Available online at www.sciencedirect.com
SCIENCE DIRECT
ELSEVIER
Journal of Computational and Applied Mathematics 195 (2006) 229–241
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

Factoring based proxy signature schemes☆

Qingshui Xue[a, b], Zhenfu Cao[a, *]

[a] *Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200030, China*
[b] *Department of Basic Theory, Shandong P.E. Institute, Jinan 250063, China*

Received 15 August 2005

Abstract

In proxy signature schemes, original signers delegate proxy signers to sign messages on behalf of original signers. Currently, most proxy signature schemes are based on the difficulty of discrete logarithms over finite field or ellipse curve addition group. Based on the difficulty of factorings of large integers, one normal proxy signature scheme and one multi-proxy signature scheme are proposed. The security properties of strong unforgeability, verifiability, strong nonrepudiation, strong identifiability, distinguishability and prevention of misuse of proxy signing power can be fulfilled by the proposed schemes. In addition, the performance of the proposed schemes is analysed as well.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Cryptograph; Digital signature; Proxy signature; Factoring; RSA

1. Introduction

In 1996, Mambo et al. [20,21] introduced the concept of one proxy signature scheme, which allows an original signer to delegate his/her signing power to a designated signer, called the proxy signer. The proxy signer could stand for the original signer to generate signatures, referred to as proxy signatures.

☆ Supported by the National Science Fund for Distinguished Young Scholars (Grant No. 60225007), the National Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20020248024) and the Key Project of Shanghai Fundamental Research Fund (Grant No. 04JC14055 and 04DZ07067).
* Corresponding author. Tel.:/fax: +86 021 62933504.
E-mail addresses: xue-qsh@cs.sjtu.edu.cn (Q. Xue), cao-zf@cs.sjtu.edu.cn (Z. Cao).

0377-0427/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2005.03.085

Mambo et al. [20,21] pointed out that proxy signature schemes should satisfy the following six properties: (1) Unforgeability: Except the original signer, only the designated proxy signer can create a valid proxy signature on behalf of the original signer. (2) Verifiability: From a proxy signature, the verifier or the receiver can confirm the original signer's agreement on the signed message. (3) Undeniability: Once a proxy signer generates a valid proxy signature on behalf of the original signer, he or she cannot deny the proxy signature that has been created by him or her. (4) Distinguishability: Anyone can differentiate the proxy signature from the original signer's normal signature. (5) Proxy signer's deviation: A proxy signer can create a proxy signature which can be detected to be valid. (6) Identifiability: The original signer can identify the identity of the corresponding proxy signer from one proxy signature.

In order to make a proxy signature scheme fairer to the original signer and the proxy signer, Lee et al. [14,15] somewhat enhanced the properties introduced by Mambo et al. The properties stated in [14,15] are described in the following: (1) Strong unforgeability: A proxy signer can create a valid proxy signature on behalf of the original signer. However, the original signer and any third party cannot generate a valid proxy signature with the name of proxy signers. (2) Strong identifiability: From a proxy signature anyone can determine the identity of the corresponding proxy signer. (3) Strong undeniability: Once a proxy signer generates a valid proxy signature on behalf of the proxy signer, the proxy signer cannot deny his signature generation against anyone. (4) Prevention of misuse: It should be confident that the proxy key pair cannot be used for other purposes. In the case of misuse, the responsibility of proxy signers should be determined explicitly.

Since proxy signature schemes were introduced, many proxy signature schemes have been proposed [1,3–32]. There are several kinds of proxy signature schemes. In a (t, n) threshold proxy signature scheme, an original signer can authorize a proxy group with n proxy members. Only the cooperation of t or more proxy members is allowed to generate proxy signatures. Until now, lots of threshold proxy signature schemes have been proposed [4,16,25,28,30,32]. In [10], the multi-proxy signature scheme was first proposed. It is a special case of threshold proxy signature schemes, which allows an original signer to authorize a group of proxy members to generate multi-proxy signatures on behalf of the original signer. In 2000, Yi et al. first proposed proxy multi-signature schemes [31]. Then some proxy multi-signature schemes were proposed [5,26]. In the type of scheme, an original signer group can authorize a proxy signer to sign messages on behalf of the original signer group. In [6,7], a new kind of proxy signature scheme, multi-proxy multi-signature schemes, was first proposed by Hwang et al. In the kind of scheme, only the cooperation of all members in the original group can authorize a proxy group. Only the cooperation of all members in the authorized proxy group could sign messages on behalf of the original group.

In 2003, Shao proposed a proxy signature scheme based on the factoring of large integers [24]. However, by our observation, his scheme is not based on factorings, but the RSA cryptosystem, for the difficulty of the RSA cryptosystem is reduced to factorings, but equivalent to factorings.

In 2001, Cao [2] proposed an improved RSA cryptosystem which is really based on the difficulty of factorings of large integers. In the paper, based on Cao's scheme, one normal proxy signature scheme and one multi-proxy signature scheme will be put forward by us. In addition, we will analyze and discuss their security and performance as well.

In the rest of the paper, we will organize the content as follows. In Section 2, we will state Cao's improved RSA cryptosystem. In Section 3, we will propose one normal proxy signature scheme and one multi-proxy signature scheme based on the improved RSA cryptosystem. As to the security of the both schemes, we will discuss them in Section 4. In Section 5, we will analyze the performance of the both schemes. Finally, the conclusion is given in Section 6.

