Bulletin of Electrical Engineering and Informatics
Vol. 4, No. 1, March 2015, pp. 73~82
ISSN: 2089-3191

Practical IBC using Hybrid-Mode Problems: Factoring and Discrete Logarithm

Chandrashekhar Meshram
Department of Applied Mathematics, Gyan Ganga Institute of Technology and Sciences,
Jabalpur (M.P.), India
e-mail: cs_meshram@rediffmail.com

Abstract

Shamir proposed the concept of the ID-based cryptosystem (IBC) in 1984. Instead of generating and publishing a public key for each user, the ID-based scheme permits each user to choose his name or network address as his public key. This is advantageous to public-key cryptosystems because the public-key verification is so easy and direct. In such a way, a large public key file is not required. Since new cryptographic schemes always face security challenges and many integer factorization problem and discrete logarithm based cryptographic systems have been deployed, therefore, the purpose of this paper is to design practical IBC using hybrid mode problems factoring and discrete logarithm. We consider the security against a conspiracy of some entities in the proposed system and show the possibility of establishing a more secure system.

Keywords*: Cryptography, Public key Cryptosystem (PKC), ID-based Cryptosystem (IBC), Discrete Logarithm (DL) and Integer Factorization (IF)*

1. Introduction

Rapid advances in computer technology and the development of the Internet are changing the way, we conduct our daily and business lives. Secrecy is an important issue with respect to sensitive data transferred over insecure public channels. In an open network environment, secret session key needs to be shared between two users before it establishes a secret communication [1-3]. While the number of users in the network is increasing, key distribution will become a serious problem. In 1976, Diffie and Hellman [4] introduced the concept of the public key distribution system (PKDS). In the PKDS, each user needs to select a secret key and compute a corresponding public key and store in the public directory. The common secrete session key, which will be shared between two users can then be determined by either user, based on his own secret key and the partner's public key. Although the PKDS provides an elegant way to solve the key distribution problem, the major concern is the authentication of the public keys used in the cryptographic algorithm.

Many attempts have been made to deal with the public key authentication issue. Kohnfelder [5] used the RSA digital signature scheme to provide public key certification. His system involves two kinds of public key cryptography: one is in modulo p, where p is a large prime number; the other is in modulo N, where N = pq, and p and q are large primes [6]. Blom [7] proposed a symmetric key generation system (SKGS) based on secret sharing schemes. The problems of SKGS however, are the difficulty of choosing a suitable threshold value and the requirement of large memory space for storing the secret shadow of each user [8-14].

In 1984, Shamir [1] introduced the concept of an IBC. In this system, each user needs to visit key authentication center (KAC) and identify himself before joining the network. Once a user's identity is accepted, the KAC will provide him with a secret key. In this way, a user needs only to know the "identity" of his communication partner and the public key of the KAC, together with his secret key, to communicate with others [15-20]. There is no public file required in this system. However, Shamir did not succeed in constructing an IBC, but only in constructing an IBS scheme [21-22]. Since then, much research has been devoted, especially in Japan, to various kinds of IBC schemes. Okamoto et al. [6] proposed an identity-based key distribution system in 1988, and later, Ohta [10] extended their scheme for user identification. These schemes use the RSA public key cryptosystem [14] for operations in modular N, where N is a

Received October 14, 2014; Revised December 20, 2014; Accepted January 10, 2015

product of two large primes, and the security of these schemes is based on the computational difficulty of factoring this large composite number N. Tsujii and Itoh [2] have also proposed an IBC based on the discrete log with single discrete exponent which uses the ElGamal public key cryptosystem.

In 1991, Maurer and Yacobi [23] developed a non-interactive ID-based public-key distribution system. In their scheme, the public keys are self-authenticated and require no further authentication by certificates. However, some problems with this scheme were found, the scheme was modified and the final version was presented [24]. In 1998, Tseng and Jan [25] improved the scheme proposed by Maurer and Yacobi, and provided a non-interactive ID-based public-key distribution system with multi-objectives such as an ID-based signature scheme, an identification scheme, and a conference key distribution system. In their scheme, the computational complexity of the system is heavy. Therefore, it is necessary to have a powerful computational capability. Harn [13] proposed public key cryptosystem design based on factoring and discrete logarithm whose security is based factoring and discrete logarithm. In 2001, Boneh and Franklin, Cocks [26] used a variant of integer factorization to construct his ID-based encryption scheme. However, the scheme is inefficient in that a plain-text message is encrypted bit-by-bit and hence the length of the output ciphertext becomes long.

