Fully Self-organized Key Management Scheme in MANET and Its Applications

Fuyou Miao, Wenjing Ruan, Xianchang Du and Suwan Wang

Abstract The paper first proposes a fully self-organized key management scheme in mobile ad hoc networks which is both certificateless and free from any trusted third party such as Certificate Authority or Key Generation Center. The scheme allows a node to set up the public/private key pair all by itself and use the public key as its identity according to the property of ad hoc networks. Based on the scheme, some applications such as Encryption, Signature, Signcryption algorithms are given. Among these algorithms, the security of encryption algorithms is detailed and the AdvancedEnc is proved to be IND-CCA secure encryption algorithm under the random oracle model. These applications show that our key management scheme is self-organized, simple, efficient and practical.

Keywords Mobile network · Ad hoc · Key management · Fully self-organized · Random oracle model

1 Introduction

Mobile ad hoc networks (MANETs) are wireless mobile multi-hops networks which do not rely on any fixed infrastructure, and can be widely used in the battlefield, counter-terrorism, emergency rescue, as well as a variety of occasions which lack of fixed network. It is a necessary complementation and extension to the Internet and it has also been one of the hotspots of network research in recent years.

F. Miao · W. Ruan (✉) · X. Du
School of Computer Science and Technology, University of Science
and Technology of China, He Fei, China
e-mail: ruanwj@mail.ustc.edu.cn

S. Wang
School of Computer Science and Technology, University of Anhui, He Fei, China

N. Chaki et al. (eds.), *Computer Networks & Communications (NetCom)*,
Lecture Notes in Electrical Engineering 131, DOI: 10.1007/978-1-4614-6154-8_38,

In addition to mobility, self-organized, MANETs also tend to have such characteristics as full distribution, and temporary. Specifically, there aren't any fixed infrastructures to provide services for nodes in the networks. All nodes are equal and often need to collaborate to complete a task; each node is completely independent to manage and control their own resources and determine their own behaviors; MANETs tend to set up for a particular purpose or temporary emergency. Mostly due to that the nodes in the network are mobile devices and their energy is often limited, so these nodes may not exist for a long time.

Similar to Internet, security in MANET is a critical problem. For instance, messages sent from a source to the destination often need to be confidential; digital signature is often required to ensure the integrity and right source of a message. All these security mechanisms are based on key management in MANETs, which provides key generation and distribution for them.

However, the traditional certificate-based key management scheme PKI needs a fixed Certificate Authority to manage certificates for all nodes; moreover, the certificate-based scheme uses a certificate to bind the public key and identity of a node which results in the complication and inefficiency of key management. The identity based key management scheme, although use identity as the public key and is of no certificate, needs a Private Key Generator as the trusted third party. Therefore, neither PKI nor identity based scheme fits for the mobile ad hoc networks, which is characterized by self-organization, distribution and autonomy.

Thus, allowing for the above properties of MANET, the paper proposes a fully self-organized key management scheme, which is both certificateless and free from any trusted third party such as Certificate Authority or Key Generator Center. The scheme allows a node to set up the public/private key pair all by itself and use the public key as its identity according to the property of ad hoc networks. Besides, some applications and proofs are given to show the practical availability.

The paper is organized as follows: In part 2, related work gives the current related mechanisms, their advantages and disadvantages; it is followed by the preliminary knowledge in part 3. Part 4 describes the specific our scheme and its applications in detail. Part 5 deals with the security proof of encryption application. At last, a brief summary of our work and some future work is given in part 6.

2 Related Works

For certificate-based key management mechanism [1], there is a Certificate Authority (CA) which is responsible for certificate issue, update and revocation. However, different from the traditional fixed networks, mobile ad hoc networks are of the above mentioned characteristics: distribution, self-organization, autonomy. So, it is difficult to designate a node as such a CA.

Authors of papers [2–4] introduce a (t, n) threshold public key cryptography in MANET in which the CA is composed of n servers. Many public key certificate management operations are completed by at least t out of n servers together. Obviously, its efficiency becomes low and some of these server nodes are easily exploited by adversaries. Many other similar schemes have the same problems.

Miao Fuyou [5] proposed a one-way hash chain based on PKI for MANETs without an online trusted third party. Through a node can initialize, refresh, and revoke its certificate just by itself, but the length of the one-way hash chain is predetermined; the largest number of update cycles is limited. Over a period of time, the one-way hash chain must be changed.