2. An improved RSA cryptosystem [2]

Select two large primes p, q at random, satisfying $p \equiv q \equiv 3 \bmod 4$, here p, q can be taken as security primes. Let $N = pq$, then $\varphi(N) = (p-1) * (q-1)$. Choose b satisfying Jacobi symbol $(\frac{b}{N}) = -1$, and then take $e \in Z$ with $\gcd(e, \frac{1}{4}\varphi(N)) = 1, 1 < e < \frac{1}{4}\varphi(N)$, $d \in Z$, can be computed by $ed \equiv \frac{1}{2}(\frac{1}{4}\varphi(N)+1)(\mathrm{mod}(\frac{1}{4}\varphi(N)))$. Finally b, e, N are published as encrypt key/verification key and d as secret key/signature key. The improved RSA cryptosystem is detailed as follows.

Encrypt algorithm: Assume that plaintext $x \in Z_N$, $\gcd(x, N) = 1$. Then

$$E(x) = \begin{cases} x^{2e} \bmod N, & \left(\frac{x}{N}\right) = 1, \\ (bx)^{2e} \bmod N, & \left(\frac{x}{N}\right) = -1. \end{cases} \tag{1}$$

Ciphertext is $(E(x), c_1, c_2)$, where

$$c_1 = \begin{cases} 0, & 2|x, \\ 1, & 2\nmid x, \end{cases} \qquad c_2 = \begin{cases} 0, & \left(\frac{x}{N}\right) = 1, \\ 1, & \left(\frac{x}{N}\right) = -1. \end{cases} \tag{2}$$

Deciphering algorithm: If $c_2 = 0$, then $x^{2e} = E(x)(\mathrm{mod}\, N)$. Compute

$$E(x)^d \equiv x^{2ed} \equiv x^{1+\frac{1}{4}\varphi(N)} \equiv \pm x \bmod N. \tag{3}$$

The parity of x can be known from identifier digit c_1. Then plaintext x can be obtained. If $c_2 = 1$, then $(bx)^{2e} \equiv E(x)(\mathrm{mod}\, N)$. Compute

$$E(x)^d \equiv (bx)^{2ed} \equiv (bx)^{(1+\frac{1}{4}\varphi(N))} \equiv \pm bx(\mathrm{mod}\, N). \tag{4}$$

That is, $x = \pm b^{-1}E(x)^d(\mathrm{mod}\, N)$. Plaintext x can be gotten from identifier digit c_1.

As to the correctness of the deciphering algorithm, there is the following theorem.

Theorem 2.1. *Suppose that there exists a polynomial time algorithm for finding the plaintext from any ciphertext of the improved RSA cryptosystem. Then there exists a polynomial time algorithm for factoring N.*

The proof of Theorem 1 is omitted here. Interested readers can refer to [2].

In the following section, based on the improved RSA cryptosystem, two types of proxy signature scheme including normal proxy signature schemes and multi-proxy signature schemes are proposed. Because the security of the improved RSA cryptosystem is based on the factoring of large integers, our schemes are also based on factorings of large integers. The schemes are stated in the following.

3. The proxy signature schemes based on the improved RSA cryptosystem

In the proposed schemes, there are four participants: the original signer, the proxy signers, the verifier and Certificate Authority (CA).

Throughout the paper, U_0 stands for the original signer and $U_{P1}, U_{P2}, \ldots, U_{Pn}$ stand for n proxy signers. For user U_i his public RSA modulus is $N_i = p_i * q_i$, where p_i and q_i are two secret large primes and $p_i \equiv q_i \equiv 3(\mathrm{mod}\,4)$. Let $U_i's$ private key be $1 < d_i < \frac{1}{4}\varphi(N_i)$ and its corresponding public key be $1 < e_i < \frac{1}{4}\varphi(N_i)$, such that $e_i d_i \equiv \frac{1}{2}(\frac{1}{4}\varphi(N_i) + 1)(\mathrm{mod}(\frac{1}{4}\varphi(N_i)))$, where $\varphi(N_i) = (p_i - 1) * (q_i - 1)$, and it is certified by CA. m is the message to be signed or encrypted. b satisfies Jacobi symbol $(\frac{b}{N_i}) = -1$. m_ω is a warrant which is generated by the original signer and contains the identities of the original signer and only one proxy signer or n proxy signers, the validity period of the proxy key, etc.

3.1. A normal proxy signature scheme

In the normal proxy signature scheme, only one original signer delegates one proxy signer, say, U_{Pi}, to sign on behalf of the original signer.