In 2004, Lee & Liao [8] design a transformation process that can transfer all of the discrete logarithm based cryptosystems into the ID-based systems rather than reinvent a new system. After 2004 several IBC [9, 15, 19, 20, 21, 22, 27] have been proposed. But in these schemes, the public key of each entity is not only an identity, but also some random number selected either by the entity or by the trusted authority. In 2009, Bellare et al. [11] provides security proof or attacks for a large number of ID-based identification and signature schemes. Underlying these is a framework that on the one hand helps explain how these schemes are derived and on the other hand enables modular security analyses, thereby helping to understand, simplify, and unify previous work. In 2010, Meshram [16] has also proposed cryptosystem based on double generalized discrete logarithm problem whose security is based on double generalized discrete logarithm problem with distinct discrete exponents in the multiplicative group of finite fields. After some time Meshram presented the modification of IBC based on the double discrete logarithm problem [17, 18, 28-30] and also proposed an Identity based beta cryptosystem, whose security is based on generalized discrete logarithm problem and integer factorization problem [31]

Based on the observation that new cryptographic schemes always face security challenges and confidentiality concerns and many integer factorization & discrete logarithm-based cryptographic systems have been deployed. The major contribution of our scheme is the key generation phase, which is just a simple transformation process with low computational complexity. No modification of the original design of the discrete logarithm and integer factorization based cryptosystems is necessary [32]. Therefore, the new scheme has the same security as the original one, and retains all of the advantages of the ID-based system.

In this paper, we design IBC for discrete logarithm problem with distinct discrete exponent and integer factorization (the basic idea of the proposed system comes on the public key cryptosystem based on discrete logarithm problem and integer factorization) because we face the problem of solving integer factorization and distinct discrete logarithm simultaneously in the multiplicative group of finite fields as compared to the other public key cryptosystem, where we face the difficulty of solving simultaneously the integer factoring and discrete logarithm problem in the common group. Here we describe further considerations such as the security of the system, the identification for senders. etc. our scheme does not require any interactive preliminary communications in each message transmission and any assumption except the intractability of the discrete logarithm problem and integer factorization problem.(this assumption seems to be quite reasonable) Thus the proposed scheme is a concrete example of an IBC which satisfies Shamir's original concept [1] in a strict sense.

The remainder of this paper is organized as follows: Section 2 presented proposed public key encryption based on factoring and discrete log. Section 3 given supporting example for the scheme. Section 4 explains consistency of the algorithm. Section 5 describes implementation of the IBC. Section 6 describes system initialization parameters. Section 7 describes protocol of the proposed IBC. Section 8 discussed security of the IBC. Conclusion is given in the final section.

2. The public Key Encryption Based on Factoring and Discrete Log

2.1 Some Notations and Parameters:

Throughout the paper, we use the following notations and parameters:

- Two large strong random primes [14] p and q which are safe primes and set the modulus $N = p * q$.
- A function $\varphi(N) = (p-1)(q-1)$ is a phi-Euler function and $gcd\,(a,b)$ is the greatest common divisor of a and b
- g is a primitive element in $Z_N^* = \{z, \gcd(z,N) = 1\}$ with order N satisfying $g^{N-1} \equiv 1(mod\ N)$.
- $h(.)$ is a cryptographic hash function [32] whose output is a t-bit length and we assume here that $t = 128$.

2.2 Materials and Methods

We present a public key cryptosystem based on hybrid-mode problems; factoring and discrete logarithms. The scheme is described in three phase's namely key generation, encryption and decryption. In key generation phase, the public and secret keys of users are calculated. Once computed, the public keys will be published in public directory so that anyone including the adversaries could access it while the secret keys remain secret except the owners. In encryption phase, the original message that to be sent is first hashed using the appropriate cryptographic hash function $h(.)$. This function determines a fixed length of output by hashing any arbitrarily length of input. Then a sender gets his hashed message encrypted. This is done by using the receiver's public key and sender's commitment of secret number. The encrypted message is then sent to the legal receiver. In decryption phase, the receiver recovers the original message by using his own secret keys and without these secret keys no one can read the original message.

