Secret Key Awareness Security Public Key Encryption Scheme

Guoyan Zhang[⋆] and Qiuliang Xu

School of Computer Science and Technology, Shandong University, Jinan 250100, China
guoyanzhang@sdu.edu.cn,xql@sdu.edu.cn

Abstract. In this paper, firstly, we introduce a new security definition called secret key awareness security which is to guarantee anyone generating the public key to know the corresponding secret key. Following, we give a concrete implementing for secret key awareness security. Secondly, we present two applications: one is in plaintext awareness security cryptosystem, and another is in certificatless public key encryption scheme.

Key words: Secret Key Awareness Security, DHK1 Assumption, Non-Interactive Zero-Knowledge, Non-Black-Box

1 Introduction

For encryption scheme, a variety of goals including indistinguishability (IND), non-malleability (NM) and plaintext awareness (PA) have been introduced which are actually stronger than the notion of [1]. Indistinguishability (IND), due to Goldwasser and Micali [2], formalizes an adversary's inability to learn any information about the plaintext x underlying a challenge ciphertext y, capturing a strong notion of privacy. Non-malleability (NM)presented by Dolev, Dwork and Naor [3] formalizes an adversary's inability, given a challenge ciphertext y, to output a different ciphertext $y^{'}$ such that the corresponding plaintexts $x, x^{'}$ are meaningfully related. Plaintext awareness defined by M. Bellare, A. Desai, D. Pointcheval and P. Rogaway in [4] formalizes an adversary's inability to create a ciphertext y without knowing its underlying plaintext x. In this paper, we give the forth goal called secret key awareness security which formalizes an adversary's inability to create a valid public key pk without knowing its underlying secret key sk.

[⋆] This work is Supported by the National Natural Science Foundation of China(No.60873232), Open Research Fund from Key Laboratory of Computer Network and Information Integration In Southeast University, Ministry of Education, China, Shandong Natural Science Foundation(No.Y2008A22) and Shandong Postdoctoral Special Fund for Innovative Research(No.200902022).

1.1 Related Work

In fact, before we formally give the security notion of secret key awareness, it has found its application. In order to get a non-interactive $\sum$-protocol, Ivan Damgard, Nelly Fazio and Antonio Nicolosi [5] have forced the verifier to generate its public key and to register it with the random number chosen by the verifier to generate its public key. This required simultaneously that the verifier knew the secret key underlying its public key. Also in [6], Jonathan Herzog, Moses Likov and Silvio Micali got a generic construction for public key encryption scheme with plaintext awareness security via key registration, in which the sender generating public key should be engaged in a zero-knowledge proof of knowledge that he knew the corresponding secret key with registration authority.

1.2 Our Contributions

We formally introduce a new security notion called secret key awareness (SKA) security in public key cryptography, and this security notion requires any adversary generating its public key must know the corresponding secret key. It is to say, there is a secret key extractor which can extract the secret key through revising the adversary or getting the adversary's transcripts as input.

Following, we give a concrete implementing for cryptosystem with secret key awareness security by modifying ELGamal encryption protocol.

Lastly, two applications are presented. One is a construction for plaintext awareness security encryption scheme. Plaintext awareness defined by Bellare, Desai, Pointcheval and Rogaway in [4] and they showed that PA+IND-CPA should imply IND-CCA2. But most of the practical encryption protocols with plaintext awareness security need random oracle. Mihir Bellare and Adriana Palacio [7] gave a new notion of plaintext awareness in standard model, in which they presented a concrete protocol with non-black-box technology. Jonathan Herzog, Moses Likov and Silvio Micali [6] got a generic public key encryption scheme with plaintext awareness security via key registration. Compared with their construction, our construction omits the registration authority and zero-knowledge proof by the use of non-black-box technology in reduction. Another is a construction for certificateless public key encryption by modifying the construction [8] which could be proved secure against the strong Type I adversary. But in the construction [8], the Type I adversary could generate the valid ciphertext different with the challenge ciphertext for the same message by replacing the users's public key without knowing the message. Our construction makes use of a CPA secure id-based public key encryption scheme, a CPA secure public key encryption scheme with secret key awareness secure, and a non-malleable $NIZK$ proof system for any NP as components, and the construction is secure against the strong Type I adversary.