3.1.1. Proxy generation phase

Step 1: The original signer U_0 computes

$$v_i = \begin{cases} h(m_\omega, \mathrm{ID}_{Pi})^{2d_0} \bmod N_0, & \left(\dfrac{h(m_\omega, \mathrm{ID}_{Pi})}{N_0}\right) = 1, \\ (bh(m_\omega, \mathrm{ID}_{Pi}))^{2d_0} \bmod N_0, & \left(\dfrac{h(m_\omega, \mathrm{ID}_{Pi})}{N_0}\right) = -1. \end{cases} \tag{5}$$

Step 2: U_0 also computes

$$u_i = \lfloor v_i / N_{Pi} \rfloor, \tag{6}$$

$$w_i = \begin{cases} v_i^{2e_{Pi}} \bmod N_{Pi}, & \left(\dfrac{v_i}{N_{Pi}}\right) = 1, \\ (bv_i)^{2e_{Pi}} \bmod N_{Pi}, & \left(\dfrac{v_i}{N_{Pi}}\right) = -1, \end{cases} \tag{7}$$

$$c'_{i1} = \begin{cases} 0, & 2|v_i, \\ 1, & 2\nmid v_i, \end{cases} \quad c'_{i2} = \begin{cases} 0, & \left(\dfrac{v_i}{N_{Pi}}\right) = 1, \\ 1, & \left(\dfrac{v_i}{N_{Pi}}\right) = -1. \end{cases} \tag{8}$$

Step 3: The original signer U_0 sends $(m_\omega, u_i, w_i, c'_{i1}, c'_{i2})$ to the proxy signer U_{Pi}.
Step 4: The proxy signer U_{Pi} first gets v'_i from (w_i, c'_{i1}, c'_{i2}), that is
If $c'_{i2} = 0$, then $w_i^{d_{Pi}} \equiv \pm v'_i (\mathrm{mod}\, N_{Pi})$. The parity of v'_i can be known from identifier digit c'_{i1}.
If $c'_{i2} = 1$, then $w_i^{d_{Pi}} \equiv \pm b v'_i (\mathrm{mod}\, N_{Pi})$. The parity of v'_i can be known from identifier digit c'_{i1}.
Then U_{Pi} computes $v_i = u_i * N_{Pi} + v'_i$.

Step 5: The proxy signer U_{Pi} will confirm the validity of v_i, that is, U_{Pi} first computes

$$c_{i1} = \begin{cases} 0, & 2|h(m_\omega, \mathrm{ID}_{Pi}), \\ 1, & 2\nmid h(m_\omega, \mathrm{ID}_{Pi}), \end{cases} \qquad c_{i2} = \begin{cases} 0, & \left(\dfrac{h(m_\omega, \mathrm{ID}_{Pi})}{N_{N_0}}\right) = 1, \\ 1, & \left(\dfrac{h(m_\omega, \mathrm{ID}_{Pi})}{N_{N_0}}\right) = -1. \end{cases} \tag{9}$$

If $c_{i2} = 0$, then $v_i^{e_0} \equiv \pm x (\mathrm{mod}\ N_0)$. The parity of x can be known from identifier digit c_{i1}.

If $c_{i2} = 1$, then $v_i^{e_0} \equiv \pm bx (\mathrm{mod}\ N_0)$. The parity of x can be known from identifier digit c_{i1}.

Then the proxy signer U_{Pi} checks whether the following congruence holds:

$$x = h(m_\omega, \mathrm{ID}_{Pi}). \tag{10}$$

If it holds, the proxy signer will regard v_i as his proxy signing key, or else the protocol will fail.

3.1.2. Proxy signature generation phase

Step 1: The proxy signer U_{Pi} chooses $t_i \in [1, n_0]$ at random and computes

$$r_i = \begin{cases} t_i^{2e_0} \bmod N_0, & \left(\dfrac{t_i}{N_0}\right) = 1, \\ (bt_i)^{2e_0} \bmod N_0, & \left(\dfrac{t_i}{N_0}\right) = -1, \end{cases} \qquad c_{i1} = \begin{cases} 0, & 2|t_i, \\ 1, & 2\nmid t_i, \end{cases}$$

$$c_{i2} = \begin{cases} 0, & \left(\dfrac{t_i}{N_0}\right) = 1, \\ 1, & \left(\dfrac{t_i}{N_0}\right) = -1. \end{cases} \tag{11}$$

Step 2: U_{Pi} also computes

$$k_i = h(m, r_i, c_{i1}, c_{i2}), \tag{12}$$

$$s_i = \begin{cases} h(m, r_i, c_{i1}, c_{i2})^{2d_{Pi}} \bmod N_{Pi}, & \left(\dfrac{h(m, r_i, c_{i1}, c_{i2})}{N_{Pi}}\right) = 1, \\ (bh(m, r_i, c_{i1}, c_{i2}))^{2d_{Pi}} \bmod N_{Pi}, & \left(\dfrac{h(m, r_i, c_{i1}, c_{i2})}{N_{Pi}}\right) = -1, \end{cases} \tag{13}$$

$$c_{si1} = \begin{cases} 0, & 2|h(m, r_i, c_{i1}, c_{i2}), \\ 1, & 2\nmid h(m, r_i, c_{i1}, c_{i2}), \end{cases} \qquad c_{si2} = \begin{cases} 0, & \left(\dfrac{h(m, r_i, c_{i1}, c_{i2})}{N_{Pi}}\right) = 1, \\ 1, & \left(\dfrac{h(m, r_i, c_{i1}, c_{i2})}{N_{Pi}}\right) = -1, \end{cases} \tag{14}$$

$$y_i = \begin{cases} t_i v_i^{ki} \bmod N_0, & \left(\dfrac{t_i}{N_0}\right) = 1, \\ bt_i v_i^{ki} \bmod N_0, & \left(\dfrac{t_i}{N_0}\right) = -1. \end{cases} \tag{15}$$

Step 3: U_{Pi} sends the proxy signature $(m, m_\omega, y_i, s_i, c_{i1}, c_{i2}, c_{si1}, c_{si2})$ to the verifier or receiver.