2.3 Key Generation

1. Pick randomly two integers $e, x < N$ from Z_N^* such that $gcd(e,N) = 1$
2. Use the extended Euclidean algorithm to compute the unique integer d, $1 \leq d \leq \varphi(N)$ such that $ed \equiv 1\ (mod\ \varphi(N))$.
3. Compute the number $y \equiv g^x\ mod\ N$.

The public key is formed by (e,y) and can be accessed in the public directory and the secret keys is given by (d,x) and only known to the legal receiver.

2.4 Encryption

The sender encrypts his message $h(m)$ as follows:

1. Select at random an integer $c < n$ from Z_N^*
2. Get the original message hashed and assume that the resultant becomes $h(m)$
3. Calculate the number:

$$C_1 \equiv g^c (mod\ N)$$

4. Disguise the message by computing $C_2 \equiv (h(m)y^{-c})^e (mod\ N)$

In the original ElGamal cryptosystem [3], we compute the cipher text C_2 in point (3) without the exponent e. In our scheme, we need this exponent to disguise our message 'twice' and to realize the hybrid-mode problems-based cryptosystem.

2.5 Decryption

The receiver decrypts the obtained encrypted message (C_1, C_2) as below:
Compute the following:

$$C_2^d C_1^x \equiv h(m) (mod\ N) \qquad \text{(A)}$$

3. Example

For purpose of validation, we illustrate an example to show the basic principle of our developed cryptosystem. Practitioners are not recommended to choose keys or parameters computed in this example in practice since inappropriate parameters would make this scheme vulnerable to attacks.

Assume that $p = 29, q = 43$. Then the modulus and its Euler-function are now given by $N = 1247$ and $\varphi(N) = 1176$. Next chooses the number $e = 11, x = 19$ and $g = 17$. Thus our public and secret keys of the scheme are $(11, 1143)$ and $(107, 19)$ respectively. To encrypt the message $h(m) = 1122$, the sender selects $c = 3$ and computes and sends receiver:

$$C_1 \equiv 17^3 (mod\ 1247) = 1172 \text{ and } C_2 \equiv (1122 \times 1143^{-3})^{11} (mod\ 1247) = 322$$

The receiver recovers the original message as below:

$$C_2^d C_1^x \equiv (\ 322^{107} \times 1172^{19}\ ) (mod\ 1247) = 1122$$

4. Consistency of the Algorithm

Theorem 1: If the algorithms of key generation and encryption run smoothly then the decryption of the encrypted message in decryption is correct.

Prof: The Eq. (A) above is true for all encrypted message (C_1, C_2) since:

$$\begin{aligned} C_2^d C_1^x &\equiv [(h(m) y^{-c})^e]^d (g^c)^x\ (mod\ N) \\ &\equiv (h(m) y^{-c})(g^{cx})\ (mod\ N) \\ &\equiv (h(m) g^{-cx})(g^{cx})\ (mod\ N) \\ &\equiv h(m) (mod\ N) \end{aligned}$$

5. Implementation of the IBC

5.1. Preparation for the center and each entity

Step 1. Each entity generates a k-dimensional binary vector for his ID. We denote entity $i's$ ID by ID_i as follows:

$$\mathrm{ID}_i = (x_{i1}, x_{i2}, x_{i3}, x_{i4}, \dots\dots\dots, x_{ik})\ , x_{ij} \in \{0,1\}\ , (1 \le j \le k) \quad (1)$$

Each entity registers his ID with the center, and the center stores it in a public file.

Step 2.: The center generates two random prime numbers p and q ,compute

$$N = pq \quad (2)$$

Then the center chooses an arbitrary random number $e, 1 \le e \le \varphi(N)$ such that gcd $\big(e, \varphi(N)\big) = 1$ where $\varphi(N) = (p-1)(q-1)$ is the Euler function of N, then the center publishes (e, N) as the public key. Any entity can compute the entity i's extended ID , $\mathrm{EID}_i$by the following:

$$\mathrm{EID}_i = (\mathrm{ID}_i)^e (mod\ N)\ = (y_{i1}, y_{i2}, y_{i3}, y_{i4}, \dots\dots\dots, y_{it})\ , y_{ij} \in \{0,1\}\ ,\ (1 \le j \le t) \quad (3)$$

where $t = |N|$ is the number of bits of N.

