Provably Secure Length-saving Public-Key Encryption Scheme under the Computational Diffie-Hellman Assumption*

Joonsang Baek†, Byoungcheon Lee‡, and Kwangjo Kim‡

† SECUi.COM, ‡ Information and Communications University (ICU)

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 ElGamal-type encryption schemes are concerned, some variants of the original ElGamal encryption scheme based on the weaker computational assumption have been proposed. For instance, security of the ElGamal variant of Fujisaki-Okamoto public-key encryption scheme and Cramer and Shoup's encryption scheme is based on the decisional Diffie-Hellman assumption (DDH-A). However, security of the recent scheme, such as Pointcheval's ElGamal encryption variant, is based on the computational Diffie-Hellman assumption (CDH-A), which is weaker than DDH-A.

In this paper, we propose new ElGamal encryption variants whose security is based on CDH-A and EC-CDH-A (the elliptic curve computational Diffie-Hellman assumption). Also, we show that the proposed variants are secure against the adaptive chosen-ciphertext attack in the random oracle model. An important feature of the proposed variants is a length-efficiency which provide shorter ciphertexts than those of other proposed schemes.

1 Introduction

1.1 Encryption Schemes Based on Diffie-Hellman Assumption

Ever since Diffie and Hellman [10] originally proposed the concept of public-key cryptosystem, extensive research has been performed in this field. In particular, the public-key encryption scheme proposed by ElGamal [11] has attracted considerable attention. When ElGamal

*This paper will appear at ETRI J. Dec.,2000.

proposed his public-key encryption scheme, it was widely believed that the security of this scheme is based on the "computational Diffie-Hellman assumption".

Roughly speaking, the computational Diffie-Hellman assumption says that for a cyclic group G, an adversary who sees g^x and g^y cannot efficiently compute g^{xy}. In this paper, we assume that the G is defined as the multiplicative group of a finite field modulo a large prime p, *i.e.*, $\mathbf{Z}_p^*$ where g is a generator for a subgroup $\mathbf{Z}_q$ of $\mathbf{Z}_p^*$ and $x, y \in \mathbf{Z}_q$. Note here that q is a large prime such that $q|p-1$.

It is true that the security of ElGamal encryption scheme is based on the computational Diffie-Hellman assumption in a passive attack model, where an adversary cannot decrypt a ciphertext (g^y, mg^{xy}) of a message m without computing g^{xy}. However, indistinguishability [14], which has been accepted as a general security notion of encryption schemes, does not require an attacker to decrypt the whole message. In the notion of indistinguishability, security of encryption scheme implies that an adversary cannot distinguish ciphertexts of two chosen messages. Consequently, it seems that the security of ElGamal encryption should depend on some stronger assumption rather than the computational Diffie-Hellman assumption. In fact, Tsiounis and Yung [16] have shown 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, g^z), where z is a randomly chosen from $\mathbf{Z}_q$ and the length of g^z is the same as that of 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 will use the term CDH-A to refer to the Diffie-Hellman assumption.

1.2 Chosen Ciphertext Security

Ever since Zheng and Seberry [17] initiated a full-scale research on the adaptive chosen-ciphertext attack, 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 before and 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 [9] were formalized and the relationship among them has been 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 [8] (based on DDH-A), DHAES [1] (based on the hash Diffie-Hellman assumption (HDH-A)), and the Fujisaki-Okamoto(F-O) scheme [12] (based on the security of any semantically secure public-key encryption schemes against chosen-plaintext attacks and therefore DDH-A). Fujisaki and Okamoto [13] also proposed a generic method that converts symmetric and asymmetric encryption schemes into an asymmetric encryption scheme secure against the adaptive chosen-ciphertext attack. More recently, Pointcheval [15] proposed a general method for converting any partially trapdoor one-way function to the public-key encryption scheme which is provably secure against the chosen-ciphertext attack. Both works are very similar and provide schemes against the adaptive chosen-ciphertext attack under CDH-A.

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. Although 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 believed to be stronger than CDH-A, even though the random oracle model is not used in this scheme. The situation remains the same in the ElGamal version of the first F-O scheme. However, the ElGamal variant of recent Pointcheval's scheme and Fujisaki and Okamoto's ElGamal variant using the integration of asymmetric and symmetric encryptions are based on CDH-A. On the other hand, compared to the original ElGamal scheme, these schemes have a disadvantage in a sense that the length of the ciphertext is expanded.

Based on aforementioned discussions, we propose another ElGamal encryption variant provably secure against chosen-ciphertext attack in the random oracle model and its elliptic curve version. The underlying computational assumption of the proposed schemes are based on CDH-A and EC-CDH-A, but the length of the ciphertext is shorter than those of other schemes based on CDH-A.