2 Preliminaries

2.1 Computational Complexity

Assumption $[DHK1]$ Let G be a prime-order-group generator. Let H be an algorithm that has access to an oracle, takes two primes and two group elements, and returns nothing. Let H^* be an algorithm that takes a pair of group elements and some state information, and returns an exponent and a new state. We call H a $DHK1$-adversary and H^* a $DHK1$-extractor. For $k \in \mathbb{N}$, we say that G satisfies the $DHK1$ assumption in the following experiment, if for every polynomial-time $DHK1$-adversary H, there exists a polynomial-time $DHK1$-extractor H^* such that

$$Adv_{G,H,H^*}^{DHK1}(k) = Pr[Exp_{G,H,H^*}^{DHK1}(k) = 1]$$

is negligible.

```
    Experiment $Exp_{G,H,H^*}^{DHK1}(k)$
    $(p,q,g) \leftarrow G(1^k); a \leftarrow \mathbb{Z}_q; A \leftarrow g^a modp$
Choose coins $R[H], R[H^*]$ for $H, H^*$, respectively; $St[H^*] \leftarrow ((p,q,g,A), R[H])$
Run $H$ on input $(p,q,g,A)$ and coins $R[H]$ until it halts, replying to its oracle queries as follows:
    -If $H$ makes query $(B,W)$ then
    $(b, St[H^*]) \leftarrow H^*((B,W), St[H^*]; R[H^*]])$
    If $W \equiv B^a(modp)$ and $B \neq g^b(modp)$ then return 1.
    Else return $b$ to $H$ as the reply End If
Return 0
```

2.2 Certificateless Public Key Encryption

Definition 1. (Certificateless Public Key Encryption). A generic certificateless public key encryption scheme consists of the following seven algorithms:

-SetUp(SU): a probabilistic polynomial time (PPT) algorithm run by a key generation center (KGC) given a security parameter k as input, which outputs a randomly chosen master secret key msk and master public key mpk. The master public key mpk includes a description of the message space $\mathcal{M}$ and ciphertext space $\mathcal{C}$.

-PartialPrivateKeyExtract(PPKE): given the master public key mpk, master secret key msk and an identifier for entity A, ID_A, the KGC runs this PPT algorithm to generate the partial private key d_A. Then the partial private key d_A is transported to entity A over a confidential and authentic channel.

-SetSecretValue(SSV): a PPT algorithm run by the entity A given master public key mpk and ID_A as input, which outputs a secret value x_A.

-SetPrivateKey(SPVK): given master public key mpk, the entity A's secret value x_A, and the entity A's partial private key d_A as input, the entity runs this PPT algorithm to generate a private key SK_A.

-SetPublicKey(SPK): given master public key mpk and the entity A's secret value x_A, and output a public key PK_A for the entity A.

-Encrypt($\mathcal{E}$): given a plaintext $M \in \mathcal{M}$, master public key mpk, an identifier ID_A and public key PK_A for an entity A as input, a sender runs this PPT algorithm to create a ciphertext $C \in \mathcal{C}$ or the null symbol $\perp$ indicating an encryption failure. This will always occur in the event that PK_A does not have the correct form.

-Decrypt($\mathcal{D}$): given master public key mpk, the entity's private key SK_A, and the ciphertext $C \in \mathcal{C}$ as inputs, the entity as a recipient runs this deterministic algorithm to get a decryption σ, which is either a plaintext message or a "reject" message.

3 Secret Key Awareness Security

3.1 Definition

Definition 2 (Secret Key Awareness Security-SKA). Let $\Pi = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ be a public key encryption scheme, let $\mathcal{B}$ be a polynomial-time adversary called public key creator who can generate a valid public key. Let $\mathcal{K}$ be an algorithm that takes the user's public key and some state information, returns the corresponding secret key and a new state. We call $\mathcal{K}$ a secret key extractor. For $k \in \mathbb{N}$, we say that a public key encryption scheme Π satisfies the secret key awareness security in the following experiment, if for every polynomial time adversary $\mathcal{B}$ generating valid public key, there is a polynomial time secret key extractor $\mathcal{K}$ such that

$$Adv^{SKA}_{\Pi,\mathcal{B},\mathcal{K}}(k) = Pr[Exp^{SKA}_{\Pi,\mathcal{B},\mathcal{K}}(k) = 1]$$