Step 3. Center's secrete information: The center chooses an arbitrary large prime p and q computes $N = pq$ and also generate n-dimensional vector $\vec{a}$ over $Z_{\varphi(N)}^*$ which satisfies

$$\vec{a} = (a_1, a_2, a_3, \dots\dots, a_n)\ , \quad 1 \le a_i \le \varphi(N),\ \ (1 \le i \le n) \quad (4)$$

$$aI \neq aJ \left(mod\ \varphi(N)\right)\ ,\ I \neq J \tag{5}$$

where I and J are n-dimensional binary vector and stores it as the centers secret information. The condition of equation (5) is necessary to avoid the accidental coincidence of some entities secrete keys. A simple way to generate the vector $\vec{a}$ is to use the Merkle and Hellman scheme [12].
The center chooses a super-increasing sequences corresponding to a as $a_i'\ (1 \leq i \leq n)$ satisfies

$$\sum_{1 \leq i \leq n} a_i' < \varphi(N) \tag{6}$$

Step 4: The center also chooses w such that $\gcd(w, \varphi(N)) = 1$, and computes n-dimensional vector $\vec{a}$ as follows

$$a_i = a_i' w (mod\ \varphi(N))(1 \leq i \leq n) \tag{7}$$

where

$$\vec{a} = (a_1, a_2, a_3, \dots, a_n) \tag{8}$$

Remark 1: It is clear that the vector $\vec{a}$ defined by Eq. (8) satisfies the Eqs. (4)-(5) the above scheme is one method of generating n vectors $\vec{a}$ satisfies Eqs. (4)-(5). However, another method might be possible.

Step 5: The center also chooses a unique integer d, $(1 \leq d \leq \varphi(N))$ such that

$$ed \equiv 1\ (mod\ \varphi(N)) \tag{9}$$

Step 6: Center public information*:* The center chooses an arbitrary generator g of $Z_{\varphi(N)}^*$ and computes n-dimensional vector h using generator g corresponding to the vector.

$$h = (h_1, h_2, h_3, \dots \dots, h_n) \tag{10}$$

$$h_i = g^{a_i} mod\ N\ (1 \leq i \leq n) \tag{11}$$

The center informs each entity (N, e, g, h) as public information.

Step 7: Each entity secrete key*:* Entity $i's$ secrete keys s_i is computed by inner product of a (the centre's secret information) and EID_i (entity $i's$ extended ID, see Eq.3)

$$\begin{aligned} s_i &= a\ \mathrm{EID}_i (mod\ \varphi(N)) \\ &= \sum_{1 \leq j \leq n} a_j y_{ij}\ mod\ (\varphi(N)) \end{aligned} \tag{12}$$

6. System Initialization Parameters

6.1 Center Secrete information

a : n -dimensional vector, d - is an integer {see Eqs. (8)-(9)}

6.2 Center public information

h : n -dimensional vector see Eqs. (10-11)} , N: large prime numbers, e: random integers , g : generator of $Z_{\varphi(N)}^*$.

6.3 Entity $i's$ secretes keys : (s_i) {see Eq. (12)}

6.4 Entity $i's$ public information: ID_i is a k −dimensional vector {see Eq. (1)}

7. Protocol of the Proposed IBC

Without loss of generality, we suppose that entity 2 sends message M to entity 1.

7.1 Encryption

Entity 2 generates EID_1 (entity $1's$ extended ID, see Eq.3) from ID_1. It then computes γ_1 from corresponding public information h and EID_1 :

$$\begin{aligned}\gamma_1 &= \prod_{1\le i\le n} {h_i}^{y_{1i}}(mod\ N)\\ &= \prod_{1\le i\le n} (g^{a_i})^{y_{1i}}(mod\ N)\\ &= g^{\sum_{1\le i\le n} a_i y_{1i} mod\,(\varphi(N))}(mod\ N)\\ &= g^{s_1}(mod\ N)\end{aligned} \quad (13)$$

