarly, each verifier's secret shadow is $f_\mathrm{V}(y_{\mathrm{V}_i})$, and the corresponding public key is determined by $y_{f\mathrm{V}_i} = g^{f_\mathrm{V}(y_{\mathrm{V}_i})} \bmod p$, for $i = 1, 2, \dots, n_3$. In summary, the proposed scheme parameters are listed as follows:

- Public information of center: p, q, g, $h(\cdot)$.
- Secret information of center: $f_\mathrm{P}(x)$, $f_\mathrm{V}(x)$.
- Public information of original signer: y_{O_i}.
- Secret information of original signer: x_{O_i}.
- Public information of proxy signer: y_{P_i}, $y_{f\mathrm{P}_i}$.
- Secret information of proxy signer: x_{P_i}, $f_\mathrm{P}(y_{\mathrm{P}_i})$.
- Public information of verifier: y_{V_i}, $y_{f\mathrm{V}_i}$.
- Secret information of verifier: x_{V_i}, $f_\mathrm{V}(y_{\mathrm{V}_i})$.
- Public information of group: Y_P, Y_V.
- Secret information of group: X_P, X_V.

2.2. Proxy share generation phase

The proposed scheme allows any t_1 original signers to represent the group to delegate the signing capability to the proxy group to sign on behalf of the original group. So, two parties, namely the original group and proxy group, are involved in this phase. Without loss of generality, assume that the t_1 original signers to delegate the signing capability are indicated as $D_\mathrm{O} = \{U_{\mathrm{O}_1}, U_{\mathrm{O}_2}, \dots, U_{\mathrm{O}_{t_1}}\}$. Let D_O be the actual original signers. D_O as a group executes the following steps to delegate the signing capability to G_P:

1. Choose a random number $a_i \in Z_q^*$ and broadcast k_i:

   $$k_i = g^{a_i} \bmod p.$$

2. For each received k_j ($j = 1, 2, \dots, t_1$; $j \neq i$), each $U_{\mathrm{O}_i} \in D_\mathrm{O}$ computes

   $$K = \prod_{i=1}^{t_1} k_i \bmod p,$$
   $$\sigma_{\mathrm{O}_i} = a_i K + x_{\mathrm{O}_i} h(K \| m_\mathrm{w} \| \mathrm{AOSID}) \bmod q.$$

3. Send σ_{O_i} to the designated clerk via a public channel.
4. After receiving σ_{O_i}, the designated clerk first computes

   $$\sigma_\mathrm{O} = \sum_{i=1}^{t_1} \sigma_{\mathrm{O}_i} \bmod q.$$

   Then, the designated clerk checks whether the following equation holds:

   $$g^{\sigma_\mathrm{O}} \overset{?}{=} K^K \prod_{i=1}^{t_1} y_{\mathrm{O}_i}^{h(K \| m_\mathrm{w} \| \mathrm{AOSID})} \bmod p.$$

If it does, the designated clerk computes

$$\sigma = t_2^{-1}\sigma_{\mathrm{O}} \bmod q.$$

If it does not, the designated clerk checks whether the following equation holds:

$$g^{\sigma_{\mathrm{O}_i}} \overset{?}{=} k_i^K y_{\mathrm{O}_i}^{h(K\|m_{\mathrm{w}}\|\mathrm{AOSID})} \bmod p.$$

The designated clerk can detect a incorrect signature and then request the actual original signer to deliver a valid one.

5. Broadcast $(\sigma, m_{\mathrm{w}}, K, \mathrm{AOSID})$ to G_{P}.

After receiving $(\sigma, m_{\mathrm{w}}, K, \mathrm{AOSID})$, each $P_i \in G_{\mathrm{P}}$ checks whether or not the following equation holds:

$$g^{\sigma} \overset{?}{=} \left(K^K \prod_{i=1}^{t_1} y_{\mathrm{O}_i}^{h(K\|m_{\mathrm{w}}\|\mathrm{AOSID})}\right)^{t_2^{-1}} \bmod p.$$

If it does, each P_i uses σ as her/his proxy share.