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 the proposed schemes and analyze their security. In Section 4, comparison of the proposed scheme with other ElGamal variants is provided and concluding remarks will follow in the final section.

2 Some Preliminaries

In this section, we briefly review the concepts of the "indistinguishability-chosen plaintext attack (IND-CPA)"[3] and the "plaintext awareness (PA)"[3].

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 3-tuple (m_0, m_1, s) where m_0 and m_1 are messages of 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 the advantage 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 space, encryption and decryption algorithms, 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 $\mathcal{A}$*

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

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 $\mathcal{E}_{pk}^H$. 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 their answers, and the answers to $\mathcal{B}$'s queries to $\mathcal{E}_{pk}^H$. 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 $\mathcal{E}_{pk}^H$-queries. For any $k \in \mathsf{N}$ define*

$$\begin{aligned} Succ_{K,\mathcal{B},\Pi}^{\mathrm{PA}} &= \Pr[H \leftarrow \mathsf{Hash}; (pk, sk) \leftarrow \mathcal{K}; (hH, C, y) \leftarrow runB^{H, \mathcal{E}_{pk}^H}(pk) : \\ &\quad K(hH, C, y, pk) = D_{sk}^H(y)]. \end{aligned}$$

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 $\mathcal{B}$, $Succ_{K,\mathcal{B},\Pi}^{PA} \geq \lambda(k)$. 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 Description of the Proposed Schemes

3.1 Multiplicative Group Variant

Our motivation for constructing the public-key encryption scheme whose security relies on CDH-A is to apply random oracle G to Diffie-Hellman key g^{xy}. Since G is assumed to be a random oracle, $G(g^{xy})$ does not reveal any (partial) information about g^{xy}. Hence, to gain any advantage, the adversary must compute g^{xy}. Also, to provide PA, we apply another random oracle H to message m concatenated by some random string s. This motivation leads to the proofs for the theorems provided later in this section. A concrete description of the proposed scheme Π_1 is the following (Note that $\oplus$ means bit-wise exclusive-OR throughout this paper.):

Finite Multiplicative Group Variant $\Pi_1 = (\mathcal{K}, \mathcal{E}, \mathcal{D})$

- Key generator $\mathcal{K}$
  - Choose a finite multiplicative group $\mathbf{Z}_p^*$. Let q be a large prime number dividing $p-1$ and let g be an element of order q in $\mathbf{Z}_p^*$.
  - $pk = (p, q, g, X(= g^x))$ and $sk = (p, q, g, x)$ where $x \in_R \mathbf{Z}_q$ and $|p| = k = k_0 + k_1$.
- Hash Function (two random oracles)
  - Choose $H : \{0,1\}^k \to \mathbf{Z}_q$, and $G : \mathbf{Z}_p^* \to \{0,1\}^k$.
- Encryption $\mathcal{E}$
  - Compute $r = g^t$ and $l = X^t$ where $t = H(m||s)$, message $m \in \{0,1\}^{k_0}$, and $s \leftarrow_R \{0,1\}^{k_1}$.
  - Compute $\mathcal{E}_{pk}(m, s) = (\alpha, \beta) = (r, G(l) \oplus (m||s))$, where message $m \in \{0,1\}^{k_0}$ and $s \leftarrow_R \{0,1\}^{k_1}$.
- Decryption $\mathcal{D}$
  - Compute $l' = \alpha^x$ and $t' = H(\beta \oplus G(l'))$.
  - If $\alpha = g^{t'}$, output $\mathcal{D}_{sk}(\alpha, \beta) = [\beta \oplus G(l')]^{k_0}$. Otherwise, output "null". Here, $[\beta \oplus G(l')]^{k_0}$ denotes the first k_0 bits of $[\beta \oplus G(l')]$.

3.2 Elliptic Curve Variant

EC-CDH-A (elliptic curve computational Diffie-Hellman assumption) is similarly defined as CDH-A. EC-CDH-A says that for a finite group G' of points on elliptic curve E, an adversary who sees aP and bP cannot efficiently compute abP. Often, E is defined on a Galois field of characteristic 2 or a prime number. Here, P is a point of order q on E, where q is a large prime such that $q|\#G'$ (the order of G'). The following description assumes that the defining field of E is a Galois field of characteristic a prime number p. Note that in order to obtain more computational efficiency using the particular scalar multiplication method such as Frobenius expansion described in [7], the defining field can be altered.

