New Secure Proxy Signature Scheme with Fault Tolerance Based On Factoring and Discrete Logarithm

H. Elkamchouchi[1], Heba G. Mohamed[2], Fatma Ahmed[3] and Dalia H. ElKamchouchi[4]
[1]*Dept. of Electrical engineering, Faculty of Engineering, Alexandria University, Egypt*
helkamchouchi@ieee.org
[2]*Dept. of Electrical engineering, Arab Academy for Science and Technology (AAST), Egypt*
heba.g.mohamed@gmail.com
[3] *Dept. of Electrical engineering, Faculty of Engineering, Alexandria University, Egypt*
moonyally@yahoo.com
[4]*Dept. of Electrical engineering, Faculty of Engineering, Alexandria University, Egypt* Daliakamsh@yahoo.com

ABSTRACT: *Digital signature is an electronic signature form used by an original signer to sign a specific document. When the original signer is not in his office or when he/she travels outside, he/she delegates his signing capability to a proxy signer and then the proxy signer generates a signing message on behalf of the original signer.During the transmission of data between the sender and receiver, errors may occur frequently. Therefore, the sender must re-transmit the data to the receiver in order to correct these errors, which makes the system very feeble. The techniques of proxy signature and fault tolerance are two important issues in modern communication.To communicate securelyover an unreliable public network, the two parties must be able to authenticate one another and agree on a secret encryption key. Authenticated key agreement protocols have an important role in building a secure communications network between the two parties. In this paper, we propose a secure proxy signature scheme with fault tolerance over an efficient and secure authenticated key agreement protocol based on factoring and the discrete logarithm problem.*

KEYWORDS:*Discrete logarithm,Factoring,Fault tolerance, Key agreement,Proxy signature.*

I. INTRODUCTION

The cryptographic treatment of proxy signature scheme was first introduced by Mambo et al. in 1996 [1]. Proxy signature is an important inquiry in the field of a digital signature. It permits an original signer to delegate his signing rights to a proxy signer, and then the proxy signer performs the message signing on behalf of the original signer. For example, a director of a company wants to survive for a long trip. He would require a proxy agent, to whom he would delegate his signing capability, and thereafter the proxy agent would sign the documents on behalf of the director. The classification of the proxy signature is dependent on the basis of delegation, namely full delegation, partial delegation and delegation by warrant, and presents a well-organized strategy.

In full delegation, the proxy signer signs document using the same secret key of the original signer given by the original signer. The drawback of proxy signature with full delegation is the difficulty to distinct/differentiate between original signer and proxy signer. In partial delegation, the proxy key is derived from the secret key of the original signer and hands it over to the proxy signer as a delegation capability. Due to partial delegation cannot restrict the proxy signer’s signing capability, he/she can misuse the delegation capability. The weaknesses of full delegation and partial delegation are eliminated by partial delegation with warrant. A warrant explicitly states the signer’s identity, delegation period, and the qualification of messages on which the proxy signer can sign.

In 1997, Kim et al. [2] proposed a scheme using the concept of partial delegation with a warrant to restrict proxy signer signing capability. In 1999, Okamoto et al. [3], for the first time, proposed proxy unprotected signature scheme based on RSA scheme. A proxy-protected signature scheme based on the RSA assumption was proposed by Lee et al. in 2001 [4], [5]. In 2002, Shum and Wei [6] proposed another proxy protected signature scheme. Shao proposed the first proxy signature scheme based on the factoring integer problem in 2003 [7]. In 2005, Zhou et al. [8] proposed two efficient proxy-protected signature schemes. Their first system is based on RSA assumption and the second strategy was based on the integer factorization problem. Also, in 2005 Han et al. [9] introduced a relatively new proxy signature scheme which is as secure as ElGamal signature [10]. Next, a signature based on two hard problems factoring and discrete logarithms was introduced by Harn [11] and Li et al. [12]. For more security, in 2013, Mat-Isa and Ismailintroduced a new proxy signature with the revocation based on factoring and discrete logarithm problems [13].