Entity 2 will use γ_1 in our propose scheme. Let $h(m)$ be a message to be transmitted. Entity 2 is select a random integer $r < N$ and computes the cipher text C as follows

$$C = (C_1, C_2)$$

$$C_1 \equiv g^r(mod\ N) \quad (14)$$

$$C_2 \equiv (h(m)y^{-r})^e(mod\ N) \quad (15)$$

The cipher text is given by $C = (C_1, C_2)$

7.2 Decryption

To recover the plaintext $h(m)$ from the cipher text
Entity 1 does the following:
Computes

$${C_1}^{s_1}\ (mod\ N) \equiv\ (g^r)^{s_1}\ (mod\ N) \quad (16)$$

Using his secrete key s_1, recovered entity 2's the message $h(m)$ by Eqs. (13) and (16) to computes

$$\begin{aligned}\left({C_1}^{s_1}\ {C_2}^{d}\right) &\equiv g^{r s_1}[(h(m)y^{-r})^e]^d(mod\ N)\\ &\equiv (h(m)y^{-r})g^{r s_1}(mod\ N)\\ &\equiv (h(m)g^{-r s_1})g^{r s_1}(mod\ N)\\ &\equiv h(m)(mod\ N)\end{aligned}$$

8. Security Analysis

In this section, we shall show six possible attacks by which an attacker may try to take down the new encryption scheme. For each attack, we define the attack and give reason why this attack could be failed.

The security of ID-based cryptosystem based on the index problem in the multiplicative cyclic group $Z^*_{\varphi(N)}$, where $N = pq$ (The factorization of *N* is known only to the center.) where $\varphi(N)$ Euler function of N. In this system Coppersmith showed an attacking method [27] such that $(n+1)$ entities conspiracy can derive the center's secret information.

Attack 1 [27]: The $(n+1)$ entities $i, (1 \le i \le n+1)$ can derive an n-dimensional vector *a'* over $Z^*_{\varphi(N)}$ which is equivalent (not necessarily identical) to the original center's secret information.
Proof: When $(n+1)$ entities' $i, (1 \le i \le n+1)$ conspire, they have the following system of linear congruences:

$$\begin{bmatrix} EID_1 \\ EID_2 \\ EID_3 \\ \vdots \\ EID_{n+1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ \vdots \\ s_{n+1} \end{bmatrix} (mod\ \varphi(N)) \quad (17)$$

Since each EID_i is an n-dimensional binary vector, there exists an $(n+1)$-dimensional vector c over the integer ring such that

$$\sum_{1 \le i \le n+1} c_i EID_i = 0 \quad (18)$$

Thus we have

$$\sum_{1 \le i \le n+1} c_i s_i = 0 (mod\ \varphi(N)) \quad (19)$$

And then

$$\sum_{1 \le i \le n+1} c_i s_i = A\ \varphi(N) \quad (20)$$

If $\neq 0$, the $(n+1)$entities can have an integer multiple of $\varphi(N)$, and they can find out the factorization of N.Then, a similar method with attack 1 is applicable. Hence, the center's secret information can be derived by $(n+1)$-entities conspiracy.

Furthermore, Shamir developed a more general attacking method [28] for the modified system such that $(n+2)$ entities conspiracy can derive the center's secret information with high probability.

Attack 2[28]: The $(n+2)$ entities i , $(1 \le i \le n+2)$ can derive the center's secret information a with high probability.

Proof: When $(n+1)$ entities i , $(1 \le i \le n+1)$ conspire, they have the following system of linear congruence's defied by Eq.(21)

$$\begin{bmatrix} EID_1 \\ EID_2 \\ EID_3 \\ \vdots \\ EID_{n+1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ \vdots \\ s_{n+1} \end{bmatrix} (mod\ \varphi(N)) \quad (21)$$

$$=Da(mod\ \varphi(N)) \quad (22)$$

Assuming that the matrix D includes n linearly independent column vectors over the integer ring, there exist some positive integers c_i $(1 \le i \le n+1)$ such that

$$\begin{bmatrix} EID_1 \\ EID_2 \\ EID_3 \\ \vdots \\ EID_{n+1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_{n+1} \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ \vdots \\ s_{n+1} \end{bmatrix} - \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{n+1} \end{bmatrix} \varphi(N) \quad (23)$$