2.3. Proxy signature generation phase

This phase has to do with the proxy group and verifier group as well. Without loss of generality, the proposed scheme allows any t_2 proxy signers to represent the proxy group to sign a message m. Let $D_{\mathrm{P}} = \{U_{\mathrm{P}_1}, U_{\mathrm{P}_2}, \ldots, U_{\mathrm{P}_{t_2}}\}$ be the actual proxy signers. D_{P} as a group executes the following steps to generate the proxy signature:

1. Each $U_{\mathrm{P}_i} \in D_{\mathrm{P}}$ chooses a random number $b_i \in Z_q^*$ and broadcasts r_{P_i}:

$$r_{\mathrm{P}_i} = g^{b_i} \bmod p.$$

2. Each $U_{\mathrm{P}_i} \in D_{\mathrm{P}}$ uses a secret shadow $f_{\mathrm{P}}(y_{\mathrm{P}_i})$ and a group public key Y_{V} of the verifiers to compute and broadcast r'_{P_i}:

$$r'_{\mathrm{P}_i} = (Y_{\mathrm{V}})^{f_{\mathrm{P}}(y_{\mathrm{P}_i})\prod_{j=1,j\neq i}^{t_2}(0-y_{\mathrm{P}_j})/(y_{\mathrm{P}_i}-y_{\mathrm{P}_j})} \bmod p.$$

3. For each received r_{P_j} and r'_{P_j} ($j = 1, 2, \ldots, t_2$; $j \neq i$), each $U_{\mathrm{P}_i} \in D_{\mathrm{P}}$ computes

$$R = \prod_{i=1}^{t_2} r_{\mathrm{P}_i} \bmod p, \qquad R' = \prod_{i=1}^{t_2} r'_{\mathrm{P}_i} \bmod p,$$

$$s_i = R' f_{\mathrm{P}}(y_{\mathrm{P}_i}) \prod_{j=1,j\neq i}^{t_2} \frac{0 - y_{\mathrm{P}_j}}{y_{\mathrm{P}_i} - y_{\mathrm{P}_j}} + b_i R + (\sigma + x_{\mathrm{P}_i})h(R\|\mathrm{APSID}\|m) \bmod q, \tag{1}$$

here, s_i is the individual proxy signature which is sent to the designated clerk.

4. After receiving s_i, the designated clerk first computes

$$S = \sum_{i=1}^{t_2} s_i \bmod q.$$

Then, the designated clerk checks whether the following equation holds:

$$g^S \overset{?}{=} Y_{\mathrm{P}}^{R'} R^R \left(K^K \prod_{i=1}^{t_1} y_{\mathrm{O}_i}^{h(K\|m_{\mathrm{w}}\|\mathrm{AOSID})} \prod_{j=1}^{t_2} y_{\mathrm{P}_j}\right)^{h(R\|\mathrm{APSID}\|m)} \bmod p.$$

If it does, all (r_i, s_i) are valid individual proxy signatures of message m. If it does not, the designated clerk checks whether the following equation holds:

$$g^{s_i} \overset{?}{=} y_{f\mathrm{P}_i}^{R'\prod_{j=1,j\neq i}^{t_2}(0-y_{\mathrm{P}_j})/(y_{\mathrm{P}_i}-y_{\mathrm{P}_j})} r_{\mathrm{P}_i}^R \left(\left(K^K \prod_{i=1}^{t_1} y_{\mathrm{O}_i}^{h(K\|m_{\mathrm{w}}\|\mathrm{AOSID})}\right)^{t_2^{-1}} y_{\mathrm{P}_i}\right)^{h(R\|\mathrm{APSID}\|m)} \bmod p.$$

The designated clerk can find a wrong signature and then ask the actual proxy signer to transfer a valid one.