3.1.3. Proxy signature verification phase

Step 1: The verifier first checks the validity of m_ω, if it can pass the verification, the verifier can obtain the identities of the original signer and the proxy signer, then can acquire the public keys of the original signer and the proxy signer from CA.

Step 2: If $c_{si2}=0$, then $s_i^{e_{Pi}} \equiv \pm k_i \pmod{N_{Pi}}$. The parity of k_i can be known from identifier digit c_{si1}. If $c_{si2}=1$, then $s_i^{e_{Pi}} \equiv \pm bk_i \pmod{N_{Pi}}$. The parity of k_i can be known from identifier digit c_{si1}.

Step 3: The verifier computes

$$y_i^{2e_0} \cdot h(m_\omega, \mathrm{ID}_{Pi})^{-2k_i} = r_i \bmod N_0, c_{i2}=0 \tag{16}$$

or

$$y_i^{2e_0} \cdot (bh(m_\omega, \mathrm{ID}_{Pi}))^{-2k_i} = r_i \bmod N_0, c_{i2}=1. \tag{17}$$

Step 4: The verifier checks whether the following congruence holds:

$$k_i = h(m, r_i, c_{i1}, c_{i2}). \tag{18}$$

If it also holds, the proxy signature $(m, m_\omega, y_i, s_i, c_{i1}, c_{i2}, c_{si1}, c_{si2})$ is valid, or else the verifier will reject it.

3.2. A multi-proxy signature scheme

In the proposed scheme, $U_{P1}, U_{P2}, \ldots, U_{Pn}$ are n proxy signers. Here assume $N_{P1} \leqslant N_{P2} \leqslant \cdots \leqslant N_{Pn}$.

3.2.1. Proxy generation phase

The phase is the same as that of the normal proxy signature scheme.

3.2.2. Proxy signature generation phase

Step 1: The proxy signer U_{P1} chooses $t_1 \in [1, n_0]$ at random and computes

$$r_1 = \begin{cases} t_1^{2e_0} \bmod N_0, & \left(\dfrac{t_1}{N_0}\right) = 1, \\ (bt_1)^{2e_0} \bmod N_0, & \left(\dfrac{t_1}{N_0}\right) = -1, \end{cases} \quad c_{11} = \begin{cases} 0, & 2|t_1, \\ 1, & 2\nmid t_1, \end{cases} \quad c_{12} = \begin{cases} 0, & \left(\dfrac{t_1}{N_0}\right) = 1, \\ 1, & \left(\dfrac{t_1}{N_0}\right) = -1 \end{cases} \tag{19}$$

and sends (r_1, c_{11}, c_{12}) to U_{P2}.

Step 2: The proxy signer U_{P2} chooses $t_2 \in [1, n_0]$ at random and computes

$$r_2 = \begin{cases} r_1 t_2^{2e_0} \bmod N_0, & \left(\dfrac{t_2}{N_0}\right) = 1, \\ r_1 (bt_2)^{2e_0} \bmod N_0, & \left(\dfrac{t_2}{N_0}\right) = -1, \end{cases} \quad c_{21} = \begin{cases} 0, & 2|t_2, \\ 1, & 2\nmid t_2, \end{cases}$$

$$c_{22} = \begin{cases} 0, & \left(\dfrac{t_2}{N_0}\right) = 1, \\ 1, & \left(\dfrac{t_2}{N_0}\right) = -1 \end{cases} \tag{20}$$

and sends $(r_2, c_{11}, c_{12}, c_{21}, c_{22})$ to U_{P3}.

...

Step n: The proxy signer U_{P_n} chooses $t_n \in [1, n_0]$ at random and computes

$$r = r_n = \begin{cases} r_{n-1} t_n^{2e_0} \bmod N_0, & \left(\frac{t_n}{N_0}\right) = 1, \\ r_{n-1}(bt_n)^{2e_0} \bmod N_0, & \left(\frac{t_n}{N_0}\right) = -1, \end{cases} \quad c_{n1} = \begin{cases} 0, & 2|t_n, \\ 1, & 2\nmid t_n, \end{cases}$$

$$c_{n2} = \begin{cases} 0, & \left(\frac{t_n}{N_0}\right) = 1, \\ 1, & \left(\frac{t_n}{N_0}\right) = -1 \end{cases}$$

and broadcasts $(r, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2})$. Then each of proxy signers can compute $k = h(m, r, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2})$.

Step n + 1: U_{P_1} computes

$$c_{s11} = \left(\frac{h(m, r, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2})}{N_{P1}}\right), \tag{21}$$

$$s_1 = \begin{cases} h(m, r, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2})^{2d_{P1}} (\bmod N_{P1}), & c_{s11} = 1, \\ (bh(m, r, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}))^{2d_{P1}} (\bmod N_{P1}), & c_{s11} = -1, \end{cases} \tag{22}$$

$$c_{s12} = \begin{cases} 0, & 2|h(m, r, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}), \\ 1, & 2\nmid h(m, r, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}), \end{cases} \tag{23}$$

$$y_1 = \begin{cases} t_1 v_1^k \bmod N_0, & \left(\frac{t_1}{N_0}\right) = 1, \\ bt_1 v_1^k \bmod N_0, & \left(\frac{t_1}{N_0}\right) = -1. \end{cases} \tag{24}$$

U_{P_1} sends $(y_1, s_1, c_{s11}, c_{s12})$ to U_{P_2}.

