Secure Length-saving ElGamal Encryption under the Computational Diffie-Hellman Assumption

Joonsang Baek, Byoungcheon Lee, and Kwangjo Kim

ICU, 58-4 Hwaam-dong,
Yusong-gu, Taejon, 305-348, S. Korea
{mohi, sultan, kkj}@icu.ac.kr

Abstract. A design of secure and efficient public key encryption schemes under weaker computational assumptions has been regarded as an important and challenging task. As far as the ElGamal-type encryption is concerned, some variants of the original ElGamal encryption scheme whose security depends on weaker computational assumption have been proposed: Though the security of the original ElGamal encryption is based on the decisional Diffie-Hellman assumption (DDH-A), the security of recent schemes such as Pointcheval's ElGamal encryption variant is based on the weaker assumption, the computational Diffie-Hellman assumption (CDH-A). In this paper, we propose a length-saving ElGamal encryption variant whose security is based on CDH-A and analyze its security in the random oracle model. Our scheme is length-efficient and provably secure which provides a shorter ciphertext than that of the Pointcheval's scheme and a formal proof of security against the chosen-ciphertext attack.

1 Introduction

1.1 Encryption Schemes Based on Diffie-Hellman Assumption

Since Diffie and Hellman[9] proposed the concept of public-key cryptosystem, extensive researches has been done in this field. In particular, the public-key encryption scheme proposed by ElGamal[10] has attracted considerable attention. When ElGamal proposed his public-key encryption scheme, it was widely believed that the security of this scheme is based on the computational assumption called "Diffie-Hellman assumption". Roughly speaking, the Diffie-Hellman assumption says that for a cyclic group G, an adversary who sees g^x and g^y cannot efficiently compute g^{xy}. Often, G is defined as a multiplicative group of a large prime modulo p, *i.e.*, Z_p^* where g is a generator and $x, y \in \mathrm{Z}_q$. Note here that q is a large prime such that $q|p-1$.

It may be true that the security of ElGamal encryption scheme depends on the Diffie-Hellman assumption since an adversary attacking this scheme cannot decrypt a ciphertext (g^y, mg^{xy}) of a message m without computing g^{xy}. However, indistinguishability[12], which has been accepted as a general security notion of encryption schemes, does not require the attacker to decrypt the whole message. In the notion of the indistinguishability, security of encryption scheme implies that the adversary cannot distinguish ciphertexts of two messages chosen by himself. Consequently, it seems that the security of ElGamal encryption should depend on some stronger assumption rather than the Diffie-Hellman assumption. In fact, Tsiounis and Yung[14] showed that the security of ElGamal encryption scheme is not based on the Diffie-Hellman assumption but based on the stronger Decisional Diffie-Hellman assumption(DDH-A). DDH-A says that an adversary who sees two distributions (g^x, g^y, g^{xy}) and (g^x, g^y, R), where R is a randomly chosen-string whose length is the same as g^{xy}, cannot distinguish these two distributions. Hence the Diffie-Hellman assumption is often called the computational Diffie-Hellman assumption(CDH-A) for the purpose of emphasizing an adversary's inability to compute the Diffie-Hellman key, g^{xy}. Throughout this paper, we use the term CDH-A to refer to the Diffie-Hellman assumption.

1.2 Chosen Ciphertext Security

Since Zheng and Seberry[15] initiated a full-scale research on adaptive chosen-ciphertext attacks, the design of public-key encryption schemes has trended toward the prevention of these attacks. In the adaptive chosen-ciphertext attack, an adversary is permitted to access a decryption function as well as an encryption function. The adversary may use this decryption function on ciphertexts chosen after obtaining the challenge ciphertext, with the only restriction that the adversary may not ask for the decryption of the challenge ciphertext itself.