The proxy signature of m is $(m_{\mathrm{w}}, K, \mathrm{AOSID}, R, S, \mathrm{APSID})$.

2.4. Proxy signature verification phase

Any t_3 out of n_3 verifiers in the group G_V can cooperate to verify the validity of the proxy signature. Let $D_\mathrm{V} = \{U_{\mathrm{V}_1}, U_{\mathrm{V}_2}, \ldots, U_{\mathrm{V}_{t_3}}\}$ be the actual verifiers. The steps of this phase are described as follows:

1. According to m_w, AOSID and APSID, each verifier gets the public keys of the original signers and proxy signers from the CA and knows who the actual original signers and the actual proxy signers are.
2. Each $U_{\mathrm{V}_i} \in D_\mathrm{V}$ uses her/his secret shadow $f_\mathrm{V}(y_{\mathrm{V}_i})$ and the group public key Y_P of the proxy signers to compute and broadcast r'_{V_i}:
$$r'_{\mathrm{V}_i} = (Y_\mathrm{P})^{f_\mathrm{V}(y_{\mathrm{V}_i})\prod_{j=1, j\neq i}^{t_3}(0-y_{\mathrm{V}_j})/(y_{\mathrm{V}_i}-y_{\mathrm{V}_j})} \bmod p.$$
3. For each received r'_{V_j} ($j = 1, 2, \ldots, t_3$; $j \neq i$), each $U_{\mathrm{V}_i} \in D_\mathrm{V}$ computes
$$R' = \prod_{i=1}^{t_3} r'_{\mathrm{V}_i} \bmod p.$$
4. Then, each verifier $U_{\mathrm{V}_i} \in D_\mathrm{V}$ can check the validity of the proxy signature of the message m through the following equation:

$$g^S \overset{?}{=} Y_\mathrm{P}^{R'} R^R \left(K^K \prod_{i=1}^{t_1} y_{\mathrm{O}_i}^{h(K\|m_\mathrm{w}\|\mathrm{AOSID})} \prod_{j=1}^{t_2} y_{\mathrm{P}_j} \right)^{h(R\|\mathrm{APSID}\|m)} \bmod p. \tag{2}$$

If the equation holds, the message m is authenticated and the proxy signature (m_w, K, AOSID, R, S, APSID) is valid.

3. Security analysis of the proposed scheme

The security of the proposed scheme is based on the well-known difficulty of computing one-way hash function and the discrete logarithm problem cryptographic assumptions. In the following paragraphs, several possible attacks will be considered, and we shall also prove that none of them can successfully break the proposed scheme.

In proposed scheme, all of the actual original signer's secret keys x_{O_i}, $i = 1, 2, \ldots, t_1$, are used in the proxy share generation phase to create a proxy share σ. Thus, it is necessary for the proxy group to verify the proxy share and the verifier group to verify the proxy signature verification equation (2) by using all of the actual original signer's public keys. These actual original signer's public keys are certified by the CA. Without knowing the original signer's secret keys, an opponent is unable to generate the proxy share σ in the proposed scheme. Assume that an opponent wants to derive the original signers' secret key x_{O_i} from the public keys $y_{\mathrm{O}_i} = g^{x_{\mathrm{O}_i}} \bmod p$. It means this attacker has to face the difficulty of solving the discrete logarithm problem. At the same time, because of the existence of the proxy share, the original group cannot deny delegating their signing capability to a proxy group.

Similarly, all of the actual proxy signers' secret keys x_{P_i}, $i = 1, 2, \ldots, t_2$ are used to generate the proxy signature in the proxy signature generation phase. Thus, it is necessary for the verifier group to verify the proxy signature verification equation (2) by using all of the actual proxy signers' public keys. These actual proxy signers' public keys are certified by the CA. Assume that an opponent wants to reveal the proxy signers' secret key x_{P_i} from the public keys $y_{\mathrm{P}_i} = g^{x_{\mathrm{P}_i}} \bmod p$. The equation is as difficult to meet as solving the discrete logarithm problem. Again, the proxy group cannot deny generating the proxy signature on behalf of the proxy group and the original signer group.