In 1984 Shamir proposed the concept of ID-based cryptography [6, 7] in which the public key can be arbitrary string. It has more advantages compared with certificate-based key management schemes. User's public key can be a hash value of user's name, address, E-mail, identity card number. The user's private key is generated by the Key Generation Center (KGC). Identity-based cryptography allows any two users can securely communicate, and need not to exchange public key certificates or save a list of public key certificates.

The above mechanism doesn't fit for MANET. Firstly, for fully self-organized mobile network model, the existence of KGC is impractical. For individual nodes, it is difficult to establish or vote a fully trusted third party through a good mechanism. Secondly, before a node joins the network it must communicate with the KGC, and its private key generation must rely on the KGC. KGC maintenance will take a lot of manpower and material resources. Such mechanism is of the key escrow problem. KGC knows all nodes' private keys, so if it is captured by an attacker or it takes the initiative to make malicious behaviors, the consequences will be devastating. Thirdly, the node private key during transmission process is easy to be intercepted, and if so, the node intercepting other's private key can impersonate other nodes in the network to communicate. As a result, some researchers have proposed the concept of distributed KGC (D-KGC) [8, 9]. In the initial stage, choose n nodes as master nodes and generate a secret share for each master node by the mechanism of (n, t) threshold. Thus any at least t master nodes can recover the master key. Each new node's private key generation must be completed by at least t master nodes rather than a single KGC. This solves the problem of key escrow, but it makes network traffic largely increase to communicate with at least t master nodes. What's more, the secure transmission of private key shares is still a problem.

3 Preliminaries

3.1 Notation

To better understand the main idea of our system, we first list the main notations used in the paper in Table 1.

Table 1 Main notations used in our system

Notations	Description
p, q	Two large primes
G_1	A q-order subgroup of the additive group of points of E/Fp
G_2	A q-order subgroup of the multiplicative group of the F^*_{p2}
e	Pairing s.t. $G_1 \times G_1 \rightarrow G_2$
H_1	A hash function: $G_2 \rightarrow \{0, 1\}n$
H	A hash function: $\{0, 1\}^* \times G_2 \rightarrow Zq^*$
P	Generator of G_1
Q	$\in G_1$, another point of E/Fp
ID_A	Network ID of node A
x_A	Secret key of node A
PA_{pub}	Public key of node A

3.2 Consideration on Notation of Nodes in MANET

It is well known that nodes in MANET are equal, and each node is both a host and a router. Unlike the identity of IP address in Internet which consists of network address and host address, the identity of a MANET node is often unstructured and used just to distinguish one node from the others. Then we can designate the public key as the identity of a MANET node, If we must distinguish between different networks, other parts can be added to the public key as the whole identity. In fact, what we need is to let the identity include the public key of a node.

Although the identity appears to be complex, we must recognize the fact that: the identity of a user is stored in the computer system and the communications among users are largely dependent on the computer system. So a public key can be used as the user's identity.

We consider a fully self-organized mobile ad hoc network without a trusted third party as the system administrator or network planner. A node joins and leaves the MANET dynamically and autonomously. There are only one or several nodes in the initial network. Subsequent nodes can freely join and leave the network. A new node which wants to join this network can select its secret key and compute the public key all by itself. Next it deals with the public key as its own identity. Since a public key is used as the identity of a node, getting the identity of a node means obtaining the public key.

3.3 Mathematical Problems

Bilinear Pairing: suppose G_1 and G_2 are an additive cyclic group and multiplication cyclic group of the same prime order q respectively, so that let bilinear pairings e: $G_1 \times G_1 \rightarrow G_2$ satisfy the following properties:

1. Bilinear: $e(aP, bQ) = e(P, Q)^{ab}$ for all $P, Q \in G_1$ and $a, b \in Zq^*$.
2. Non-degenerate: exist $P \in G_1$and $Q \in G_1$, make $e(P, Q) \neq 1$.
3. Computability: There is an effective algorithm to compute e(P, Q), for any P, $Q \in G_1$.

The Bilinear Diffie-Hellman Problem (BDH): Let G_1, G_2 be an additive cyclic group and multiplication cyclic group of prime order q and $e: G_1 \times G_1 \rightarrow G_2$, be an admissible bilinear map. Given {P, aP, bP, cP}, to compute $e(P, P)^{abc}$ is called Bilinear Diffie-Hellman Problem (BDH), where P is a generator of G_1, a, b, c are random values in Zq*. The BDH problem is supposed to be intractable at present and is one of the foundations for constructing a public key scheme.