Elliptic Curve Variant $\Pi_2 = (\mathcal{K}, \mathcal{E}, \mathcal{D})$

- Key generator $\mathcal{K}$
  - Choose a non-supersingular elliptic curve defined on Galois field $GF(p)$, $E(GF(p))$, and calculate the order of $E(GF(p))$, $\#E(GF(p))$. Let q be a large prime number dividing $\#E(GF(p))$ and let P be a point of order q on $E(GF(p))$.
  - $pk = (E, P, q, W(= uP))$ and $sk = (E, P, q, u)$ where $u \in_R GF(q)$ and $|p| = k = k_0 + k_1$.
- Hash Function (two random oracles)
  - Choose $H : \{0,1\}^k \to GF(q)$, and $G : GF(p) \to \{0,1\}^k$.
- Encryption $\mathcal{E}$
  - Compute $R = tP$ and $S = tW$ where $t = H(m||s)$, message $m \in \{0,1\}^{k_0}$, and $s \leftarrow_R \{0,1\}^{k_1}$.
  - $\mathcal{E}_{pk}(m, s) = (A, B) = (R, G(x_S) \oplus (m||s))$ where x_S is the x-coordinate of S.
- Decryption $\mathcal{D}$
  - Compute $S' = uA$ and $t' = H(B \oplus G(x_{S'}))$.
  - If $A = t'P$, output $\mathcal{D}_{sk}(A, B) = [B \oplus G(x_{S'})]^{k_0}$. Otherwise, output "null". Here, $x_{S'}$ denotes the x-coordinate of S' and $[B \oplus G(x_{S'})]^{k_0}$ denotes the first k_0 bits of $[B \oplus G(x_{S'})]$.

3.3 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 imply 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 $\Pi_1 = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ in a chosen-plaintext scenario, then 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 $\Pi_1 = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ 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. Let q_G and q_H denote the numbers of queries to G and H, respectively. Our proving strategy is to use $\mathcal{A}$ to construct an adversary $\mathcal{B}$ that breaks CDH-A. Suppose that $X(= g^x)$ and $Y(= g^y)$ are given to $\mathcal{B}$. $\mathcal{B}$ performs the following game:

- First give X, as a public key, to $\mathcal{A}$ and then run $\mathcal{A}$. When A_1 makes any new 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 any new oracle query j to H, $\mathcal{B}$ chooses a random string in Z_q and answers it as $H(j)$. A_1 finally outputs two messages m_0 and m_1. $\mathcal{B}$ then selects $b \in \{0,1\}$ at random, takes a random string T in $\{0,1\}^k$ for $G(X^y)$, and outputs $(\alpha, \beta) = (Y, T \oplus (m_b||s))$ as a ciphertext.
- The ciphertext (α, β) is provided as an input to A_2. If A_2 makes oracle queries, $\mathcal{B}$ answers as above and A_2 outputs its answer $d \in \{0,1\}$.
- $\mathcal{B}$ chooses $Q \in_R [1, q_G]$ and stops the game at the Q-th query (without waiting d output by A_2) hoping that X^y has been asked to G. Then, $\mathcal{B}$ outputs this query.

Now let us define the following two events, $AskG$ and $AskH$.

- $AskG$: The query X^y was made to G.
- $AskH$: The query $(m||s)$ for some messages m and s chosen at the beginning by $\mathcal{B}$, is made to H.

We say that the adversary $\mathcal{A}$ wins the game if some of above events occur. Let Adv denote the advantage of the adversary according to the game described above.

Thanks to a random simulation of G and H, this game perfectly simulates the real attack of $\mathcal{A}$ except the case where $AskG$ or $AskH$ occurs. But this case makes the adversary win in our game, therefore, $Adv \geq Adv_{\mathcal{A}} = \epsilon$. However, since the adversary gains no advantage neither $AskG$ nor $AskH$, we obtain $Adv \leq \Pr[AskG \vee AskH]$. This leads to $\epsilon \leq \Pr[AskG \vee AskH]$.

Furthermore,

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

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

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

With probability $1/q_G$, the Q-th query to G is X^y, *i.e.*, the probability that X^y is asked to G at the Q-th query is lower-bounded by $(1/q_G)(\epsilon - q_H/2^{k_1})$. 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_1 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 $\Pi_1 = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is secure in the sense of PA.*

Proof: Since we have shown that Π_1 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), $y = (\alpha, \beta) \notin C$ (a ciphertext produced by $\mathcal{B}$ which is not in C), and the public key X are given to K. The knowledge extractor K works as follows:

- K considers all the query-answer pairs gG and hH, respectively, and checks that there exist pairs (g_u, G_u) and (h_v, H_v) such that $y = (\alpha, \beta) = (g^{H_v}, G_u \oplus h_v)$ and $g_u = X^{H_v}$.
- At most one $\{(g_u, G_u), (h_v, H_v)\}$ may satisfy $\alpha = g^{H_v}$, $\beta = G_u \oplus h_v$, and $g_u = X^{H_v}$. If there exists such pairs, K returns $m = [h_v]^{k_0}$ and s. Otherwise, outputs ε (null). (The ciphertext is considered as an invalid one and therefore be rejected.)