Step n + 2: U_{P_2} computes

$$s_2 = \begin{cases} s_1^{2d_{P2}} (\bmod N_{P2}), & \left(\frac{s_1}{N_{P2}}\right) = 1, \\ (bs_1)^{2d_{P2}} (\bmod N_{P2}), & \left(\frac{s_1}{N_{P2}}\right) = -1, \end{cases} \quad c_{s21} = \begin{cases} 0, & 2|s_1, \\ 1, & 2\nmid s_1, \end{cases} \tag{25}$$

$$c_{s22} = \left(\frac{s_1}{N_{P2}}\right), \quad y_2 = \begin{cases} y_1 t_2 v_2^k \bmod N_0, & \left(\frac{t_2}{N_0}\right) = 1, \\ y_1 bt_2 v_2^k \bmod N_0, & \left(\frac{t_2}{N_0}\right) = -1. \end{cases} \tag{26}$$

U_{P_2} sends $(y_2, s_2, c_{s11}, c_{s12}, c_{s21}, c_{s22})$ to U_{P_3}.

...

Step $n+n$: U_{P_n} computes

$$s = s_n = \begin{cases} s_{n-1}^{2d_{Pn}} (\bmod N_{Pn}), & \left(\dfrac{s_{n-1}}{N_{Pn}}\right) = 1, \\ (bs_{n-1})^{2d_{Pn}} (\bmod N_{Pn}), & \left(\dfrac{s_{n-1}}{N_{Pn}}\right) = -1, \end{cases} \quad c_{sn1} = \begin{cases} 0, & 2|s_{n-1}, \\ 1, & 2\nmid s_{n-1}, \end{cases} \tag{27}$$

$$c_{sn2} = \left(\frac{s_{n-1}}{N_{Pn}}\right), \quad y = y_n = \begin{cases} y_{n-1} t_n v_n^k \bmod N_0, & \left(\dfrac{t_n}{N_0}\right) = 1, \\ y_{n-1} b t_n v_n^k \bmod N_0, & \left(\dfrac{t_n}{N_0}\right) = -1. \end{cases} \tag{28}$$

U_{P_n} sends the proxy signature $(m, m_\omega, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}, y, s, c_{s11}, c_{s12}, \ldots, c_{sn1}, c_{sn2})$ to the verifier or receiver.

3.2.3. Proxy signature verification phase

Step 1: The verifier first checks the validity of m_ω, if it can pass the verification, the verifier can obtain the public key of the original signer, the identity ID_i and the public key of the proxy signer U_{Pi} $(i = 1, 2, \ldots, n)$ from CA.

Step 2: If $c_{sn2} = 1$, then $s^{e_{Pn}} \equiv \pm s_{n-1} (\bmod N_{Pn})$. The parity of s_{n-1} can be known from identifier digit c_{sn1}. If $c_{sn2} = -1$, then $s^{e_{Pn}} \equiv \pm b s_{n-1} (\bmod N_{Pn})$. The parity of s_{n-1} can be known from identifier digit c_{sn1}.

...

If $c_{s22} = 1$, then $s_2^{e_{P2}} \equiv \pm s_1 (\bmod N_{P2})$. The parity of s_1 can be known from identifier digit c_{s21}. If $c_{s22} = -1$, then $s_2^{e_{P2}} \equiv \pm b s_1 (\bmod N_{P2})$. The parity of s_1 can be known from identifier digit c_{s21}.

If $c_{s12} = 1$, then $s_1^{e_{P1}} \equiv \pm k (\bmod N_{P1})$. The parity of k can be known from identifier digit c_{s11}. If $c_{s12} = -1$, then $s_1^{e_{P1}} \equiv \pm bk (\bmod N_{P1})$. The parity of k can be known from identifier digit c_{s11}.

Step 3: The verifier computes

$$y^{2e_0} \cdot (h(m_\omega, \mathrm{ID}_{P1}) \cdot \cdots \cdot h(m_\omega, \mathrm{ID}_{Pn}))^{2k} (\bmod N_0) = r \tag{29}$$

and checks whether the following congruence holds:

$$k = h(m, r, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}). \tag{30}$$

If it also holds, the proxy signature $(m, m_\omega, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}, y, s, c_{s11}, c_{s12}, \ldots, c_{sn1}, c_{sn2})$ is valid, or else the verifier will reject it.

Theorem 3.1. *The verification Eq.* (30) *holds if* $(m, m_\omega, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}, y, s, c_{s11}, c_{s12}, \ldots, c_{sn1}, c_{sn2})$ *is one valid proxy signature.*

Proof.