is negligible.

```
    Experiment $Exp^{SKA}_{\Pi,\mathcal{B},\mathcal{K}}(k)$
    $(PK = (pk, parameters), SK) \leftarrow \mathcal{G}(1^k)$
Choose coins $R[\mathcal{B}], R[\mathcal{K}]$ for $\mathcal{B}, \mathcal{K}$, respectively; $St[\mathcal{K}] \leftarrow (parameters, R[\mathcal{B}])$
Run $\mathcal{B}$ on input $parameters$ and coins $R[\mathcal{B}]$ until it halts, replying to its oracle queries as follows:
    -If $\mathcal{B}$ makes query $pk^{'}$ then
    $(x, St[\mathcal{K}]) \leftarrow \mathcal{K}[pk^{'}, St[\mathcal{K}]; R[\mathcal{K}]]$
    If $x$ isn't the secret key corresponding with $pk^{'}$ then return 1.
    Else return $x$ to $\mathcal{B}$ as the reply End If
Return 0
```

From the above definition, the secret key extractor is non-black-box, and allows the extractor code to depend non-uniformly on the code of the public key creator. Again, this is done in order to increase the possibility of finding constructions. Evidence of the power of non-black-box formulations is provided in another context by [9].

In order to obtain this kind of security, there are two usual technology considered, one is the non-interactive zero knowledge proof system to prove the possession of the secret key, and the knowledge extractor can extract corresponding knowledge by revising the adversaries creating the valid public key, but the existence of the non-interactive zero-knowledge greatly reduces the efficiency of schemes. Another makes use of a non-black-box reduction model, in which the knowledge extractor can get the code of the adversary, and the $R[\mathcal{B}]$ including the random choice of the adversaries, and certainly can extract the corresponding secret key.

3.2 Implementing Secret Key Awareness with Non-Black-Box Technology

Intuitively, the secret key awareness security is to guarantee the public key has the correct format so that anyone generating the public key to know the corresponding secret key. In order to obtain this kind of security, a traditional public key cryptographic scheme needn't be changed a lot, it is to say, the security requirement can't make the scheme more realizable and more secure, but it is easy to make a scheme to satisfy secret key awareness security without effecting the scheme's security and efficiency, For example, we can easily modify the key generation algorithm of $ELGamal$ encryption scheme to make it be secret key awareness security as follows:

- **SetUp:** Let p be a prime, and g_1, g_2 be generators of $\mathcal{Z}_p$. The private key x is an integer between 1 and $p-2$. Let $y_1 = g_1^x modp, y_2 = g_2^x modp$. The public key is the tuple (p, g_1, g_2, y_1, y_2).
- $\mathcal{E}^{ELG}$ **and** $\mathcal{D}^{ELG}$**:** The encryption algorithm $\mathcal{E}^{ELG}$ and the decryption algorithm $\mathcal{D}^{ELG}$ are the same to the original $ElGamal$ encryption scheme.

Theorem 1. *The above encryption scheme modified from ElGamal encryption scheme is secret key awareness secure under the* $DHK1$ *assumption and the other security properties are same with ElGamal encryption scheme.*

Proof: According to $DHK1$ assumption, obviously, the above scheme is secret key awareness secure. Furthermore, the extra public key information g_2, y_2 have not leaked any information about the secret key, and the encryption algorithm and decryption scheme are same to the original scheme, so the other security properties are remained.

4 Two Applications

4.1 A Construction for Encryption Scheme with Plaintext Awareness Security

Let $\pi_1 = (G, E, D)$ is an indistinguishable secure cryptosystem against chosen plaintext attack with secret key awareness security, and $\pi_2 = (G^{'}, E^{'}, D^{'})$ is a semantically secure cryptosystem against chosen plaintext attack. π is a non-interactive zero knowledge protocol with non-malleable for NP.

A construction for $CCA2$ secure encryption scheme with plaintext awareness security $\mathit{\Pi} = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ is the following:

-$\mathcal{G}$: **(Key Generation)**

(Receiver Key Generation): Run $G^{'}$ to generate two public-secret key pairs (e_1, d_1) and (e_2, d_2). Choose a random δ. The receiver's public key is $pk_r = (e_1, e_2, \delta)$, and the secret key is $sk_r = (d_1, d_2)$.

(Sender Key Generation): Run G to generate the sender's public-secret key pair (e_s, d_s).