Due to the rapid growth in modern communication systems, fault tolerance and data security are two important issues in a secure transaction. During the transmission of data between the sender and receiver, errors may occur frequently. Therefore, the sender must re-transmit the data to the receiver in order to correct these errors, which makes the system very feeble.Digital signature schemes with fault tolerance make it possible for error detections and corrections during the processes of data computations and transmissions. Previously, Zhang [14] and Lee and Tsai [15] have respectively proposed two efficient fault-tolerant schemes based on the RSA cryptosystem. Both of them can efficiently check the sender's identity and keep the confidentiality of the transmitted document. Furthermore, they can detect the errors and correct them. However, these schemes have a common weakness in security. Huifang Xue [16] has improved the mechanism of Lee and Tsai by providing extra security against Chosen Ciphertext Attacks (CCA) using a permutation matrix. If a malicious looks into the message he will find it difficult to understand or calculate checksum/ hash value due to the randomization of permutation matrix.

The two parties must authenticate mutually and agree on a secret encryption key to communicate together securely. To achieve this, key establishment protocols are applied at the beginning of a communication session in order to verify the parties' identities and build a common session key. Authenticated key agreement protocols have an important role in establishing secure communications between the two parties over the open network. The most famous protocol for key agreement was proposed by Diffie and Hellman, which is based on the concept of public-key cryptography (DL) [17]. There are two types of the Diffie-Hellman protocol namely static and ephemeral. In the first one, the parities exchange static public keys, and in the second, they exchange ephemeral public keys [18]. The important feature of the designed protocol is the established session key is formed as a combination of static and ephemeral private keys of two parties.

In this paper, we propose a secure proxy signature scheme over an efficient and secure authenticated key agreement protocol based on two hard problems; factoring and discrete logarithm problems. The designed protocol for authenticated key agreement is secure as well as efficient and provides authentication between two entities before exchanging the session keys. The remaining parts of this paper are organized as follows: In Section II, we elaborate security properties of the proxy signature scheme. Next, we discuss the designed protocol in Section III. In Section IV, we proposed our proxy signature scheme. We analyze the security properties and common attacks of our proposed scheme in Section V. Finally, in Section VI, we give our conclusion.

II. SECURITY REQUIREMENTS OF PROXY SIGNATURE

The security requirements for any proxy signature are first studied in [14] and later were improved in [1], [2]. According to them, a secure proxy signature scheme is expected to satisfy the following five requirements :

1. Verifiability: A verifier can be confident of the original signer's agreement on the signed message from a proxy signature
2. Strong unforgeability: Only the designated proxy signer can generate a valid proxy signature.
3. Strong identifiability: The identity of the proxy signer can be determined by any verifier from a proxy signature.
4. Strong undeniability: The proxy signer cannot repudiate the signature creation against anyone else, once he creates a valid proxy signature on behalf of an original signer.
5. Prevention of misuse: The responsibility of the proxy signer should be determined explicitly if he misuses the proxy key for the purposes other than generating a valid proxy signature.

III. NEW KEY AGREEMENT PROTOCOL

The used protocol for the authenticated key agreement [19] provides authentication between the two parties A and B before exchanging the session keys. The protocol consists of three phases; The Registration

Phase, The Transfer and Substantiation Phase, and The Key Generation Phase. Fig. 1 shows the overall operation of the new protocol.

$$\begin{array}{ccc}
\text{A} & & \text{B} \\
s_1 \equiv (y_B)^{-r_A} \cdot (t_B)^{x_A} & \overset{s_1}{\underset{s_2}{\rightleftarrows}} & .s_2 \equiv (y_A)^{-r_B} \cdot (t_A)^{x_B}\ . \\
v_2 = \left((t_B)^{x_A} \cdot s_2 \right)^{-\bar{r}_A} & & v_1 = \left((t_A)^{x_B} \cdot s_1 \right)^{-\bar{r}_B} \\
\equiv y_B & & \equiv y_A \\
K_{AB} \equiv v_2' \cdot t_B{}^{x_A} \cdot t_B{}^{r_A} & & K_{AB} \equiv v_1' \cdot t_A{}^{x_B} \cdot t_A{}^{r_B} \\
\equiv y_B{}^{r_A} \cdot t_B{}^{x_A} \cdot t_B{}^{r_A} & & \equiv y_A{}^{r_B} \cdot t_A{}^{x_B} \cdot t_A{}^{r_B} \\
K_{AB} \equiv g^{x_B r_A + x_A r_B + r_A r_B} & & K_{AB} \equiv g^{x_A r_B + x_B r_A + r_B r_A}
\end{array}$$