$$
\begin{aligned}
y^{2e_0} \bmod N_0 = {} & \\
y_n^{2e_0} = {} & \begin{cases} y_{n-1}^{2e_0} t_n^{2e_0} v_n^{2e_0 k} (\bmod N_0), & \left(\frac{t_n}{N_0}\right) = 1 \\ y_{n-1}^{2e_0} (bt_n)^{2e_0} v_n^{2e_0 k} (\bmod N_0), & \left(\frac{t_n}{N_0}\right) = -1 \end{cases} \\
= {} & \begin{cases} y_{n-2}^{2e_0} (t_{n-1}t_n)^{2e_0} v_{n-1}^{2e_0 k} v_n^{2e_0 k} (\bmod N_0), & \left(\frac{t_n}{N_0}\right) = 1, \left(\frac{t_{n-1}}{N_0}\right) = 1 \\ y_{n-2}^{2e_0} (bt_{n-1}t_n)^{2e_0} v_{n-1}^{2e_0 k} v_n^{2e_0 k} (\bmod N_0), & \left(\frac{t_n}{N_0}\right) = 1, \left(\frac{t_{n-1}}{N_0}\right) = -1 \\ y_{n-2}^{2e_0} (t_{n-1}bt_n)^{2e_0} v_{n-1}^{2e_0 k} v_n^{2e_0 k} (\bmod N_0), & \left(\frac{t_n}{N_0}\right) = -1, \left(\frac{t_{n-1}}{N_0}\right) = 1 \\ y_{n-2}^{2e_0} (bt_{n-1}bt_n)^{2e_0} v_{n-1}^{2e_0 k} v_n^{2e_0 k} (\bmod N_0), & \left(\frac{t_n}{N_0}\right) = -1, \left(\frac{t_{n-1}}{N_0}\right) = -1 \end{cases} \\
= {} & \begin{cases} y_{n-2}^{2e_0} (t_{n-1}t_n)^{2e_0} h(m_\omega, \mathrm{ID}_{n-1})^{2k} h(m_\omega, \mathrm{ID}_n)^{2k} (\bmod N_0), & 1, 1 \\ y_{n-2}^{2e_0} (bt_{n-1}t_n)^{2e_0} h(m_\omega, \mathrm{ID}_{n-1})^{2k} h(m_\omega, \mathrm{ID}_n)^{2k} (\bmod N_0), & 1, -1 \\ y_{n-2}^{2e_0} (t_{n-1}bt_n)^{2e_0} h(m_\omega, \mathrm{ID}_{n-1})^{2k} h(m_\omega, \mathrm{ID}_n)^{2k} (\bmod N_0), & -1, 1 \\ y_{n-2}^{2e_0} (bt_{n-1}bt_n)^{2e_0} h(m_\omega, \mathrm{ID}_{n-1})^{2k} h(m_\omega, \mathrm{ID}_n)^{2k} (\bmod N_0), & -1, -1 \end{cases} \\
= {} & \ldots \\
= {} & (b^j t_1 \cdot \ldots \cdot t_n)^{2e_0} (h(m_\omega, \mathrm{ID}_{P1}) \cdot \ldots \cdot h(m_\omega, \mathrm{ID}_{Pn}))^{2k} \bmod N_0,
\end{aligned}
$$

where j is the number of the element -1 in the set $\{c_{12}, c_{22}, \ldots, c_{n2}\}$.

$$
\begin{aligned}
r = r_n = {} & \begin{cases} r_{n-1} t_n^{2e_0} \bmod N_0, & \left(\frac{t_n}{N_0}\right) = 1 \\ r_{n-1} (bt_n)^{2e_0} \bmod N_0, & \left(\frac{t_n}{N_0}\right) = -1 \end{cases} \\
= {} & \begin{cases} r_{n-2} t_{n-1}^{2e_0} t_n^{2e_0} \bmod N_0, & \left(\frac{t_n}{N_0}\right) = 1, \left(\frac{t_{n-1}}{N_0}\right) = 1 \\ r_{n-2} (bt_{n-1})^{2e_0} t_n^{2e_0} \bmod N_0, & \left(\frac{t_n}{N_0}\right) = 1, \left(\frac{t_{n-1}}{N_0}\right) = -1 \\ r_{n-2} t_{n-1}^{2e_0} (bt_n)^{2e_0} \bmod N_0, & \left(\frac{t_n}{N_0}\right) = -1, \left(\frac{t_{n-1}}{N_0}\right) = 1 \\ r_{n-2} (bt_{n-1})^{2e_0} (bt_n)^{2e_0} \bmod N_0, & \left(\frac{t_n}{N_0}\right) = -1, \left(\frac{t_{n-1}}{N_0}\right) = -1 \end{cases} \\
= {} & \ldots \\
= {} & (b^j t_1 \cdot \ldots \cdot t_n)^{2e_0} \bmod N_0.
\end{aligned}
$$

Thus we can obtain $y^{2e_0} \cdot (h(m_\omega, \mathrm{ID}_{P1}) \cdot \ldots \cdot h(m_\omega, \mathrm{ID}_{Pn}))^{-2k} \bmod N_0 = (b^j t_1 \cdot \ldots \cdot t_n)^{2e_0} (h(m_\omega, \mathrm{ID}_{P1}) \cdot \ldots \cdot h(m_\omega, \mathrm{ID}_{Pn}))^{2k} (h(m_\omega, \mathrm{ID}_{P1}) \cdot \ldots \cdot h(m_\omega, \mathrm{ID}_{Pn}))^{-2k} = (b^j t_1 \cdot \ldots \cdot t_n)^{2e_0} \bmod N_0 = r$. Then we can have the verification Eq. (30).