Thus, Eq. (23) can be rewritten by the following:

$$\begin{bmatrix} EID_1 \\ EID_2 \\ EID_3 \\ \vdots \\ EID_{n+1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \\ -1 \end{bmatrix} = - \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{n+1} \end{bmatrix} \varphi(N) \quad (24)$$

$$= D'a' \quad (25)$$

From the assumption that the matrix D in Eq. (22) includes n linearly independent column vectors over the integer ring, it follows that the matrix D' is nonsingular over the integer ring (i.e., det $(D') \neq 0$) with overwhelming probability, and thus, we have $a' \neq (mod\ \varphi(N))$. On the other hand, we have the following system of linear congruence's:

$$D'a' = 0(mod\ \varphi(N)) \tag{26}$$

If the matrix D' is nonsingular over $Z^*_{\varphi(N)}$, then $a' = (mod\ \varphi(N))$, and this contradicts the above results. Thus, the matrix D' is singular over $Z^*_{\varphi(N)}$,and we have det $(D') = 0(mod\ \varphi(N))$ with high probability. Hence, det (D') is divisible by $\varphi(N)$ with high probability. Furthermore, consider the case where the other $(n+1)$ entities among $(n+2)$ conspire, and define the matrix D'' in a way similar to the above. Then, det (D'') is divisible by $\varphi(N)$ with high probability. Hence, GCD (det(D'),det (D'')) gives $e\varphi(N)$ where *e* is a small positive integer. By the above procedure, we can evaluate $\varphi(N)$ efficiently. An additional procedure to find the center's secret information is completely the same as attack 1.

Attack 3: An attacker wishes to obtain all secrete keys using all information available from the system. In this case, attacker needs to solve integer factorization problem and discrete logarithm problem simultaneously. The best way to factorize $N = pq$ is by using the number field sieve method (NFS) [30].but this method is just dependent on the size of modulus N.It is computationally infeasible to factor a 1024-bit integer and to increase the security of our scheme; we should select strong primes [14] to avid attacks using special purpose factorization algorithms. To maintain the same security level for discrete logarithm problem with double distinct discrete exponent, one must uses $N = pq$ with $\left(\frac{p-1}{2}\right)$ and $\left(\frac{q-1}{2}\right)$ respectively is product of two 512-bit primes.

Attack 4: Assume that the attacker successfully solves the factoring problem so that he knows secrete key d.Thus he may obtain

$$\begin{aligned} {C_2}^d &\equiv (h(m)y^{-r})^{ed}(mod\ N) \\ &\equiv h(m)g^{-rs_1}(mod\ N) \end{aligned}$$

Unfortunately, at this stage he still does not knows secrete s_a and cannot exact the plaintext $h(m)$ from the above expression.

Attack 5: An attacker is able to obtain the secrete integer s_1 from $\gamma_1 \equiv g^{s_1}(mod\ N)$. He could derive the plaintext $h(m)$ if and only if he manages to get $h(m)y^{-r}$,but this is impossible since he learns nothing about the integer d.

Attack 6: An attacker might try to impersonate entity 1 by developing some relation between w and w' since $\gamma_1 = Y^{ws_1}(mod\ N)$ and $\gamma_1' = Y^{w's_1}(mod\ N)$ by knowing γ_1, w, w' the attacker can derive γ_1' as $\gamma_1' = \gamma_1^{w^{-1}w'}(mod\ N)$ without knowing s_i however trying to obtain w from g is equivalent to compute the discrete logarithm problem.

9. Enhancement of Security and Processing Cost

The center's secret information for the original system in Section 7 is derived by n entities conspiracy. In this subsection, we consider the practical countermeasure for the enhancement of the security of the system. (For simplicity, assume that n = 512 throughout this subsection.) The center partitions a 512-dimensional binary vector B into 256 segments, every two bits, such as

$$\begin{aligned} \mathrm{B} &= (\mathrm{b}_1, \mathrm{b}_2, \mathrm{b}_3, \dots\dots \mathrm{b}_{511}, \mathrm{b}_{512}) \\ &= (\mathrm{seg}_1, \mathrm{seg}_2, \mathrm{seg}_3, \dots\dots \mathrm{seg}_{511}, \mathrm{seg}_{512}) \end{aligned} \tag{27}$$