-$\mathcal{E}$: On input $(m, (e_1, e_2, \delta), (e_s))$, compute $c_1 = E^{'}(e_1, m), c_2 = E^{'}(e_2, m)$ and $c_3 = E(e_s, m)$. With public reference string δ, compute a non-malleable non-interactive zero knowledge protocol π to prove that c_1, c_2 and c_3 are all the ciphertexts of the same message under the public key of e_1, e_2 and e_s respectively. Output c_1, c_2, c_3 and π as the ciphertext.

-$\mathcal{D}$: Receive ciphertext (c_1, c_2, c_3, π). Verify the validity of π. If so, output $D^{'}(c_1, d_1)$. Otherwise, output $\perp$.

Theorem 2. *Π is indistinguishable secure against the chosen ciphertext attack assuming public key encryption schemes π_1 and π_2 are indistinguishable secure against chosen plaintext attack.*

Proof: The security proof is similar to the proof of chosen ciphertext security in [4], and we simply describe it here.

Suppose there is an adversary A that succeeds in an adaptive chosen ciphertext attack against scheme Π with non-negligible advantage ε, we can construct two adversaries B_1 against scheme π_1 and B_2 against scheme π_2 in chosen plaintext attack.

Given $(e, 1^k)$, B_2 sets $e_2 = e$ and $(e_1, d_1) = G^{'}(1^k)$. He generates a public reference string δ for non-interactive zero knowledge proof protocol π. Set $(e_s, d_s) = G(1^k)$. B_2 can finish the chosen plaintext attack using A as following:

- When A asks the decryption oracle (c_1, c_2, c_3, π), B_2 checks the validity of π. If invalid, he outputs $\perp$. Else, he returns $D^{'}(d_1, c_1)$. According to the soundness of the non-interactive zero-knowledge protocol, B_2 always correctly answers the decryption oracle.
- When gives two equal length messages (m_0, m_1), B_2 sends out and receives the challenge c. Supposing $c = E^{'}(e, m_a)$, he sets $c_2 = c$ and respectively chooses $b \in \{0, 1\}$ and $\beta \in \{0, 1\}$ with probability $1/2$. B_2 computes $c_1 = E^{'}(e_1, m_b)$ and $c_3 = E(e_s, m_\beta)$. Then there are mainly three cases:
  - $b = a = \beta$: in this case, B_2 can output correctly a with probability $3/4 + \varepsilon/2$.
  - $b = a \neq \beta$: in this case, we assume that A can guess a with probability x. Then B_2 can output a with probability $1/2 + x/2$.
  - $b \neq a$: in this case, B_2 can output a with probability $1/4$.

  Taking into account all cases, the probability that B_2 is correct is

$$7/16 + (\varepsilon + x)/8.$$

We can see that B_2 will succeed with non-negligible advantage if $\varepsilon + x$ is non-negligibly different from $1/2$. This conflicts with the security of scheme π_2.

If $\varepsilon + x$ is negligibly different from $1/2$, we can construct B_1 to break the chosen plaintext security of π_1: Given $(e, 1^k)$, B_1 sets $(e_1, d_1) = G^{'}(1^k)$, $(e_2, d_2) = G^{'}(1^k)$ and $e_s = e$, B_1 can finish the chosen ciphertext attack using A as following:

- When A asks the decryption oracle (c_1, c_2, c_3, π), B_1 checks the validity of π. If invalid, he outputs $\perp$. Else, he returns $D^{'}(d_1, c_1)$. According to the soundness of the non-interactive zero-knowledge protocol, B_1 always correctly answers the decryption oracle.
- When A gives two equal length messages (m_0, m_1), B_1 outputs the message pair and receives the challenge c. Supposing $c = E(e, m_a)$, he sets $c_3 = c$ and chooses $b \in \{0, 1\}$ with probability $1/2$. B_1 computes $c_1 = E^{'}(e_1, m_b)$ and $c_2 = E^{'}(e_2, m_b)$. Then there are mainly two cases:
  - $b = a$: in this case, B_1 can correctly output a with probability $1/2 + \varepsilon$.
  - $b \neq a$: in this case, Then B_1 can output a with probability $1 - x$.

  Taking into account all cases, the probability that B_1 is correct is $1/2 + \varepsilon/2$, which conflicts with the security of scheme π_1.