4. Security analysis and discussion

4.1. Proxy protected

In the normal proxy signature scheme, the original signer can get $(r_i, y_i, c_{i1}, c_{i2})$ by selecting random $t_i \in [1, N_0]$, however, he/she cannot obtain s_i as he/she has no knowledge of the proxy signer's private key d_{Pi}. Similarly, in the multi-proxy signature scheme, the original signer can get $(c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}, y)$ by selecting $t_1, \ldots, t_n \in [1, N_0]$ at random, whereas, he/she cannot get s since he/she has not any knowledge of all proxy signers' private keys. Thus the original signer cannot forge a valid proxy signature $(m, m_\omega, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}, y, s, c_{s11}, c_{s12}, \ldots, c_{sn1}, c_{sn2})$. Therefore, the proposed schemes can provide the function of proxy protection.

4.2. Strong unforgeability

From Section 4.1, we know that on giving any message the original signer cannot create a valid proxy signature. Of course, the third party cannot generate any valid proxy signature on giving any message either. Similarly, less than n proxy signers cannot generate a valid proxy signature on given any message.

In addition, from the proxy signature $(m, m_\omega, y_i, c_{i1}, c_{i2})$ or $(m, m_\omega, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}, y, s, c_{s11}, c_{s12}, \ldots, c_{sn1}, c_{sn2})$, the original signer cannot generate another valid proxy signature on the same message m since from the verification equation, the original signer will be faced with the difficulty of solving factorings of large integers and breaking the secure one-way hash function.

In the multi-proxy signature scheme, less than n proxy signers cannot generate a valid proxy signature on any message since they have no knowledge of the other proxy signers' private keys. From the multi-proxy signature $(m, m_\omega, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}, y, s, c_{s11}, c_{s12}, \ldots, c_{sn1}, c_{sn2})$, less than n proxy signers cannot generate another valid proxy signature on the same message m due to the same reason either.

Third, it is evident that the third party cannot produce a valid proxy signature on any message or a known message for the third party knows less secret information of proxy signers than the original signer or less than n proxy signers.

From the three aspects, we can conclude that the proposed schemes can provide the security property of strong unforgeability.

4.3. Strong nonrepudiation

In the normal proxy signature scheme and the multi-proxy signature scheme, when the proxy signature is verified, proxy signers' identities ID_{Pi} and public keys e_{Pi} are used in the verification congruence, which means that the proxy signers cannot deny having signed the message on behalf of the original signer to any person.

4.4. Strong identifiability

As stated in Section 4.3, any person can verify the validity of the proxy signature $(m, m_\omega, y_i, c_{i1}, c_{i2})$ or $(m, m_\omega, c_{11}, c_{12}, \ldots, c_{n1}, c_{n2}, y, s, c_{s11}, c_{s12}, \ldots, c_{sn1}, c_{sn2})$ by using the proxy signers' public keys and identities. Thus any person can know the identities of the proxy signers. Therefore, the property of the strong identifiability can be gotten in the proposed schemes.

4.5. Distinguishability

In the proposed schemes, when the proxy signature is verified, not only proxy signers but also the original signer's public keys and identities are used in the verification phase, so we can regard it as a proxy signature, not a normal signature. Thus, any one can distinguish the proxy signature from normal signatures.

4.6. Verifiability

In the proposed schemes, on one hand, from the warrant m_ω, the verifier can know who are the original signer and the proxy signers. On the other hand, when the proxy signature is verified, the original signer and the proxy signers' public keys and identities are used in the verification congruence. Thus the original signer cannot deny having delegated his signing capability to the designated proxy signers. That is to say, a verifier can be convinced of the original signer's agreement on the signed message. Therefore, the proposed schemes can fulfill the security property.

5. Performance of the proposed schemes

To design proxy signature schemes really based on factorings of large integers, the improved RSA encryption and signature schemes are used. Compared with standard RSA cryptosystems, the procedure of the encryption and signature costs more computation overheads, and the procedure of the decryption and the signature verification costs almost the same computation overheads. Compared with the proxy-protected proxy signature schemes based on standard RSA cryptosystem proposed by Shao [3], in the proxy generation phase, the two proposed proxy signature schemes cost a little more computation and communication overheads; in the proxy signature generation phase, our schemes also cost a little more computation and communication overheads; in the proxy signature verification phase, our schemes cost nearly the same computation and communication overheads. Shao's schemes are based on the standard RSA cryptosystem whose difficulty is equivalent to the factoring, whereas, our schemes are based on the improved RSA cryptosystem whose difficulty is the factoring of large integers.

6. Conclusions

We have proposed one normal proxy signature scheme and one multi-proxy signature scheme which are really based on the difficulty of factorings of large integers. Both the schemes can provide the following security properties: (1) Proxy protected; (2) Verifiability; (3) Strong unforgeability; (4) Strong

nonrepudiation; (5) Strong identifiability; (6) Distinguishability. Besides, we have analyzed and compared the performance of the proposed schemes with that of Shao's scheme.