Then, the center defines $\mathrm{a(i;jk)}\ (1 \leq \mathrm{i} \leq 256; \mathrm{j,k} \in \{0,1\})$ appropriately, computes $\mathrm{h(i;jk)}, (1 \leq \mathrm{i} \leq 256; \mathrm{j,k} \in 0,1,$

$$\mathrm{h(i;jk)} = \mathrm{g}^{\mathrm{a(i;jk)}} (\mathrm{mod\ N}) \quad (28)$$

for each $\mathrm{seg_i}$, and publishes the table including every $\mathrm{h(i;jk)}$ to all entities. Furthermore, the center computes each entity's secret key s_{k} by

$$s_{\mathrm{k}} = \sum_{1 \le \mathrm{i} \le 256} a(i; \mathrm{seg_{ki}})(mod\ \varphi(N)) \quad (29)$$

Depending on Eq.(12).The entity k's extended identity, $\mathrm{EID_k}$, where $\mathrm{EID_k}$ is partitioned into 256 segments,every two bits such as $EID_k = (\mathrm{seg_{k1}}, \mathrm{seg_{k2}}, \mathrm{seg_{k3}}, \ldots \ldots \mathrm{seg_{k255}}, \mathrm{seg_{k256}})$ the center distributes it to each entity through a highly secure channel.

9.1 Encryption

Entity 2 computes γ_1' ,

$$\gamma_1' = \prod_{1 \le \mathrm{i} \le 256} \mathrm{h(i;seg_{1i})} \ (\mathrm{mod\ N}) \quad (30)$$

from $\mathrm{EID_1}$ and the published table. Entity 2 uses γ_1' as γ_1 in the original system (in Section 7) to encrypt the message M.

9.2 Decryption

This is exactly the same as in the original system in Section 6. In the original system in Section 6, the center's secret information is derived by 512 entities conspiracy, while in the above system it is derived by 1024 (= 4 x 256) entities conspiracy. Furthermore, the running cost for encryption-key generation in the above system is about half of the original system. However, the center's public information in the above system is about twice than the original system. Further generalizations, e.g., each $\mathrm{EID_i}$ is partitioned into 128 segments every four bits, etc., are possible.

10. Conclusion

In this present paper an IBC for integer factorization and discrete logarithm in the multiplicative group of finite fields. The proposed scheme satisfies Shamir's original concepts in a strict sense, i.e. it does not require any interactive preliminary communications in each data transmission and has no assumption that tamper free modules are available. This kind of scheme definitely provides a new scheme with a longer and higher level of security than the schemes that based on a factoring and discrete logarithm. The proposed scheme also requires minimal operations in encryption and decryption algorithms and thus makes it very efficient. Based on the fact that re-inventing a new scheme involves many uncertain and unknown threats, and integer factorization and discrete logarithm based schemes are widely deployed. This solution can be directly deployed in the currently used system with very low cost. Therefore, our new scheme is more practical and has the same security as the original integer factorization and discrete logarithm based system.

References

[1] Shamir A. "*Identity-based cryptosystem and signature scheme*". Advances in Cryptology: Proceedings of Crypto' (Lecture Notes in Computer Science 196). Berlin, West Germany: Springer-Verlag, 1985, 84:47-53.
[2] Tsujii S and Itoh T. "An ID-based cryptosystem based on the discrete logarithm problem". *IEEE Jounral on selected areas in communications.* 1989, 7:467-473.
[3] ElGmal T. "A public key cryptosystem and a signature scheme based on discrete logarithms". *IEEE Trans. Inform. Theory*. 1995; 31:469-472.
[4] W Diffie W and Hellman ME. "New direction in cryptography". *IEEE Trans. Inform. Theory*. 1976; 22: 644-654.
[5] Kohnfelder LM. "*A method for certification*". Lab. Comput. Sci. Mass. Inst. Technol. Cambridge, MA. 1978.
[6] Okamoto E and Tanaka K. "Key distribution system based on identification information". *IEEE J. SeIecr. Areas Commun*. 1989; 7: 481-485.