With this simulation, only valid ciphertext will be decrypted. However, there is a possibility that a valid ciphertext can be produced without asking queries to both G and H. But, at most one value for $H(m||s)$ can be accepted since the encryption function Π_1 is an injection. Then,

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

Here, $AskG$ denotes an event that there exists a pair (g_u, G_u) in the list gG such that $y = (\alpha, \beta) = (g^{H_v}, G_u \oplus h_v)$ for some (h_v, H_v) in the list hH. Similarly, $AskH$ is an event that there exists a pair (h_v, H_v) in the list hH such that $y = (\alpha, \beta) = (g^{H_v}, G_u \oplus h_v)$ for some (g_u, G_u) in the list gG.

Hence, the probability of wrong decryption (rejection of valid ciphertext) is upper-bounded by $1/q + 1/2^k$. Therefore, the probability of no wrong decryption, *i.e.*, $1 - \Pr[Fail]$ is given by

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

¶

4 Comparison with Other Schemes

We compare the length of ciphertext of the proposed scheme with the original ElGamal encryption scheme and other ElGamal-type encryption schemes such as the ElGamal encryption variant of the F-O scheme, and the Pointcheval's scheme. We assume that all the encryption schemes in this section are defined in the finite mulitiplicative group $\mathbf{Z}_p^*$. Note that all the schemes discussed in this section provide a shorter ciphertext length if elliptic curves are employed.

Before 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 : $(g^{H(m||s)}, X^{H(m||s)} r, G(r) \oplus (m||s))$
- The proposed 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	Proposed 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: $k = |p|$ (the length of the prime number 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, the proposed scheme guarantees sound security and length-efficiency. Under the CDH-A, it is secure in the sense of IND-CCA2. We now provide a more detailed explanation on the length of a ciphertext. In the F-O scheme, the length of a ciphertext is $2k$. A ciphertext of the proposed 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 expanded to $3k$. Compared with the Pointcheval's scheme, the proposed scheme effectively reduces the length of a ciphertext under the same circumstances, *i.e.*, the security of both schemes is based on CDH-A and two random oracles are used. Note that the message to one ciphertext ratio (a measure for how many lengths of plaintext can be encrypted per a ciphertext) the original ElGamal scheme is the largest since no additional random string follows the message m being encrypted. However, as widely known, the original ElGamal scheme is insecure against chosen-ciphertext attack. Note that the message to ciphertext ratios of other three schemes are the same.

As also can be seen from the table, the computation cost required in the proposed scheme to encrypt and decrypt messages is estimated to be the same as that of the Pointcheval's scheme. Note that we have omitted the computation required to generate public key.

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:

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

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

In this paper, we have proposed another ElGamal encryption variant whose security is based on CDH-A and its elliptic curve version whose security is based on EC-CDH-A, both of which are much weaker than DDH-A and EC-DDH-A (the elliptic curve decisional Diffie-Hellman assumption), respectively. Moreover, the lengths of a ciphertext of the proposed scheme is reduced compared with the recent Pointcheval's ElGamal variant, which is based on CDH-A. Also, the proposed 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 other than the one in [8] is an interesting and meaningful future work.

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] J. Cheon, S. Park, C. Park and S. Hahn, "Scalar multiplication on elliptic curves by Frobenius expansions", *ETRI J., Vol.21, No.1*, pp. 27-38, March 1999.

[8] R. Cramer and V. Shoup, "A practical public key cryptosystem provably secure sgainst adaptive chosen ciphertext attack", *In Advances in Cryptology - Proceedings of Crypto'98, LNCS 1462*, pp. 13-25, Springer-Verlag, 1998.

[9] D. Dolev, C. Dwork, and M. Naor, "Non-malleable cryptography", *Proceedings of 23rd STOC.* , ACM Press 1991.

[10] W. Diffie and M.Hellman, "New directions in cryptography", *IEEE Transactions on Information Theory, IT-22(6)*, pp.644-654, 1976.

[11] 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.

[12] 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.

[13] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes", *In Advances in Cryptology - Proceedings of Crypto'99, LNCS 1666*, pp.537-554, Springer-Verlag, 1999.

[14] S. Goldwasser and S. Micali, "A probabilistic encryption", *Journal of Computer and System Sciences, 28*, pp. 270-299, 1984.

[15] D. Pointcheval, "Chosen-ciphertext security for any one-way cryptosystem", *PKC'2000, LNCS 1751*, pp.129-146, Springer-Verlag, 2000.

[16] Y. Tsiounis and M. Yung, "On the security of ElGamal based encryption", *PKC'98, LNCS 1431*, pp.117-134, Springer-Verlag, 1998.

[17] 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.