References

[1] A. Boldyreva, A. Palacio, B. Warinschi, Secure proxy signature schemes for delegation of signing rights, http://venona.antioffline.com/2003/096.pdf, 2003.
[2] Z.F. Cao, A threshold key escrow scheme based on public cryptosystem, Sci. China (Series E) 44 (4) (2001) 441–448.
[3] T.S. Chen, T.P. Liu, Y.F. Chung, A proxy-protected proxy signature scheme based on elliptic curve cryptosystem, Proceeding of IEEE TENCON, 2002, pp. 184–187.
[4] C.L. Hsu, T.S. Wu, T. C Wu, Improvement of threshold proxy signature scheme, Appl. Math. Comput. 136 (2003) 315–321.
[5] S.J. Hwang, C.C. Chen, A new proxy multi-signature scheme, International Workshop on Cryptology and Network Security 2001, pp. 26–28.
[6] S.J. Hwang, C.C Chen, A new multi-proxy multi-signature scheme, National Computer Symposium: Information Security, 2001, pp. F019–F026.
[7] S.J. Hwang, C.C. Chen, New multi-proxy multi-signature schemes, Appl. Math. Comput. 147 (1) (2004) 57–67.
[8] M.S. Hwang, E.J.L. Lu, I.C. Lin, A practical (t, n) threshold proxy signature scheme based on the RSA cryptosystem, IEEE Trans. Knowledge Data Eng. 15 (6) (2003) 1552–1560.
[9] S.J. Hwang, C.H. Shi, Specifiable proxy signature schemes, National Computer Symposium, 1999, pp. 190–197.
[10] S.J. Hwang, C.H. Shi, A simple multi-proxy signature scheme, Proceeding of the Tenth National Conference on Information Security, 2000, pp. 134–138.
[11] M.S. Hwang, S.F. Tzeng, C.S. Tsai, Generalization of proxy signature based on elliptic curves, Comput. Standards Interfaces 26 (2) (2004) 73–84.
[12] S. Kim, S. Park, D. Won, Proxy signatures, revisited, Proceedings of the ICICS'97, International Conference on Information and Communications Security, Lecture Notes in Computer Science, vol. 1334, 1997, pp. 223–232.
[13] N.Y. Lee, T. Hwang, C.H. Wang, On Zhang's nonrepudiable proxy signature schemes, Advanced in Cryptology ASICCRYPT'98(ACISP'98), Lecture Notes in Computer Science, vol. 1438, 1998, pp. 415–422.
[14] B. Lee, H. Kim, K. Kim, Secure mobile agent using strong non-designated proxy signature, Proceedings of ACISP2001, Lecture Notes in Computer Science, vol. 2119, 2001, pp. 474–486.
[15] B. Lee, H. Kim, K. Kim, Strong proxy signature and its application, Proceedings of ACISP2001, vol. 11B-1, 2001, pp. 603–608.
[16] J.G. Li, Z.F. Cao, Improvement of a threshold proxy signature scheme, J. Comput. Res. Dev. 39 (11) (2002) 515–518 (in Chinese).
[17] J.G. Li, Z.F. Cao, Y.C. Zhang, Improvement of M-U-O and K-P-W proxy signature schemes, J. Harbin Inst. Tech. (New Series) (2002) 145–148.
[18] J.G. Li, Z.F. Cao, Y.C. Zhang, Nonrepudiable proxy multi-signature schemes, J. Comput. Sci. Technol. 18 (3) (2003) 399–402.
[19] L.H. Li, S.F. Tzeng, M.S. Hwang, Generalization of proxy signature-based on discrete logarithms, Comput. Security 22 (3) (2003) 245–255.
[20] M. Mambo, K. Usuda, E. Okamoto, Proxy signature for delegating signing operation, Proceedings of the 3rd ACM Conference on Computer and Communications Security, 1996, pp. 48–57.
[21] M. Mambo, K. Usuda, E. Okamoto, Proxy signatures: delegation of the power to sign messages, IEICE Trans. Fund. E79-A (9) (1996) 1338–1354.
[22] B.C. Neuman, Proxy-based authorization and accounting for distributed systems, Proceedings of the 13th International Conference on Distributed Computing Systems, 1993, pp. 283–291.
[23] T. Pedersen, Distributed provers with applications to undeniable signatures, Proceedings of EUROCRYPT'91, Lecture Notes in Computer Science, vol. 547, 1991, pp. 221–238.
[24] Z.H. Shao, Proxy signature schemes based on factoring, Inform. Process. Lett. 85 (2003) 137–143.
[25] H.M. Sun, A efficient nonrepudiable threshold proxy signature scheme with known signers, Comput. Comm. 22 (8) (1999) 717–722.

[26] H.M. Sun, On proxy (multi-) signature schemes, International Computer Symposium, 2000, pp. 65–72.
[27] H.M. Sun, B.T. Hsieh, Remarks on two nonrepudiable proxy signature schemes. Proceedings of the Ninth National Conference on Information Security, 1999, pp. 241–246.
[28] H.M. Sun, N.Y. Lee, T. Hwang, Threshold proxy signature, IEEE Proc. Comput. Digital Tech. 146 (5) (1999) 259–263.
[29] V. Varadharajan, P. Allen, S. Black, An analysis of the proxy problem in distributed systems, Proceedings of the 1991 IEEE Computer Society Symposium on Research in Security and Privacy, 1991, pp. 255–275.
[30] C.Y. Yang, S.F. Tzeng, M.S. Hwang, On the efficiency of nonrepudiable threshold proxy signature scheme with known signers, J. Syst. Software 73 (3) (2004) 507–514.
[31] L. Yi, G. Bai, G. Xiao, Proxy multi-signature schemes: a new type of proxy signature scheme, Electron. Lett. 36 (6) (2000) 527–528.
[32] K. Zhang, Threshold proxy signature schemes, Information Security Workshop, 1997, pp. 191–197.