Several security notions on the (adaptive or non-adaptive) chosen-ciphertext attack including non-malleability[8] were formalized and the relationship among them was shown in [3]. Public-key encryption schemes secure against the adaptive chosen-ciphertext attack proposed so far include OAEP[5] (based on the RSA function), the Cramer-Shoup scheme[7] (based on the DDH-A), DHAES[1] (based on the hash Diffie-Hellman assumption(HDH-A)), and the Fujisaki-Okamoto(F-O) scheme[11] (based on the security of any semantically secure public-key encryption schemes). More recently, a general method for converting any partially trapdoor one-way function to the public-key encryption scheme that is secure against the chosen-ciphertext attack was proposed by Pointcheval[13].

The Cramer-Shoup scheme is said to be unique since it does not impose any ideal assumption on the underlying hash function as other schemes do. Though the use of the ideal hash function model, *i.e.*, the random oracle model[4], is still controversial[6], this paradigm often yields much more efficient schemes than those in the standard model[2].

We note here that the underlying computational assumption of Cramer-Shoup scheme is DDH-A, which is much stronger than CDH-A, though the random oracle model is not used in the this scheme. The situation remains the same in the ElGamal version of the F-O scheme. However, underlying computational assumption of the ElGamal version of recent Pointcheval's scheme is CDH-A, which is weaker than DDH-A. On the other hand, one deficiency of this scheme is a message expansion: To encrypt a message m, one must compute $(g^{H(m||s)}, rX^{H(m||s)}, G(r) \oplus (m||s))$, where $X(= g^x)$ is a public key, $r \in \mathrm{Z}_p^*$ and

$s \in \mathrm{Z}_q$ are appropriate length of random strings. Here, both G and H are random oracles. Consequently, the length of a ciphertext is 1.5 times longer than that of the original ElGamal version of the F-O scheme. In this paper, we propose another ElGamal encryption variant provably secure against chosen-ciphertext attack in the random oracle model. The underlying computational assumption of our scheme is based on CDH-A, but the length of ciphertext is reduced compared with the Pointcheval's scheme.

The organization of this paper is as follows: We briefly review the notions of chosen-ciphertext security for public-key encryption schemes in Section 2. In Section 3, we describe our proposed scheme and analyze its security. In Section 4, comparison of our scheme with other ElGamal variants is provided and concluding remarks will follow in the final section.

2 Some Preliminaries

2.1 CDH-A with a Random Oracle

Recall that CDH-A implies an adversary's inability to compute g^{xy} seeing g^x and g^y. Though the adversary sees g^x, g^y, and $H(g^{xy})$ where H is a random oracle, he still cannot compute g^{xy} with the same degree as seeing g^x and g^y. Since H is assumed to be a random oracle, $H(g^{xy})$ does not reveal any (partial) information about g^{xy}. Namely, $H(g^{xy})$ does not provide any advantage for computing g^{xy} to the adversary. This equivalent version of CDH-A will be used in the proof in Section 3.

2.2 Notions of Security

Though there are several security notions on the chosen-plaintext and the chosen-ciphertext attacks, we briefly review two notions, the indistinguishability-chosen plaintext attack(IND-CPA) [3, 12] and the plaintext awareness(PA)[3, 5].

Security against the chosen-plaintext attack for public-key encryption schemes is defined by using the following experiment: Let $\mathcal{A}$ be an adversary with two algorithms A_1 and A_2. The "find"-stage algorithm A_1 is run on the public key, *pk*. At the end of A_1's execution, it outputs a triple (m_0, m_1, s) where m_0 and m_1 are messages that have the same length and s is a state information. Then one of m_0 and m_1 is selected at random and ciphertext y is determined by encrypting m_b ($b \in_R \{0, 1\}$) under *pk*. The job of the "guess"-stage algorithm A_2 is to determine if y was selected as the encryption of m_0 or m_1, namely to determine the bit b. If a probability that A_2 outputs b is negligible, we say that the public-key encryption scheme is secure in the sense of IND-CPA. Now, we formally define this experiment as follows:

Definition 1 (IND-CPA). *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a public-key encryption scheme, where $\mathcal{K}$, $\mathcal{E}$,and $\mathcal{D}$ denote a key generation algorithm, encryption algorithm, and decryption algorithm, respectively. Let $\mathcal{A}(A_1, A_2)$ be an adversary where A_1 denotes a "find"-stage algorithm and A_2 denotes a "guess"-stage algorithm. Also,* *let (sk, pk) be a secret and public key pair and let s be a state information. If the advantage of $\mathcal{A}$*

$$Adv^{\mathrm{IND-CPA}}_{\mathcal{A},\Pi} = 2 \cdot [(sk, pk) \leftarrow \mathcal{K}; (m_0, m_1, s) \leftarrow A_1(\mathsf{find}, s); b \leftarrow \{0,1\}; \\ y \leftarrow \mathcal{E}_{pk}(m_b) : A_2(\mathsf{guess}, pk, s, y) = b] - 1$$

is negligible, we say that Π is secure in the sense of IND-CPA.

The plaintext awareness(PA), first defined by Bellare and Rogaway[5], formalizes an adversary's inability to create the ciphertext y without "knowing" its corresponding plaintext x.

We note that PA has only been defined in the random oracle model. An adversary $\mathcal{B}$ for PA is given a public key pk and access to the random oracle H. We also provide $\mathcal{B}$ with an oracle for $\mathcal{E}^H_{pk}$. The adversary outputs a ciphertext y. To be PA, the adversary $\mathcal{B}$ should necessarily know the decryption m of its output. To formalize this, it is required that there exists an algorithm K (knowledge extractor) that could have output m just by looking at the public key, $\mathcal{B}$'s H-queries and the answers to them, and the answers to $\mathcal{B}$'s queries to $\mathcal{E}^H_{pk}$. The following is a formal definition of PA.

Definition 2 (PA). *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a public-key encryption scheme, let $\mathcal{B}$ be an adversary, let $hH = \{(h_1, H_1), (h_2, H_2), \ldots, (h_{q_H}, H_{q_H})\}$ be a list of all of $\mathcal{B}$'s oracle queries, $h_1, h_2, \ldots, h_{q_H}$, and the corresponding answers $H_1, H_2, \ldots,$ H_{q_H}, and let K be a knowledge extractor. Let $C = \{y_1, y_2, \ldots, y_{q_H}\}$ denote the answers(ciphertexts) as a result of $\mathcal{E}^H_{pk}$-queries. For any $k \in \mathsf{N}$ define*

$$Succ^{\mathrm{PA}}_{K,\mathcal{B},\Pi} = \Pr[H \leftarrow \mathsf{Hash}; (pk, sk) \leftarrow \mathcal{K}; (hH, C, y) \leftarrow runB^{H, \mathcal{E}^H_{pk}(pk)} : \\ K(hH, C, y, pk) = D^H_{sk}(y)].$$

For $y \notin C$, we say that K is a $\lambda(k)$-extractor if K has running time polynomial in the length of its inputs and for every $\mathcal{B}$, $Succ^{PA}_{K,\mathcal{B},\Pi} \geq \lambda(k)$. We say that Π is secure in the sense of PA if Π is secure in the sense of IND-CPA and there exists a $\lambda(k)$-extractor K where $1 - \lambda(k)$ is negligible.

3 Secure Length-saving ElGamal Encryption Variant

3.1 Description of Our Scheme

To provide security against the chosen-plaintext attack under CDH-A, we apply a random oracle G to the Diffie-Hellman key $X^{H(m||s)}$. Also, to provide PA, we apply another random oracle H to message m concatenated by some random string s. A concrete description of our scheme Π is as follows:

Secure Length-saving ElGamal Encryption Variant $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$

- Key generator $\mathcal{K}$
  $(pk, sk) \leftarrow \mathcal{K}(1^k)$, $pk = (p, q, g, X(= g^x))$ and $sk = (p, q, g, x)$ where $x \in_R \mathrm{Z}_q$, $|p| = k = k_0 + k_1$, and $q|p-1$, a large prime number.