Fig.1. Overall operation of the proposed protocol

The system picks short-term private key r_A, r_B, they are random integers $2 \le r_A, r_B < n1$, and $GCD(r, n1) = 1$. $n1 = (p-1)(q-1)$ where p , q are large safe prime numbers normally at least 512 bits. t_A, t_B are short-term public keys where $t_A = g^{r_A} \ mod \ n$ and $t_B = g^{r_B} \ mod \ n$, g is a generator of Z_p^* and $n = pq$ long term public key at least 1024 bits. Then, the system picks long-term private keys x_A, x_B they are random integer where $2 \le x_A, x_B < n1$ and $GCD(x, n1) = 1$ and compute long-term public key y_A , y_B where $y_A = g^{x_B} \ mod \ n$ and $y_B = g^{x_A} \ mod \ n$. K_{AB} is the shared secret key calculated by the new secure protocol between the two parties A and B.

In the new protocol, there is only one message sent from one entity to another. The message is sent from A to B and vice versa from B to A , both have the same structure and independent of each other. The protocol has low communication overhead where, the total number of transmitted bits is $|n|$. The protocol has low complexity (complexity is 4) since the protocol needs only four exponential operations. So, it provides desirable performance attributes.

IV. IUONAND CHIN CHANG'S SCHEME

Iuon and Chin Chang's scheme [15] is developed from the concept of meta-ElGamal signature scheme [14] and the concept of Zhang's fault-tolerant signature scheme. In ElGamal digital signature scheme, a system first chooses a large prime *p* and a generator *g*, such that $g \in Z_p^*$ with order *p* − 1. Both *p* and *g* can be shared among a system of users. To generate a key pair, the signer *A* first chooses a random number x_A , $x_A \in Z_{p-1}$ and calculates $y_A = g^{x_A} \ mod \ p$. *A* keeps x_A secret and publishes y_A . Suppose that the signer Alice will send a message with her signature to the receiver Bob. Alice possesses a secret key x_A and a public key y_A . The proposed scheme can be divided into two procedures:

1. The signature generation procedure,
2. The fault tolerance and signature verification procedure.

4.1 The Signature Generation Procedure

1. Alice first divides the transmitted message *M*into numerical 3 × 3 message matrices X_l 's , such that

$$X_l = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \qquad (1)$$

Where m_{ij} , $1 \le i \le 3$, $1 \le j \le 3$, is a message block and $m_{ij} \in Z_{p-1}$

2. For each message matrix X_l, Alice calculates its signature and constructs an expand matrix D_l, such that

$$D_l = \begin{bmatrix} m_{11} & m_{12} & m_{13} & r_1 & s_1 & t_1 \\ m_{21} & m_{22} & m_{23} & r_2 & s_2 & t_2 \\ m_{31} & m_{32} & m_{33} & r_3 & s_3 & t_3 \\ r^1 & r^2 & r^3 & & & \\ s^1 & s^2 & s^3 & & & \\ t^1 & t^2 & t^3 & & & \end{bmatrix} \tag{2}$$

The r_i, s_i, t_i, r^i, s^i *and* t^i can be calculated by using the following equations

$$r_i = g^{k_i} \; mod \; p \, , \tag{3}$$

$$t_i = \sum_{j=1}^{3} m_{ij} \bmod p-1 \, , \tag{4}$$

$$s_i = (H(m_{i1}) \cdot t_i - H(m_{i2}) \cdot r_i \cdot x_A)(H(m_{i3}) \cdot k_i)^{-1} \bmod p-1 \, , \tag{5}$$

$$r^j = g^{k_j} \; mod \; p \, , \tag{6}$$

$$t^j = \sum_{i=1}^{3} m_{ij} \bmod p-1 \, , \tag{7}$$

$$s^j = (H(m_{1j}) \cdot t^j - H(m_{2j}) \cdot r^j \cdot x_A)(H(m_{3j}) \cdot k^j)^{-1} \bmod p-1 \, , \tag{8}$$

where $H()$ is a public one-way hash function.