4 Fully Self-Organized Key Management Scheme

4.1 Public Key Scheme

As mentioned above, the identity of an ad hoc network node could be the IP/MAC address, email address or any other information that can be used to identify the node uniquely. Thus we can designate the public key as the identity of an ad hoc node.

Setup: Assume all nodes in an ad hoc network share the parameters {Q, P}, where $P, Q \in G_1$, G_1 is a cyclic additive group with the prime order of q; P is a generator of G_1, G_2 is a cyclic multiplicative group with the same order q, e is a bilinear pairing: $G_1 \times G_1 \rightarrow G_2$;

Key Generation: Before joining an ad hoc, a node A first selects by itself a private key $x_A \in Zq^*$, computes public key $PA_{pub} = x_A P$, and specifies PA_{pub} as its identity. Thus the key pair of node A is (x_A, PA_{pub}). Obviously, (x_A, PA_{pub}) is different from an identity based key pair, a node once obtains PA_{pub} , the identity of A, it knows the public key of A as well.

Consideration on Key Update/ Revocation: As mentioned above, a mobile ad hoc network is often temporary and nodes exist just for a short period of time; moreover, the public key is used as the identity of a node; therefore, the public key of a node should keep unchanged once it joins in the network. In fact, if a node updates its public key (identity) silently or leaves the network [10, 11], it means the node revokes its old public key (identity).

The proposed public key scheme has many advantages. (1) Efficient and Simple: Compared to the traditional certificate-based key management schemes, the proposed scheme is free from all the certificate operations such as distribution, validation, update and revocation; in contrast with ID-based schemes, the proposed scheme needs not to communicate with the KGC. That is, the proposed scheme is a light-weight one. (2) Self-organized: on the one hand, the traditional certificate-based key management relies on the support of CA online. On the other hand, although ID-based key management scheme can directly acquire the public key of a node according to its identity information, but the calculation of the private key must rely on KGC. Once the KGC is comprised, the security of the whole system will collapse. But the proposed scheme does not need any support of TTP and all key management operations are done by the node itself.

4.2 Some Applications

Next, we will give some applications of the proposed key management scheme to show the availability. Among these applications we detail the encryption algorithm AdvancedEnc and prove it to be IND-CCA secure.

4.2.1 BasicEnc: An IND-CPA Secure Encryption Algorithm

Encrypt: if node A wants to send a message m to node B (which holds the private key $\mathrm{x_B}$, and the public key $\mathrm{PB}_{pub} = \mathrm{x_B P}$), it does as follows:
A first selects r∈ Zq* at random, computes:

$$\begin{aligned} \mathrm{Q_{prt}} &= \mathrm{rQ} \\ \mathrm{U} &= \mathrm{rP}, \text{ keeps r and } \mathrm{Q_{prt}} \text{ in private}, \\ \mathrm{V} &= \mathbf{m} \oplus \mathrm{H_1}(\mathrm{e}(\mathrm{PB}_{pub}, \mathrm{Q_{prt}})) \end{aligned} \tag{1}$$

Thus {U, V} is the ciphertext C that A sends to B.
Decrypt: Receiving the ciphertext C, B decrypts it as follows to get the original message:

$$\begin{aligned} \mathbf{m} &= \mathrm{V} \oplus \mathrm{H_1}(\mathrm{e}(\mathrm{U}, \mathrm{x_B Q})) \qquad (2) \\ &\quad \text{for} \mathrm{V} \oplus \mathrm{H_1}(\mathrm{e}(\mathrm{U}, \mathrm{x_B Q})) \\ &= \mathrm{V} \oplus \mathrm{H_1}(\mathrm{e}(\mathrm{rP}, \mathrm{x_B Q})) \\ &= \mathrm{V} \oplus \mathrm{H_1}(\mathrm{e}(\mathrm{x_B P}, \mathrm{rQ})) \\ &= \mathrm{V} \oplus \mathrm{H_1}(\mathrm{e}(\mathrm{PB}_{pub}, \mathrm{Q_{prt}})) \\ &= \mathbf{m}; \end{aligned}$$

We can simply find from the above equation that only node A and node B have the ability to resume the message m from C.