- Hash Function (two random oracles)
  $H : \{0,1\}^k \to \mathrm{Z}_q$, and $G : \mathrm{Z}_p^* \to \{0,1\}^k$

- Encryption $\mathcal{E}$
  $\mathcal{E}_{pk}(m, s) = (\alpha, \beta) = (g^{H(m||s)}, G(X^{H(m||s)} \bmod p) \oplus (m||s))$, where message $m \in \{0,1\}^{k_0}$ and $s \leftarrow_R \{0,1\}^{k_1}$

- Decryption $\mathcal{D}$

$$\mathcal{D}_{sk}(\alpha, \beta) = \begin{cases} [\beta \oplus G(\alpha^x \bmod p)]^{k_0} & \text{if } \alpha = g^{H(\beta \oplus G(\alpha^x \bmod p))} \\ \varepsilon(\text{null}) & \text{otherwise} \end{cases}$$

  where $[\beta \oplus G(\alpha^x \bmod p)]^{k_0}$ denotes the first k_0 bit of $[\beta \oplus G(\alpha^x \bmod p)]$.

3.2 Security Analysis

In this section, we show that our ElGamal encryption variant is secure in the sense of IND-CPA under CDH-A and there exists a knowledge extractor K. Note that the security in the sense of IND-CPA and the existence of a knowledge extractor mean the security in the sense of PA. By the result of [3], this implies security against the adaptive chosen-ciphertext attack(IND-CCA2)

Theorem 1. *If there exists an adversary attacking the encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ in a chosen-plaintext scenario, we can construct an adversary that breaks CDH-A in the random oracle model with non-negligible probability.*

Proof. Let $\mathcal{A} = (A_1, A_2)$ be an adversary attacking $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ in a chosen-plaintext scenario and ϵ be an advantage of $\mathcal{A}$. Recall that A_1 denotes the "find"-stage algorithm and A_2 denotes the "guess"-stage algorithm. Assume that both G and H are random oracles. Our proving strategy is to use $\mathcal{A}$ to construct an adversary $\mathcal{B}$ that breaks CDH-A. Suppose that $X(= g^x)$, $Y(= g^y)$ and $T(= G(X^y))$ are given to $\mathcal{B}$. $\mathcal{B}$ works as follows:

- Run $\mathcal{A}$. When A_1 makes oracle query j to G, $\mathcal{B}$ chooses a random string in $\{0,1\}^k$ and answers it as $G(j)$. Similarly, if A_1 makes oracle query j to H, $\mathcal{B}$ chooses a random string in Z_q and answers it as $H(j)$. $\mathcal{A}$ finally outputs two messages m_0 and m_1. Then $\mathcal{B}$ selects $b \in \{0,1\}$ at random and outputs X as a public key and $(\alpha, \beta) = (Y, T \oplus (m_b||s))$ as a ciphertext.
- The ciphertext (α, β) is inputted to A_2. Then, A_2 makes oracles queries as A_1 did.
- When A_2 returns its answer $d \in \{0,1\}$, $\mathcal{B}$ returns a set of all the oracle queries to G.

Let $Succ\mathcal{A}$ be an event that A_2 correctly guesses the bit b, *i.e.*, outputs $d = b$. Then $\Pr[Succ\mathcal{A}] = \frac{1}{2} + \frac{\epsilon}{2}$ by definition. Now we define the following two events. Let $AskG$ be an event that the query X^y was made to G and let $AskH$ be an event that a query $m||s$ for some message m and s chosen at the beginning by $\mathcal{B}$, is made to H. Now, define $H(m_b||s)$ as y and $T(= G(X^y))$ as $\beta \oplus (m_b||s)$.

Now, let us scrutinize $\mathcal{A}$'s advantage ϵ. Recall that $\mathcal{A}$ is given $(Y, T \oplus (m_b||s))$. If the query asked to H is $(m_b||s)$, he will succeed. Also, if the query asked to G is X^y, he will also succeed. That is, $\mathcal{A}$'s advantage ϵ depends on the event $AskG$ or $AskH$. Hence we get

$$\frac{1}{2} + \frac{\epsilon}{2} = \frac{1}{2} + \frac{\Pr[AskG \vee AskH]}{2}.$$