Theorem 3. *Π is plaintext awareness security assuming public key encryption scheme π_1 is secret key awareness secure.*

Proof: The proof is simple. Assuming that A is an attacker who generates its public key pk with random coin $R[A]$ and ciphertext (c_1, c_2, c_3, π). Because of the secret key awareness security of scheme π_1, there is a secret key extractor $\mathcal{K}$ which can extract secret key d_s, given A's random coin $R[A]$. Then if π is invalid, output $\perp$. Else, output the message $m = D(d_s, c_3)$. According to the soundness of the non-malleable $NIZK$ protocol, the output is always correct.

4.2 A Construction for Certificatless Encryption Scheme Secure in the Strong Model

Let $\Pi^{IBE} = (Setup^{IBE}, Extract^{IBE}, \mathcal{E}^{IBE}, \mathcal{D}^{IBE})$ be an IBE scheme secure against chosen plaintext attack and $\Pi^{PKE} = (\mathcal{K}^{PKE}, \mathcal{E}^{PKE}_{pk}, \mathcal{D}^{PKE}_{sk})$ denotes a traditional public key encryption scheme that is CPA-secure with secret key awareness security. Let (f, P, V, S_1, S_2) be a statistically sound and computationally simulation-sound $NIZK$ proof system for the language

$$L = \{(C_1, pk_A, ID_A, C_2, pk_B) | \exists (m, r_1, r_2),$$
$$C_1 = \mathcal{E}^{PKE}_{pk_A}(\mathcal{E}^{IBE}_{ID_A}(m, r_1)) \wedge C_2 = \mathcal{E}^{PKE}_{pk_B}(m, r_2)\}.$$

A generic construction for certificateless public key encryption scheme $\Pi^{CLE} = (SU, PPKE, SSV, SPVK, SPK, \mathcal{E}, \mathcal{D})$ can be obtained in the following:

Setup (SU): is an algorithm running the setup algorithm of Π^{IBE}. The message space of Π^{CLE} is the common part of the message space of Π^{PKE} and the message space of Π^{IBE}, while its ciphertext space is the one of Π^{PKE}. Both schemes have to be compatible in that the plaintext space of Π^{PKE} must contain the ciphertext space of Π^{IBE}. Choose a random sring σ

Partial-Private-Key-Extract (PPKE): is the private key generation algorithm of Π^{IBE}, and runs the algorithm for the identity ID_A to get the partial private key d_A.

Set-Secret-Value (SSV): run the key generation procedure of Π^{PKE} to obtain a private key sk_A and a public key pk_A. sk_A is the secret value.

Set-Private-Key (SPVK): return $SA = (d_A, sk_A)$, where d_A is obtained by running the key generation algorithm of Π^{IBE} for the identity ID_A and sk_A is entity A's secret value obtained from Π^{PKE}'s key generation algorithm.

Set-Public-Key (SPK): output (ID_A, pk_A, σ) as the public key.

Encrypt ($\mathcal{E}$): to encrypt m using the identifier ID_A and the public key pk_A,

- check that pk_A has the right format for Π^{PKE}.
- run the key generation procedure of Π^{PKE} to obtain a private key sk_B and a public key pk_B.
- compute and output the ciphertext

$$C_1 = \mathcal{E}^{PKE}_{pk_A}(\mathcal{E}^{IBE}_{ID_A}(m, r_1)),$$

$$C_2 = \mathcal{E}^{PKE}_{pk_B}(m, r_2),$$

$$x \leftarrow (c_1, pk_A, ID_A, c_2, pk_B), \pi \leftarrow P(x, m, r_1, r_2, \sigma), C = (C_1, C_2, \pi, pk_B),$$

where $\mathcal{E}^{PKE}_{pk_A}$,$\mathcal{E}^{PKE}_{pk_B}$ and $\mathcal{E}^{IBE}_{ID_A}$ respectively denote the encryption algorithms of Π^{PKE} and Π^{IBE} for the public key pk_A,pk_B and the identity ID_A.

Decrypt ($\mathcal{D}$): to decrypt C using $SA = (d_A, sk_A)$,

- $x \leftarrow (c_1, pk_A, ID_A, c_2, pk_B)$, and verify $NIZK$, if $V(x, \pi, \sigma) \neq 1$, output $\perp$.
- else, compute $\mathcal{D}^{PKE}_{sk_A}(C_1)$ using the decryption algorithm of Π^{PKE}. If the result is $\perp$, return $\perp$ and reject the ciphertext. Otherwise,compute $\mathcal{D}^{IBE}_{d_A}(\mathcal{D}^{PKE}_{sk_A}(C_1))$ using the decryption algorithm of Π^{IBE} and return the result.