4.2.2 AdvancedEnc: An IND-CCA Secure Encryption Algorithm

Now in order to improve the security of BasicEnc to make it to be an IND-CCA2 secure (IND-CCA for short), we modify it as follows according to paper [12].

Let SynEnc be a symmetric encryption algorithm ($C = a \oplus M$; $M = C \oplus a$). Let the modified encryption algorithm be called Epk' described as follows:
$\text{AdvancedEnc}(M) = \{E_{pk}(\theta, H_2(\theta, M)); M \oplus H_3(\theta)\}$

Let n be the size of plaintext M, H_2 and H_3 are two hash functions, $H_2 : \{0, 1\}^n \times \{0, 1\}^n \rightarrow Zq^*$, $H_3 : \{0, 1\}^n \rightarrow \{0, 1\}^n$.
Key extraction: same as before.
Encrypt: select θ in $\{0, 1\}^n$ randomly, computes $r = H_2(\theta, M)$, the cipher text is

$$\begin{aligned} C &= \{rP, \theta \oplus H_1[e(PB_{pub}, rQ)], M \oplus H_3(\theta)\} \\ &= \{U, V, W\} \end{aligned}$$

Decrypt: ① $\theta = V \oplus H_1[e(x_B Q, U)]$
② $M = W \oplus H_3(\theta)$
③ if $r = H_2(\theta, M)$, accepted; else rejected.

4.2.3 Signature Protocol

Sign: if node A needs to sign a message **m**, it can do as follows:
A selects a random number $k \in Zq^*$ and computes $P_{prt} = rP$ and $Q_{pub} = rQ$, to sign m, it does as follows:

$$\begin{aligned} r &= e(x_A Q, P)^k \\ v &= H(m, r) \\ U &= (v + k)x_A Q \end{aligned} \tag{3}$$

Thus the signature of m is {U, v}.
Verify: If the verifier, e.g. node B, wants to verify the signature {U, v}, it computes:

$$r = e(U, P)e(vQ, -PA_{pub}) \tag{4}$$

And checks if $v = H(m, r)$ holds, if it does, the verification succeeds, or else it fails. That is because:

$$\begin{aligned} r &= e(U, P)e(vQ, -PA_{pub}) \\ &= e((v + k)xAQ, P)e(vQ, -xAP) \\ &= e(k\,xAQ, P) \\ &= e(xAQ, P)k \end{aligned}$$

The above procedure shows that only A has the ability to produce the signature because the private xA is used during the signing.

4.2.4 Signcryption Protocol

Sign and encrypt: if A needs to send a message m to B with confidentiality and integrity guaranteed, it computes c as the Signcryption:

$$c = m \oplus H_1(e(x_A Q, PB_{pub}))$$

Verify and decrypt: After receiving c from A, node B resumes m as follows:

$$m = c \oplus H_1(e(x_B Q, PA_{pub}))$$

5 Proof of Security for Our Encryption Mechanism in Random Oracle Model

In order to show the availability of the proposed fully self-organized key management scheme, we first prove the BasicEnc is IND-CPA secure and then prove that the AdvancedEnc is IND-CCA secure [13–15].

Lemma 5.1 Let H_1 be a random oracle: $G_2 \rightarrow \{0, 1\}^n$. n is the space size of plaintext and k is security parameter. Suppose A is an IND-CPA adversary with advantages ε (k) against BasicEnc and makes Q_{H1} queries to H_1 in total. Then there exists an algorithm B which can solve the BDH problem for G with advantage at least $2\,\varepsilon(k)/Q_{H1}$ and a running time O (time (A)).

Proof Assume CH is the BDH challenger and it always responds to B. The BDH parameters $\{q, G_1, G_2, e\}$ is produced by G and a random instance of BDH problem is {P, aP, bP, cP}, where a, b, c are randomly selected in Zq* and P is random value in G_1. Now let the BDH solution is $D = e(P, P)^{abc}$. CH sends {P, aP, bP, cP} to B, next B will find out the D through A.

Setup: Algorithm B creates $KEY_{pub} = \{q, G_1, G_2, e, n, PK_{pub}, Q, H_1\}$, where $PK_{pub} = aP, Q = bP$, and H_1 is a random oracle controlled by B. then B can be a challenger of A and sends the KEY_{pub} to adversary A. In fact we can see that the corresponding private key is a. But only CH knows it.
Query to H_1: the adversary A can issue a query to the random oracle H_1 at any time. Algorithm B will do the followings to respond to a query Q_i:

1. If Q_i has already appeared before, B will return the same result as the last denoted by $\{Q_i, H_i\}$;
2. Else then B selects a random number $H_i \in \{0, 1\}^n$ to respond to A and records $\{Q_i, H_i\}$ in the local.