4.2 The Fault Tolerance and Signature Verification Procedure

1. Bob first detects errors by checking the equations

$$t_i = \sum_{j=1}^{3} m_{ij} \bmod p \quad \text{and} \, t^j = \sum_{i=1}^{3} m_{ij} \bmod p \tag{9}$$

If there is an error in $m_{uv}, 1 \le u, v \le 3$, we must have that $t_u \ne \sum_{j=1}^{3} m_{uj} \bmod p-1$ and $t^v \ne \sum_{i=1}^{3} m_{iv} \bmod p-1$

Therefore, the error could beeasily detected.

2. After the error is detected in m_{uv}, it may be corrected by using either one of the following two equations

$$m_{uv} = t_u - \sum_{j \ne v} m_{uj} \bmod p$$

$$m_{uv} = t^v - \sum_{i \ne u} m_{iv} \bmod p \tag{10}$$

3. After correcting the errors, Bob has to verify the validity of the recovery and its corresponding signatures by checking whether

$$g^{H(m_{i1}) \cdot t_i} = y_A^{H(m_{i2}) \cdot r} \cdot r_i^{H(m_{i3}) \cdot s_i} \bmod p$$

$$g^{H(m_{1j}) \cdot t^j} = y_A^{H(m_{2j}) \cdot r^j} \cdot r^{j^{H(m_{3j}) \cdot s^j}} \bmod p \tag{11}$$

or not. If the above verifications are positive, Bob will believe that the contents of the recovered messages are valid. Otherwise, Bob can choose not to accept the receipted messages.

V. THE PROPOSED PROXY SIGNATURE SCHEME

The proposed proxy scheme is based on the new authenticated key agreement protocol with two hard problems factoring and discrete logarithm problems. The system is divided into four phases: System setup, Proxy key generation, Proxy key verification, Proxy signature generation and Proxy signature verification

5.1 System Setup

For the convenience of describing our work, we define the parameters as follows:

A:	Original signer
P:	Proxy signer
B:	Receiver
p,q :	Two large prime number
(e_A,d_A):	Secret key of original signer, $d_A = e_A^{-1} \bmod n_A$
(e_A,n_A):	Public key of original signer
$h(\,)$:	A secure one-way hash function.
K_{AP} :	Shared secret key between A and P
m_w :	A warrant
ID_A, ID_P :	Identity of A and P
G :	Subgroup of Z_p^* of order $p'q'$.
g :	Generator of G .
x_A, x_P :	Long-term private keys of A and P.
y_A, y_P :	Long-term public keys: $y_A \equiv g^{x_A} \bmod p$ and $y_P \equiv g^{x_P} \bmod p$.

5.2 Proxy Key Generation

1. The original signer entity *A*first divides the transmitted message *M* into numerical 3 × 3 message matrices and do the following:

- Selects an arbitrary integer value $k_i, k^j \in Z_{p-1}$
- Find $r_i = g^{k_i} \bmod p$ *and* $r^j = g^{k^j} \bmod p$
- Calculate warrant m_w where, m_w must be created from ID_A, ID_P and other data on the delegation.
- Compute $h(m_w \square r_i \square K_{AP})$ *and* $h(m_w \square r^j \square K_{AP})$
- Find $\sigma_i = k_i + x_A * h(m_w \square r_i \square K_{AP}) \bmod p-1$, $\sigma^j = k^j + x_A * h(m_w \square r^j \square K_{AP}) \bmod p-1$

for all $1 \le i \le 3, 1 \le j \le 3$.

- Compute $u_i = \sigma_i^{d_A} \bmod n_A$, $u^j = \sigma^{j^{d_A}} \bmod n_A$
- Send $(m_w, r_i, r^j, K_{AP}, u_i, u^j, \sigma_i, \sigma^j)$ to the proxy signer in the secure channel.