Furthermore,

$$\begin{aligned}\Pr[AskG \vee AskH] &\leq \Pr[AskG] + \Pr[AskH] \\ &= \Pr[AskG] + \Pr[AskH|AskG]\Pr[AskG] \\ &\quad +\Pr[AskH|\neg AskG]\Pr[\neg AskG] \\ &\leq 2 \cdot \Pr[AskG] + \Pr[AskH|\neg AskG]\end{aligned}$$

Yet, the probability that the event $AskH$ takes place is very small provided that $\neg AskG$ is true. More precisely,

$$\Pr[AskH|\neg AskG] \leq \frac{q_H}{2^{k_1}}.$$

Therefore, we have

$$\Pr[AskG] \geq \frac{\epsilon}{2} - \frac{q_H}{2^{k_1+1}}.$$

This implies that the probability that X^y lies in the set of all the oracle queries to G is greater than $\frac{\epsilon}{2} - \frac{q_H}{2^{k_1+1}}$. Hence if the advantage ϵ of $\mathcal{A}$ is non-negligible, $\mathcal{B}$ breaks CDH-A with non-negligible probability. □

Now we construct a knowledge extractor K. Note that the existence of K implies security in the sense of PA under the assumption that Π is secure in the sense of IND-CPA.

Theorem 2. *Let $\mathcal{B}$ be an adversary for PA. Then there exists a knowledge $\lambda(k)$-extractor K and hence $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is secure in the sense of PA.*

Proof. Since we have shown that Π is secure in the sense of IND-CPA, we only need to construct a knowledge-extractor K. Assume that $gG = \{(g_1, G_1), (g_2, G_2), \ldots, (g_{q_G}, G_{q_G})\}$, $hH = \{(h_1, H_1), (h_2, H_2), \ldots, (h_{q_H}, H_{q_H})\}$(all the random oracle query-answer pairs of $\mathcal{B}$), $C = \{y_1, y_2, \ldots, y_E\}$(a set of ciphertexts that $\mathcal{B}$ has obtained from the interaction with the random oracles and the encryption oracle), $z = (\alpha, \beta) \notin C$ (a ciphertext produced by $\mathcal{B}$ which is not in C), and the public key X are given to K. It works as follows:

- For all gG and hH, K checks whether there exists a pair (g_q, h_r) such that $z = (\alpha, \beta) = (g^{H_r}, G_q \oplus h_r)$.
- If there exists a pair, K returns $m = [h_r]^{k_0}$ and s. Otherwise, outputs ε(null).

Next we think of the probability that K outputs the plaintext m correctly, namely $m = \mathcal{D}_{sk}(y)$. Let $Fail$ be an event that $m \neq \mathcal{D}'_{sk}(y)$ and let $AskG$ be an event that there exists a pair (g_q, G_q) in the list gG such that $z = (\alpha, \beta) = (g^{H_r}, G_q \oplus h_r)$. Similarly, let $AskH$ denote an event that there exists a pair (h_r, H_r) in the list hH such that $z = (\alpha, \beta) = (g^{H_r}, G_q \oplus h_r)$. Then,

$$\begin{aligned}\Pr[Fail] &= \Pr[Fail|AskG \wedge AskH]\Pr[AskG \wedge AskH] + \\ &\quad \Pr[Fail|\neg AskG \vee \neg AskH]\Pr[\neg AskG \vee \neg AskH] \\ &\leq 0 + \Pr[Fail|\neg AskG \vee \neg AskH].\end{aligned}$$

We now determine the upper bound of $\Pr[Fail|\neg AskG \vee \neg AskH]$. For *valid* y, there exists h and g such that $z = (g^{H(h)}, G(g) \oplus h)$. As $y \notin C$, it follows that $h \neq \mathcal{D}(y_i)$ for every $y_i \in C$. However,

$$\begin{aligned}\Pr[valid|\neg AskG \vee \neg AskH] &= \frac{\Pr[valid \wedge (\neg AskG \vee \neg AskH)]}{\Pr[(\neg AskG \vee \neg AskH)]} \\ &\leq \frac{\Pr[valid \wedge \neg AskH]}{\Pr[\neg AskH]} \\ &+ \frac{\Pr[valid \wedge \neg AskG \wedge \neg AskH]}{\Pr[\neg AskG]} \\ &\leq \Pr[valid|\neg AskH] + \Pr[valid|\neg AskG] \\ &\leq \frac{1}{2^k} + \frac{1}{2^{k_1}}.\end{aligned}$$