Without loss of generality, if the opponent tries to derive the proxy group secret key X_P and verifier group secret key X_V from their corresponding public keys $Y_\mathrm{P} = g^{X_\mathrm{P}} \bmod p$ and $Y_\mathrm{V} = g^{X_\mathrm{V}} \bmod p$, she/he also has to solve the same problem. As to the forgery attack, we consider the security of the proxy signature verification equation:

$$g^S \overset{?}{=} Y_\mathrm{P}^{R'} R^R \times \left(K^K \prod_{i=1}^{t_1} y_{\mathrm{O}_i}^{h(K\|m_\mathrm{w}\|\mathrm{AOSID})} \prod_{j=1}^{t_2} y_{\mathrm{P}_j} \right)^{h(R\|\mathrm{APSID}\|m)} \bmod p.$$

In this case, an outsider may try to forge a valid proxy signature to pass the proxy signature verification. Suppose

$$V_{\mathrm{O}} = K^K \prod_{i=1}^{t_1} y_{\mathrm{O}_i}^{h(K\|m_{\mathrm{w}}\|\mathrm{AOSID})} \bmod p,$$

$$V_{\mathrm{P}} = \prod_{j=1}^{t_2} y_{\mathrm{P}_j} \bmod p.$$

We rewrite the proxy signature verification equation as

$$g^S = Y_{\mathrm{P}}^{R'} R^R (V_{\mathrm{O}} V_{\mathrm{P}})^{h(R\|\mathrm{APSID}\|m)} \bmod p.$$

The value of V_{O} depends on the parameters K, m_{w} and AOSID, while V_{P} is a fixed value as the proxy signers' public keys are certified by the CA. Similarly, Y_{P} is a fixed value as the proxy signer group public key is certified by the CA. Given m^*, APSID*, V_{O}^* and R'^*, it is hard to determine R^* and S^* because of the difficulty of solving the discrete logarithm problem and the one-way hash function cryptographic assumptions. Again, given m^*, APSID*, R'^*, R^* and S^*, one can compute a V_{O}^* such that this equation holds. However, it is difficult to find m_{w}^*, AOSID* and K^* such that the equation

$$V_{\mathrm{O}} = K^K \prod_{i=1}^{t_1} y_{\mathrm{O}_i}^{h(K\|m_{\mathrm{w}}\|\mathrm{AOSID})} \bmod p$$

holds. The difficulty here is also based on the discrete logarithm problem and the one-way hash function cryptographic assumptions. Therefore, the proxy signature verification equation is secure to against forgery attack.

Consider the conspiracy attack. Assume that $t_2 - 1$ proxy signers of the group G_{P} conspire to derive some other proxy signers' secret keys and the proxy group secret key. They will have to first reconstruct the polynomial function $f_{\mathrm{P}}(x)$ and compute the other proxy signers' secret shadows $f_{\mathrm{P}}(y_{\mathrm{P}_i})$ for some $U_{\mathrm{P}_i} \in G_{\mathrm{P}}$ and also obtain the proxy group secret key $f_{\mathrm{P}}(0)$ for G_{P}. However, the secret polynomial function $f_{\mathrm{P}}(x)$ can be reconstructed only by obtaining at least t_2 proxy signers' secret shadows $f_{\mathrm{P}}(y_{\mathrm{P}_i})$'s of G_{P}. Thus, any t_2-1 conspirators or less cannot reveal any other proxy signers' secret shadows and the proxy group secret key, even though they release their secret shadows to each other. Similarly, any $t_3 - 1$ verifiers or less in the verifier group G_{V} cannot obtain any other verifiers' secret shadows. Therefore, neither proxy signers' conspiracy attack nor verifiers' conspiracy attack can be successful.