2.The proxy signer does the following:

- Shares a key d_A with original signer
- Checks the validity of $(m_w, r_i, r^j, K_{AP}, u_i, u^j, \sigma_i, \sigma^j)$ by verifying whether or not the following equation holds

$$g^{u_i^{e_A}} \equiv r_i\, y_A^{h(m_w \square r_i \square K_{AP})} \; \textit{and} \; g^{u^{j^{e_A}}} \equiv r^j\, y_A^{h(m_w \square r^j \square K_{AP})} . \quad (12)$$

If the verification is successful, the proxy signer then computes an alternative proxy private/public key pair σ_{pr} and y_{pr} , respectively, such that

$$\begin{aligned} \sigma_{i_{pr}} &= \sigma_i + x_P * h(m_w \square r_i \square K_{AP}) \bmod p-1 \\ y'_{i_{pr}} &= g^{\sigma_{pr}} \bmod p \end{aligned} \quad (13)$$

$$\begin{aligned} \sigma_{pr}^{j} &= \sigma^{j} + x_{P} * h(m_{w} \square r^{j} \square K_{AP}) \bmod p-1 \\ y_{pr}^{'j} &= g^{\sigma_{pr}^{j}} \bmod p \end{aligned} \qquad (14)$$

5.3 Signature Generation

Now, the proxy signer P will sign a message M on behalf of the original signer, he uses σ_{pr} to perform a signing operation. The proxy signature on the message M is as follows

$$s_{i} = (H(m_{i1}) \cdot t_{i} - H(m_{i2}) \cdot \sigma_{i} \cdot y_{i_{pr}}^{'})(H(m_{i3}) \cdot \sigma_{i_{pr}})^{-1} \bmod p-1,$$

$$s^{j} = (H(m_{1j}) \cdot t^{j} - H(m_{2j}) \cdot \sigma^{j} \cdot y_{pr}^{'j})(H(m_{3j}) \cdot \sigma_{pr}^{j})^{-1} \bmod p-1, \quad (15)$$

Where $t_{i} = \sum_{j=1}^{3} m_{ij} \bmod p-1$ and $t^{j} = \sum_{i=1}^{3} m_{ij} \bmod p-1$, For all $1 \le i \le 3, 1 \le j \le 3$

5.4 Fault tolerance and Signature Verification

1. The receiver B first detects errors by checking the equations

$$t_{i} = \sum_{j=1}^{3} m_{ij} \bmod p \quad \text{and} \quad t^{j} = \sum_{i=1}^{3} m_{ij} \bmod p \quad (16)$$

If there is an error in $m_{uv}, 1 \le u, v \le 3$, we must have that $t_{u} \ne \sum_{j=1}^{3} m_{uj} \bmod p-1$ and $t^{v} \ne \sum_{i=1}^{3} m_{iv} \bmod p-1$

Therefore, the error could be easily detected.

2. After the error is detected in m_{uv}, it may be corrected by using either one of the following two equations

$$m_{uv} = t_{u} - \sum_{j \ne v} m_{uj} \bmod p$$

$$m_{uv} = t^{v} - \sum_{i \ne u} m_{iv} \bmod p \quad (17)$$

3. The receiver Breceive the signed message and he has to check whether or not the following equations hold:

$$g^{H(m_{i1}) \cdot t_{i}} = y_{i_{pr}}^{H(m_{i3}) \cdot s_{i}} \cdot (r_{i} \cdot y_{A}^{h(m_{w} \cdot r_{i})})^{H(m_{i2}) \cdot y_{i_{pr}}} \bmod p$$

$$g^{H(m_{1j}) \cdot t^{j}} = y_{pr}^{j\, H(m_{3j}) \cdot s^{j}} \cdot (r^{j} \cdot y_{A}^{h(m_{w} \cdot r^{j})})^{H(m_{2j}) \cdot y_{pr}^{j}} \bmod p$$

$$y_{p\mathrm{r}}^{'} = r(y_{A} y_{P})^{h(m_{w} \square r \square K_{AP})} \mathrm{mod}p(18)$$

VI. SECURITY ANALYSIS

In the following, we show that the proposed schemes satisfy the security features, namely, verifiability, strong unforgeability, strong, undeniability, strong identifiability, and prevention of misuse.