The security of Π^{CLE} can be proved by the security of Π^{PKE} and Π^{IBE}.

Theorem 4. *The above certificateless encryption scheme is Strong Type I and Strong Type II secure if Π^{IBE} and Π^{PKE} are secure against chosen plaintext attack and Π^{PKE} is also secret key awareness secure.*

5 Conclusion

This paper firstly introduces a new secure notion called secret key awareness security in public key cryptography, this secure notion requires any adversary creating users' public keys must know the corresponding secret keys. Then we give a concrete implementing for it. Following, we present two applications, one is a construction for a plaintext awareness secure encryption scheme, and another is concrete construction for certificateless cryptosystem secure in strong attack model. Although there isn't great improvement in the two aspects by the use of the new security notion, we believe that secret key awareness security is of more independent interesting.

References

1. Sattam S. Al-Riyami, Kenneth G. Paterson. Certificateless public key cryptography. In: Chi-Sung Laih, editors, Proceeding of the ASIACRYPT03. LNCS, Vol. 2894. pp. 452-473. Springer-Verlag, (2003).
2. J. Baek, R. Safavi-Naini, and W. Susilo. Certificateless public key encryption without pairing. In J. Zhou and J. Lopez, editors, Proceedings of the 8th International Conference on Information Security (ISC 2005). LNCS, Vol. 3650. pp. 134-148. Springer-Verlag, (2005).
3. K. Bentahar, P. Farshim, J. Malone-Lee, and N. P. Smart. Generic constructions of identity-based and certificateless KEMs. Available from `http://eprint.iacr.org/2005/058,2005`.
4. Z. Cheng and R. Comley. Efficient certificateless public key encryption. Available from `http://eprint.iacr.org/2005/012/,2005`.
5. B. Libert and J.-J. Quisquater. On constructing certificateless cryptosystems from identity based encryption. In M. Yung, Y. Dodis, A. Kiayias, and T. Malkin, editors, Public Key Cryptography (PKC 2006). LNCS, Vol. 3958. pp. 474-490. Springer-Verlag, (2006).
6. B. C. Hu, D. S. Wong, Z. Zhang, and X. Deng. Key replacement attack against a generic construction of certificateless signature. In L. M. Batten and R. Safavi-Naini, editors, 11th Australasian Conference on Information Security and Privacy (ACISP 2006). LNCS, Vol. 4058. pp. 235-246. Springer-Verlag, (2006).
7. X. Huang, W. Susilo, Y. Mu, and F. Zhang. On the security of certificateless signature schemes from Asiacrypt 2003. In Y. Li and Y. Mu, editors, Cryptology and Network Security (CANS 2005). LNCS, Vol. 3810. pp. 13-25. Springer-Verlag, (2005).
8. D. H. Yum and P. J. Lee. Generic construction of certificateless signature. In H. Wang, J. Pieprzyk, and V. Varadharajan, editors, 9th Australasian Conference in Information Security and Privacy (ACISP 2004). LNCS, Vol. 3108. pp. 200-211. Springer-Verlag, (2004).
9. Z. Zhang, D. S. Wong, J. Xu, and D. Feng. Certificateless public-key signature: Security model and ecient construction. In J. Zhou, M. Yung, and F. Bao, editors, Applied Cryptography and Network Security. LNCS, Vol 3989. PP. 293-308. Springer-Verlag, 2006.
10. A. W. Dent. A survey of certificateless encryption scheme and security models. Available from `http://eprint.iacr.org/2006/211,2006.`
11. Dent. A. W, Libert. B, Paterson. K. G. Certificateless encryption schemes strongly secure in the standard model. In R. Cramer, editor(s), 11th International Workshop on Practice and Theory in Public-Key Cryptography (PKC 2008). LNCS, Vol. 4939. pp. 344-359. Springer-Verlag, 2008.
12. B. Barak. How to go beyond the black-box simulation barrier. Proceedings of the 42nd Symposium on Foundations of Computer Science. pp. 106-115. IEEE Press, 2001.
13. Guoyan Zhang, Xiaoyun Wang. Certificateless Encryption Scheme Secure in the Standard Model. In Tsinghua Science and Technology, 14(4). pp. 122-127. 2009.