Assume any t_2 or more proxy signers in G_{P} work together to reconstruct the secret polynomial function $f_{\mathrm{P}}(x)$ and proxy group secret key. Thus, they can easily derive any other proxy signer U_{P_j}'s secret shadow. However, they cannot derive P_j's secret key from x_{P_i} via Eq. (1) with the given s_j because to obtain b_j means to solve the discrete logarithm problem.

4. Conclusions

A new threshold multi-proxy multi-signature scheme with (t_3, n_3) shared verification is proposed in this paper. In the proposed scheme, a subset of original signers can authenticate a designated proxy group to sign on behalf of the original group. A message m has to be signed by a subset of proxy signers who can represent the proxy group. Then, the proxy signature is sent to the verifier group. A subset of verifiers in the verifier group can also represent the group to authenticate the proxy signature. Furthermore, these actual proxy signers cannot deny the fact that they have signed the proxy signature. Based on the difficulty of the one-way hash function and the discrete logarithm problem, the security of the proposed scheme is confirmed. Some possible attacks are considered, and none of them can successfully break the proposed scheme.

Acknowledgements

This research was partially supported by the National Science Council, Taiwan, ROC, under contract no. NSC90-2213-E-324-004.

References

[1] C.-L. Hsu, T.-S. Wu, T.-C. Wu, New nonrepudiable threshold proxy signature scheme with known signers, J. Syst. Softw. (58) (2) (2001) 119–124.

[2] C.-L. Hsu, T.-C. Wu, Authenticated encryption scheme with (t, n) shared verification, IEE Proc. Comput. Digit. Technol. 145 (2) (1998) 117–120.

[3] M.S. Hwang, I.C. Lin, E.J.L. Lu, A secure nonrepudiable threshold proxy signature scheme with known signers, Int. J. Inform. 11 (2) (2000) 1–8.

[4] S. Kim, S. Park, D. Won, Proxy signatures, revisited, in: Proceedings of the ICICS'97, Lecture Notes in Computer Science, vol. 1334, 1997, pp. 223–232.

[5] N.Y. Lee, T. Hwang, C.H. Wang, On Zhang's nonrepudiable proxy signature schemes, in: Proceedings of the ACISP'98, Lecture Notes in Computer Science, vol. 1438, July 1998, pp. 415–422.

[6] M. Mambo, K. Usuda, E. Okamoto, Proxy signatures: delegation of the power to sign message, IEICE Trans. Fund. E79-A (1996) 1338–1353.

[7] M. Mambo, K. Usuda, E. Okamoto, Proxy signatures for delegating signing operation, in: Proceedings of the Third ACM Conference on Computer and Communications Security, 1996, pp. 48–57.

[8] H.M. Sun, An efficient nonrepudiable threshold proxy signature scheme with known signers, Comput. Commun. 22 (8) (1999) 717–722.

[9] S.-F. Tzeng, C.-Y. Yang, M.-S. Hwang, A new multi-proxy multi-signature scheme, Technical Report No. CYUT-IM-TR-2002-004, CYUT, 2002.

[10] C.-T. Wang, C.-C. Cheng, C.-H. Lin, Generalization of threshold signature and authenticated encryption for group communications, IEICE Trans. Fund. E83-A (6) (2000) 1228–1237.

[11] C.-Y. Yang, S.-F. Tzeng, M.-S. Hwang, A new nonrepudiable threshold proxy signature scheme with valid delegation period, Technical Report No. CYUT-IM-TR-2002-005, CYUT, 2002.

[12] C.-Y. Yang, S.-F. Tzeng, M.-S. Hwang, A nonrepudiable threshold proxy signature scheme with known signers, Technical Report No. CYUT-IM-TR-2002-003, CYUT, 2002.

[13] L. Yi, G. Bai, G. Xiao, Proxy multi-signature scheme: a new type of proxy signature scheme, Electron. Lett. 36 (6) (2000) 527–528.

[14] K. Zhang, Threshold proxy signature schemes, in: Proceedings of the 1997 Information Security Workshop, 1997, pp. 191–197.