6.1 Verifiability

According to the step 1 of the fault tolerance and the signature verification procedure, if an error occurs in m_{ij}, therefore $t_{i} \ne m_{i1} + m_{i2} + m_{i3} \bmod p$ and $t^{j} \ne m_{1j} + m_{2j} + m_{3j} \bmod p$. The fault message m_{ij} can be recovered by computing, if the rest of the messages m_{ik} 's where k = 1 *to* 3 and $k \ne j$, in the i^{th} row are correct. On the otherhand, if the rest of the messages m_{kj} 's, where k = 1 *to* 3 and $k \ne i$, in the j^{th} column are correct, the fault message m_{ij} also can be recovered by computing $m_{ij} = t^{j} - (\sum_{k=1 to 3, k \ne i} m_{kj}) \bmod p$ .Therefore, an error is correctable only when no other errors simultaneouslyoccur in the same row i and the same column j. In the proposed scheme, we can correctfour errors in a message matrix X at most. Figure 2 illustrates the correctableconditions when four errors simultaneously occur in a message matrix. Therefore,all the four errors can be corrected by using the check-sums in either the row orthe column direction.

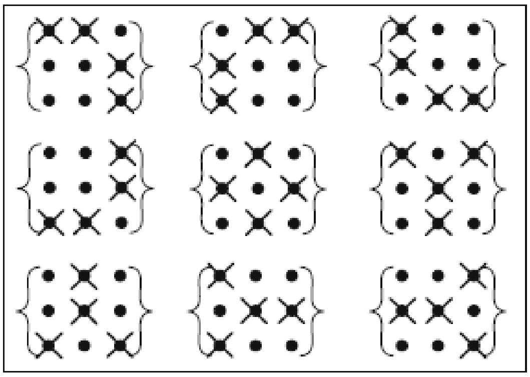

Fig. 2. The correctable conditions when there are four errors simultaneously occurring in a message matrix

According to the step 2, the receiver *B* can check the verification equation:

$$y'_{p\mathrm{r}} = g^{\sigma_{p\mathrm{r}}} \bmod p$$

$$= g^{\sigma + x_p * h(m_w \square r \square K_{AP})} \bmod p$$

$$= g^{\sigma} g^{x_p *(h(m_w \square r \square K_{AP})} \bmod p$$

$$= g^{k + x_A * h(m_w \square r \square K_{AP})} g^{x_p * h(m_w \square r \square K_{AP})} \bmod p$$

$$= g^{k} g^{x_A * h(m_w \square r \square K_{AP})} g^{x_p * h(m_w \square r \square K_{AP})} \bmod p$$

$$= g^{k} (g^{x_A} g^{x_p})^{h(m_w \square r \square K_{AP})} \bmod p$$

$$= r(y_A y_p)^{h(m_w \square r \square K_{AP})} \bmod p$$

6.2 Strong Unforgeability

In this scheme, the proxy signature is created with the proxy signer's secret key x_p and delegated proxy key σ. The proxy key is bound with the original signer's secret key x_A and the session key K_{AP}. No one (including the original signer) can construct the proxy signature. If the original signer tries to construct the proxy private key from a proxy public key, he/she will need to solve the discrete logarithm problem. However, the discrete logarithm problem is difficult. Moreover, from Equation (12) the verification of $h(m_w \square r_A \square K_{AP})$ with the signed message prevents the dishonest party from the creation of forged proxy signature. Therefore, any party, including the original signer cannot forge a valid proxy signature and thus the proposed scheme satisfies the unforgeability property.

6.3 Strong Identifiability

Any verifier can determine the identity of the proxy signer from the proxy signatures created by the proxy signer. Therefore, in the proposed scheme, any verifier can identify the identity of the proxy signer from the proxy signature generated by himon the message *M*.