[7] Blom R. “An optimal class of symmetric key generation systems”. In *Proc. Eurocryp ‘84,* Pans, France. 1984; 335-338.
[8] Lee WB and Liao KC. “Constructing identity-based cryptosystems for discrete logarithm based cryptosystems”. *Journal of Network and Computer Applications*. 2004; 27:191–199.
[9] Hwang MS, Lo JW and Lin SC. “An efficient user identification scheme based on ID-based cryptosystem”. *Computer Standards & Interfaces*. 2004; 26: 565–569.
[10] Ohta K. “Efficient identification and signatureschemes”. *Electron. Lett*. 1988; 24(2):115-116.
[11] Bellare M, Namprempre C and Neven G. “Security Proofs for Identity-Based Identification and Signature Schemes”. *J. Cryptol*. 2009; 22: 1–61.
[12] Merkle RC and Hellman ME. “Hiding information and signatures in trapdoor knapsacks”. *IEEE Trans. Inform. Theory*. 1978; 24: 525-530.
[13] Harn L. “Public key cryptosystem design based on factoring and discrete logarithm”. *IEE Pro. Comput. Digit.Tech*. 1994; 141(3): 193-195.
[14] Gordon J. “Strong RSA keys”. *Electron. Lett*. 1984; 20(12): 514-516.
[15] Kiltz E and Vahlis Y. “CCA2 Secure IBE: Standard model efficiency through authenticated symmetric encryption”. In *CT-RSA*, Vol. 4964 of Lecture Notes in Computer Science 2008: 221–239 (Springer).
[16] Meshram C. “A Cryptosystem based on Double Generalized Discrete Logarithm Problem”. *Int. J. Contemp. Math. Sciences*. 2011; 6(6): 285 -297.
[17] Meshram C. “Modified ID-Based Public key Cryptosystem using Double Discrete Logarithm Problem”. *International Journal of Advanced Computer Science and Applications.* 2010; 1(6): 30-34.
[18] Meshram C & Shyam Sundar Agrawal SS. “An ID-Based Public Key Cryptosystem based on Integer Factoring and Double Discrete Logarithm Problem”. *Information Assurance and Security Letters*. 2010; 1: 29-34.
[19] Gangishetti R, Gorantla MC, Das ML, Saxena A. “Threshold key issuing in identity-based cryptosystems”. *Computer Standards & Interfaces*. 2007; 29: 260–264.
[20] Sun J, Zhang C, Zhang Y, and Fang Y. “An Identity-Based Security System for User Privacy in Vehicular Ad Hoc Networks”. *IEEE Tran. On Parall and Distributed Systems*. 2010; 27(9): 1227-1239.
[21] Boneh D and Franklin MK. “Identity based encryption from the Weil pairing”. *SIAM Journal on Computing*. 2003; 32(3): 586–615.
[22] Boneh D, Canetti R, Halevi S, and Katz J. “Chosen-ciphertext security from identity-based encryption”. *SIAM Journal on Computing.* 2006; 5(36): 1301–1328.
[23] Maurer UM and Yacobi Y. “Non-interactive public key cryptography”. Cryptology—Eurocrypt’91, New York: Springer. 1991: 498–507.
[24] Maurer UM and Yacobi Y. “A non-interactive public-key distribution system”. *Des. Codes. Cryptogr*. 1996; 9(3): 305–316.
[25] Tseng YM and Jan JK. “ID-based cryptographic schemes using a non-interactive public-key distribution system”. The *14th Annual Computer Security Applications Conference*. 1998: 237–243.
[26] Cocks C. “*An Identity Based Encryption Scheme Based on Quadratic Residues*”. Cryptography and Coding - Institute of Mathematics and Its Applications International Conference on Cryptography and Coding {Proceedings of IMA 2001, LNCS 2260:360-363, Springer-Verlag, (2001)}.
[27] Coppersmith D. “private communication”. Nov. 1987.
[28] Shamir A. “private communication”. June 1988.
[29] Barnett S. “Matrix methods for engineers and scientists”. McGraw-Hill Book Company. 1979.
[30] Lenstra AK, Lenstra HW, Manesse MS, and Pollard JM. “*The number field sieve*”. Proc. 22nd ACM Symp. On Theory of Computing, Baltimore, Maryland, USA. 1990: 564-572.
[31] Meshram C and Meshram SA. “*An Identity based Beta Cryptosystem*”. IEEE Proceedings of 7th International Conference on Information Assurance and Security (IAS 2011). 2011: 298-303.
[32] Schneier B. “*Applied Cryptography: Protocols, Algorithms and Source Code in C*”. 2nd Edn. Wiley. 1996.