Adversary A prepares two messages $\{M_0, M_1\}$ to B which are to be challenged and B sends them to CH. Then CH selects {P, aP, bP, cP} and send it to B. B returns

$C = M_b \oplus r$ to A, r is selected randomly; $b = 0$ or 1. And B sends ciphertext $C' = \{cP, C\}$ to A. by our definition, $Mb = R \oplus H_1[e(cP, aQ)] = R \oplus H_1$ [D].
Guess: Now A guesses $b' = 0$ or 1. According to the assumption, the correct probability for $b' = b$ is $\varepsilon + 1/2$. Then B selects a $\{\mathbf{Q}_i, H_i\}$ randomly from the local records and sends it to CH as the solution of BDH problem.

Since B provides a real attack environment for A, the probability that A comes up with a query for $H_1(D)$ among Q_{H1} queries (such an event is denoted by E) in above process is the same as in the real environment. Next, we will prove $\Pr[E] \geq 2\,\varepsilon$.

In the real attack, if A never issues a related query and just guesses b′ by intuition, it will success with the advantage $\Pr[b' = b|\neg E] = 1/2$.
Because

$$\Pr[b' = b] = \Pr[b' = b|\neg E] \cdot \Pr[\neg E] + \Pr[b' = b|E] \cdot \Pr[E]$$

And

$$0 \leq \Pr[b' = b|E] \leq 1$$

$$\Pr[b' = b] \geq 1/2 + \varepsilon$$

So

$$1/2 + \varepsilon \leq \Pr[b' = b] \leq \Pr[b' = b|\neg E] \cdot \Pr[\neg E] + \Pr[E]$$

Namely

$$1/2 + \varepsilon \leq \Pr[b' = b] \leq (1/2) \cdot \Pr[\neg E] + \Pr[E]$$
$$1/2 + \varepsilon \leq \Pr[b' = b] \leq (1/2)(1 + \Pr[E])$$

Then it follows

$$\Pr[E] \geq 2\,\varepsilon$$

Now we can know that among the total Q_{H1} queries, such an event E appears with the advantage at least 2ε.Thus, B will select $\mathbf{Q_i}$ from the local records as the response to CH with the correct probability at least $2\varepsilon / Q_{H1}$.

According to the above proof process, it follows that if there is an IND-CPA adversary A against BasicEnc, also we can find out an algorithm B which can solve the BDH problem with the non-negligible probability of at least $2\varepsilon / Q_{H1}$.

So the proposed fully self-organized key management scheme can be used to construct an IND-CPA secure encryption algorithm.

Theorem 5.2 (Chosen-Ciphertext Security) suppose that BasicEnc is a γ-uniform (γ is a probability value) asymmetric one-way encryption algorithm with (ε_1, t_1) and n is the size of plaintext M. Then the AdvancedEnc is IND-CCA secure in the random oracle model with (ε, t). Suppose the adversary of AdvancedEnc issues Q_d decryption queries, Q_{H2} queries to H_2 and Q_{H3} queries to H_3. (See Theorem 14 in paper [12])

$$\varepsilon = (2(Q_{H2} + Q_{H3})\varepsilon_1 + \varepsilon_2 + 1)(1 - 2\varepsilon_1 - 2\varepsilon_2 - \gamma - 2^{-n})^{-Qd} - 1$$
$$t = \min(t_1, t_2) - O((Q_{H2} + Q_{H3}) \cdot n)$$

For the modified encryption algorithm AdvancedEnc, θ, H_2 and H_3 are additional. We have already proved that BasicEnc is IND-CPA secure. To prove the Theorem 5.2, we come up with the Theorem 5.3.

Lemma 5.3 suppose that BasicEnc is a γ-uniform (γ is a probability value) asymmetric IND-CPA encryption algorithm in random oracle. Then it must be a γ-uniform (γ is a probability value) asymmetric one-way encryption algorithm under chosen plaintext attack.