6.4 Strong Undeniability:

In the proposed scheme, from Equations (13,14) the involvements of both original signer and proxy signer are determined by the secret keys x_p and d_A from the proxy signature. Thus, the proxy signer and the original signer cannot deny their involvement in a valid proxy signature. So, the scheme satisfies the undeniability property.

6.5 Prevention of Misuse

In the proposed scheme, the proxy signer cannot forge the delegated rights. The responsibility of the proxy signer is determined from the warrant m_w in the case of the proxy signer's misuse. Therefore, the original signer's misuse is also prevented because he cannot compute a valid proxy signature against the proxy signer.

Next, we show that our scheme is heuristically secured by considering the following five most common attacks.

Known-Key Security (K-KS): In the proposed scheme, if an established session key between original signer and proxy signer is disclosed, the adversary is unable to learn the other established session keys. In each run of the proposed scheme between the two parties, a unique session key which depends on r_A and r_P should be produced. Therefore, the adversary cannot compute K_{AP} and cannot calculate $\sigma = k + x_A * (h(m_w □ r □ K_{AP}) \bmod p - 1$.

(Perfect) Forward Secrecy: If both secret keys of two parties are compromised, the adversary is unable to derive old session keys, established by two parties. The protocol also possesses forward secrecy. Suppose that adversary compromises the private keys x_A, he/she cannot calculate $\sigma = k + x_A * h(m_w □ r □ K_{AP}) \bmod p - 1$. Moreover, the secrecy of previous session keys established by honest parties is not affected,because an adversary who captured the private key x_A should extract the ephemeral keys r_A or r_P from the exchanged values to know the previous or next session keys between them. However, this is DLP (Discrete Logarithm Problem). On the other hand, assume adversary is able to solve FAC problem that means he/she knows the prime factorization of n_A and can compute d_A; however, he/she cannot compute $\sigma = k + x_A * h(m_w □ r □ K_{AP}) \bmod p - 1$since no information is available for x_A. Thus, he/she still fails to produce σ_A send to proxy signer.

Key-Compromise Impersonation (K-CI): When the private key of original signer is compromised, it may be desirable that this event does not enable an adversary to impersonate the other entities to A. Suppose that's long-term private key x_A, is disclosed. Now, an opponent who knows this value can clearly impersonate A. In the proposed scheme, the opponent cannot impersonate P to A and compute $\sigma_{pr} = \sigma + x_P * h(m_w □ r □ K_{AP}) \bmod p - 1$ without knowing the$P$'s long-term private key x_P. From the success of the impersonation, the opponent must know A's ephemeral key r_A. So, in this case, the opponent should extract the value r_A from $t_A \equiv g^{r_A} \bmod n$; however, he/she cannot calculate the sharing key, and this is DLP. Furthermore, he cannot compute $u = \sigma^{d_A} \bmod n_A$ which is the RSA

Unknown Key-Share (UK-S): The original signer A cannot be coerced into sharing a key with the proxy signer P without the knowledge of the original signer, i.e., A believes that the key is shared with some entity $C \neq P$, and P believes that the key is shared with A. The used protocol prevents unknown key-share. Corresponding to the proxy signer's public static and ephemeral keys y_P, t_P, an adversary cannot register proxy signer's public keys y_P, t_P as its own, and according to the assumption of this protocol that s_2 has verified that P possesses the private static and ephemeral keys x_P, r_P, respectively. So an adversary cannot deceive the original assuming that $\sigma_{pr} = \sigma + x_P * h(m_w □ r □ K_{AP}) \bmod p - 1$was originated from him. Therefore, the original signer cannot be coerced into sharing K_{AP} with the proxy signer without his/her knowledge.

VII. CONCLUSION

In this paper, we proposed a new secure proxy signature with fault tolerance and a new key agreement protocol based on factoring and discrete logarithms. Our scheme does not consider the proxy revocation mechanism. The scheme provides a higher level of security than a single hard problem is based on two hard problems. Furthermore, it satisfies the capability of correcting four at most errors for each 3×3 message matrix.On the other hand, the scheme satisfies the necessary security requirements of proxy signature and has a secure channel to deliver the proxy key, through the designed new protocol that meets the security attributes under the assumption of DLP and RSA.