On the other hand, if $\neg AskG$ or $\neg AskH$ is true, from the construction of K, it always outputs ε(null), i.e., y is *invalid*. This means that $\Pr[Fail|\neg AskG \vee \neg AskH] = \Pr[valid|\neg AskG \vee \neg AskH] \leq \frac{1}{2^k} + \frac{1}{2^{k_1}}$. Namely, the probability of rejection of valid ciphertext is upper-bounded by $1/2^k + 1/2^{k_1}$. Consequently,

$$\lambda(k) = 1 - \Pr[Fail] \geq 1 - \frac{1}{2^k} - \frac{1}{2^{k_1}}.$$

□

As mentioned before, we get the following corollary from Theorems 1 and 2.

Corollary 1. *Our scheme is secure in the sense of IND-CCA2.*

4 Comparison with Other Schemes

We compare the length of ciphertext of our scheme with the original ElGamal encryption scheme and other ElGamal-type encryption schemes such as ElGamal encryption variant of the F-O scheme, and the Pointcheval's ElGamal encryption variant.

For comparison, we briefly describe how four schemes encrypt a message m.

- ElGamal scheme : $(g^y, X^y m)$
- F-O scheme : $(g^{H(m||s)}, X^{H(m||s)} \oplus (m||s))$
- Pointcheval's scheme : $(g^{H(m||s)}, X^{H(m||s)} r, G(r) \oplus (m||s))$
- Our scheme : $(g^{H(m||s)}, G(X^{H(m||s)}) \oplus (m||s))$

We summarize the cryptographic characteristics of four schemes in Table 1.

	ElGamal	F-O	Pointcheval	Our scheme
Length	$2k$	$2k$	$3k$	$2k$
Number of ROs	None	1	2	2
Assumption	DDH-A	DDH-A	CDH-A	CDH-A
Security	IND-CPA	IND-CCA2	IND-CCA2	IND-CCA2
Comp. for Enc.	2E	2E+H	2E+2H	2E+2H
Comp. for Dec.	E	2E+H	2E+2H	2E+2H

Table 1. Comparison with Other ElGamal Variants, where: $k = |\mathbb{Z}_p^*|$, RO = Random Oracle, E= Exponentiation, H= Random oracle computation, Comp. for Enc.= Computation for Encryption, Comp. for Dec.=Computation for Decryption

As can be seen from the table, our scheme guarantees sound security and length-efficiency: Under the CDH-A, it is secure in the sense of IND-CCA2. Now we explain more on the length of a ciphertext. In the F-O scheme, the length of a ciphertext is $2k$ where $k = |\mathbb{Z}_p^*|$. A ciphertext of our scheme has the same length as those of the original ElGamal scheme and the F-O scheme when the length of output of G, which is used as the random oracle, is set to k. In the Pointcheval's scheme, the length of ciphertext is extended to $3k$. Compared with the Pointcheval's scheme, our scheme effectively saves the length of a ciphertext under the same circumstances - the security of both schemes is based on CDH-A and two random oracles are used. Note that the message per ciphertext ratio of the original ElGamal scheme is the biggest since no additional random string follows the message m being encrypted. However, as already known, the original ElGamal scheme is insecure against chosen-ciphertext attack. The message per ciphertext ratios of other three schemes are the same.

For computational efficiency, the computation cost required in our scheme to encrypt and decrypt messages is estimated to be the same as that of the Pointcheval's scheme. We omit the computation required to generate public key which can be done previously.

Finally, we mention about implementation of the random oracle G. To implement this function, one can use the heuristic method described in [4] and [5] as follows:

$$G(X^y) = g(\langle 0 \rangle, X^y) || g(\langle 1 \rangle, X^y) || g(\langle 2 \rangle, X^y) || \ldots,$$

where g is an efficient cryptographic hash function such as SHA-1 or MD5 which outputs 160 bits or 128 bits respectively and the notation $\langle i \rangle$ denotes a binary 32-bit word encoding of integer i.