Proof One-way encryption is weaker security notion compared to IND-CPA encryption. In the sense of OWE, let A is an adversary with public key *pk* and cipher text *C*. A can never get decryption services from anywhere and can just select a random message *m* from plaintext space to break the ciphertext with the successful probability of $1/2^n$ if n is the bit length of the plaintext.

However, beside the same prerequisites with OWE, an IND-CPA adversary B can also interact with its challenger and access the random oracles. That is to say, the IND-CPA adversary B is more capable than the OWE adversary in breaking ciphertexts. In order to distinguish between the M_0 and M_1, A has two means: (1) randomly guesses Mb from the two plaintexts; (2) Directly cracks the ciphertext to find out the plaintext. The successful probability of (1) is 1/2, and that of (2) relies on queries to the random oracle. Therefore, ignoring (1), we think that the successful probability for B in (2) must be greater than A in the sense of OWE since they have the same target to break a ciphertext. So if the BasicEnc is IND-CPA secure, it is certainly a OWE algorithm.

Thus the Theorem 5.2 can be proved to be correct by Lemma 5.3, it in turn follows that our modified encryption algorithm AdvancedEnc is IND-CCA secure in the random oracle model with a non-negligible advantage.

6 Conclusions

Since the attributes of full organization, distribution and temporary existence in mobile ad hoc networks, Traditional certificate based key management schemes and ID-based scheme can not fit for the network. Therefore, we propose a fully self-organized key management scheme, which is efficient, simple and TTP-free. To demonstrate the availability, some algorithms including encryption, signature, signcryption are given. Among these algorithms, the paper focuses on the security of encryption algorithms and proves the AdvancedEnc to be IND-CCA secure, which shows that our fully self-organized key management scheme is practical. In fact, we can also construct the practical signature, signcryption and authentication algorithms and prove them to be practical in security.

Acknowledgments The paper is supported by NSF of China (granted No. 60970128, No. 61170233, No. 61272472), Youth Innovation Foundation of USTC of 2010 and China Postdoctoral Science Foundation (No. 2011M501397).

References

1. Hegland AM, Winjum E (2006) Survey of key management in ad hoc networks. IEEE Commun Surv Tutor 8(5):48–66
2. Munivel E, Ajit GM (2010) Efficient public key infrastructure implementation wireless sensor networks. ICWCSC. doi:10.1109
3. Nisbet A, Rashid MA, Alam F (2010) The quest for optimum server location selection in mobile ad hoc networks utilising threshold cryptography. In: International conference on information technology: new generations, pp 891–896
4. Yan X, Fuyou M (2003) Secure distributed authentication based on multi-hop signing with encrypted signature functions in mobile ad hoc networks. Acta Electron Sinica 31(2):161–162
5. Miao Fuyou, Xiong Yan (2005) PKIM: a public key infrastructure for mobile ad hoc network. Chin J Electron 14(4):594–598
6. Shamir A (1985) Identity-based cryptosystems and signature schemes. Adv Cryptol 196:47–53
7. Li L, Wang Z, Liu W, Wang Y (2011) A certificateless key management scheme in mobile ad hoc networks. In: International conference on, wireless communications, pp 1–4
8. Li L-C, Liu R-S (2010) Securing cluster-based ad hoc networks with distributed authorities. IEEE Trans Wirel Commun 9(10):3072–3081
9. Lee H, Kim J, Son J, Kim S, Oh H (2012) ID-based key management scheme using threshold decryption for OPMD environment. IEEE ICCE-2012, pp 733–734
10. Wang Y, Zou X (2007) An efficient key revocation scheme for wireless sensor networks. IEEE ICC, pp 1260–1265
11. Li L, Wang Z, Liu W, Wang Y (2011) A certificateless key management scheme in mobile ad hoc networks. Int Conf Wirel Commun, pp 1–4
12. Fujisaki E, Okamoto T (1999) Secure integration of asymmetric and symmetric encryption schemes. In: Advances in cryptology-crypto '99, Lecture notes in computer science, vol 1666. Springer-Verlag, pp 537–554
13. Boneh D, Franklin M (2001) Identity-based encryption from the weil pairing. Comput Sci 2139:213–229
14. Katz J, Lindell Y (2008) Introduction to modern cryptogarphy: principles and protocols. National Defense Industry Press, China
15. Bellare M, Desai A, Pointcheval D, Rogaway P (1998) Relations among notions of security for public-key encryption schemes. Comput Sci 1462:26–45