REFERENCES

[1] M. Mambo, K. Usuda, E. Okamoto, Proxy signature: delegation of the power to sign the message, IEICE Trans. Fundamentals E79-A (9) (1996) PP. 1338 - 1353.

[2] S. Kim, S. Park and D. Won, "Proxy signatures", In: ICICS97, LNCS 1334, Springer-Verlag, (1997), pp. 223-232.

[3] T. Okamoto, M. Tada and E. Okamoto, "Extended proxy signatures for smart card", In: Proceedings of Information Security Workshop 99, LNCS 1729, Springer-Verlag, (1999), pp. 247-258.

[4] B. Lee, H. Kim and K. Kim, "Secure mobile agent using strong non-designated proxy signature", In: Information security and private (ACISP01), LNCS 2119, Springer-Verlag, (2001), pp. 474-486.

[5] B. Lee, H. Kim and K. Kim, "Strong proxy signature and its applications", In: Proceeding of the 2001 symposium on cryptography and information security (SCIS01), vol. 2, no. 2, (2001), pp. 603-608.

[6] K. Shum and V. K. Wei, "A strong proxy signature scheme with proxy signer privacy protection", In: Proceedings of IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE02), (2002).

[7] Z. Shao, "Proxy signature schemes based on factoring", Inform Process Lett., no. 85, (2003), pp. 137 143.

[8] Y. Zhou, Z. Cao and R. Lu, "Provably secure proxy-protected signature schemes based on factoring", Appl Math Comput., vol. 164, no. 1, (2005), pp. 83-98

[9] S.Han, E. Chang, J.Wang, W.Liu, A New Proxy Signature Scheme As Secure As Elgamal Signature, World Academy of Science, Engineering and Technology, 11(2005), 27-31.

[10] T. Elgamal, A Public Key Cryptosystem and Signature Scheme Based On Discrete Logarithms, IEEE Trans. Information Theory, 1985, 469-472..

[11] L. Li, S. Tzeng, M. Hwang, Improvement of signature based on factoring and discrete logarithms, Applied Mathematics and Computation, 161(2005), 49-54.

[12] L. Harn, Public Key Cryptosystem Design Based on Factoring and Discrete logarithms, ZEE Proceeding Computer Digit Tech 141(3), 193-195.

[13] M. Mat-Isa, E. S. Ismail," A new proxy signature with revocation based on factoring and discrete logarithm ", Applied Mathematical Sciences, Vol. 7, 2013, no. 123, 6141-6152

[14] C.N. Zhang, "Integrated Approach for Fault Tolerance and Digital Signature in RSA," IEEE Proceedings-Computers & Digital Techniques, vol. 146, no. 3, pp. 151-159, 1999

[15] N. Lee and W. Tsai, "Efficient Fault-tolerant Scheme basde on the RSA system," IEEE Proceedings – Computer and Digital Techniques, vol. 150, no. 1, pp. 17-20, 2003.

[16] Xue, H. (2010) Improving the Fault-Tolerant Scheme Based on the RSA System. International Symposium on Computational Intelligence and Design, Hangzhou, 29-31 October 2010, 31-33.

[17] W. Diffie and M. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, vol. IT-l 22, no. 6, PP. 644-654, November, 1976.

[18] K. Chalkias, F. Mpaldimtsi, D. H. Varsakelis, and G. Stephanides, "On the Key-compromise impersonation vulnerability of one-pass key establishment protocols," in Proc. International Conference on Security and Cryptography (SECRYPT 2007), Barcelona, Spain, July 28-31, 2007

[19] H. Elkamchouchi, M. R. M. Rizk, and Fatma Ahmed," A New Secure Protocol for Authenticated Key Agreement," IACSIT International Journal of Engineering and Technology, Vol. 5, No. 2, April 2013,pp.245B. Lee, H. Kim and K. Kim, "Secure mobile agent using strong non-designated proxy signature", In: Information security and private (ACISP01), LNCS 2119, Springer-Verlag, (2001), pp. 474-486.