5 Concluding Remarks

We have proposed another variant of the ElGamal encryption scheme. The security of our scheme depends on CDH-A which is much weaker than DDH-A. Moreover, the length of a ciphertext is reduced compared with the recent Pointcheval's ElGamal variant which is based on CDH-A. Also, our scheme provides the same degree of computational efficiency as other proposed schemes.

However, as done in other practical schemes, the random oracle model is employed to provide provable security. A construction of "practical" public-key encryption schemes secure against active adversaries without random oracle is, of course, an interesting and meaningful challenge.

Acknowledgements

The authors would like to thank the anonymous referees for their helpful comments on preparation for the final version of our paper.

References

1. M. Abdalla, M. Bellare, and P.Rogaway, "DHAES: An Encryption Scheme Based on Diffie-Hellman Problem", *IEEE P1363a Submission*, 1998, Available at `http://grouper.ieee.org/groups/1363/addendum.html`.
2. M. Bellare, "Practice-Oriented Provable-Security", *In the First International Workshop on Information Security - Proceedings of ISW'97, LNCS 1396*, Springer-Verlag, 1998.
3. M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway, "Relations Among Notions of Security for Public-Key Encryption Schemes", *In Advances in Cryptology - Proceedings of Crypto'98, LNCS 1462*, pp.26-45, Springer-Verlag, 1998.
4. M. Bellare and P. Rogaway, "Random Oracles are Practical : A Paradigm for Designing Efficient Protocols", *ACM Conference on Computer and Communications Security*, pp.62-73, 1993.
5. M. Bellare and P. Rogaway, "Optimal Asymmetric Encryption - How to Encrypt with RSA", *In Advances in Cryptology - Proceedings of Eurocrypt'94, LNCS 950*, pp.92-111, Springer-Verlag, 1995.
6. R. Canetti, O. Goldreich, and S. Halevi, "The Random Oracle Methodology, Revisited", *Proceedings of the 30th Annual Symposium on the Theory of Computing, ACM*, 1998.
7. R. Cramer and V. Shoup, "A Practical Public Key Cryptosystem Provably Secure Against Adaptive Chosen Ciphertext Attack", *In Advances in Cryptology - Proceedings of Crypto'98, LNCS 1462*, pp. 13-25, Springer-Verlag, 1998.
8. D. Dolev, C. Dwork, and M. Naor, "Non-Malleable Cryptography", *Proceedings of 23rd STOC.* , ACM Press 1991.

9. W. Diffie and M.Hellman, “New Directions in Cryptography”, *IEEE Transactions on Information Theory, IT-22(6)*, pp.644-654, 1976.
10. T. ElGamal, “A Public Key Cryptosystems and a Signature Schems Based on Discrete Logarithms”, *IEEE Transactions on Information Theory, IT-31(4)*, pp.469-472, 1985.
11. E. Fujisaki and T. Okamoto, “How to Enhance the Security of Public-Key Encryption at Minimum Cost”, *PKC’99, LNCS 1560*, pp.53-68, Springer-Verlag, 1999.
12. S. Goldwasser and S. Micali, “A Probabilistic Encryption”, *Journal of Computer and System Sciences, 28*, pp. 270-299, 1984.
13. D. Pointcheval, “Chosen-Ciphertext Security for any One-Way Cryptosystem”, *PKC’2000, LNCS 1751*, pp.129-146, Springer-Verlag, 2000.
14. Y. Tsiounis and M. Yung, “On the Security of ElGamal Based Encryption”, *PKC’98, LNCS 1431*, pp.117-134, Springer-Verlag, 1998.
15. Y. Zheng and J. Seberry. “Practical Approaches to Attaining Security Against Adaptively Chosen Ciphertext Attacks”, *In Advances in Cryptology - Proceedings of Crypto’92, LNCS 740*, pp. 292-304, Springer-Verlag, 1993.