International Journal of Cryptology Research 12(1): 13 - 30 (2022)

A Security-Mediated ElGamal Identity-Based Encryption Scheme

Boon Chian Tea[*1] and **Muhammad Rezal Kamel Ariffin**[1,2]

[1]*Institute for Mathematical Research (INSPEM), Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*
[2]*Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*

E-mail: gs48109@student.upm.edu.my; rezal@upm.edu.my
[*]*Corresponding author*

ABSTRACT

Tea et al. recently designed a security-mediated encryption scheme, utilizing an ElGamal variant as the primitive. The underlying assumption of the proposed scheme lies in the difficulty of solving the computational Diffie-Hellman problem. This paper extends the security-mediated ElGamal encryption scheme by Tea et al. into a mediated ElGamal IBE scheme. We modified the original mediated ElGamal encryption scheme by allowing a sender to compute the user's public key using some identity elements to suit the identity-based setting. We prove that the proposed mediated ElGamal IBE scheme is secure indistinguishably against chosen-ciphertext attack under the hardness assumption of the computational Diffie-Hellman problem.

Keywords: Mediated ElGamal IBE, Computational Diffie-Hellman, Security Mediator

1 INTRODUCTION

The concept of security-mediated (hereafter addressed simply as mediated) cryptography was introduced by Boneh et al. (2001) in their proposed mRSA to overcome the key revocation issue in public-key infrastructure (PKI). In such mediated cryptography, the receiver's private key is split into two parts by a certificate authority (CA). One is distributed to the receiver, and the other part is sent to the security mediator (SEM). Before recovering the entire plaintext, a receiver who wishes to decrypt a ciphertext must relay it to SEM for partial decryption. If such receiver is revoked due to maliciousness, CA will instruct SEM to stop performing such partial decryption for the receiver.

Mediated cryptography then expanded further since then, including the proposal of an identity-based mediated encryption scheme (IB-mRSA/OAEP) by Ding and Tsudik (2003) utilizing mRSA as the base. Chow et al. (2006) next introduced the notion of Security-Mediated Certificateless (SMC) cryptography to resolve the key escrow problem in the mediated schemes. The authors generalized the SMC framework and included the design of an IND-CCA secure, lightweight SMC scheme. To this end, the design of the SMC scheme is pairing-based, and the authors remain an open problem of realizing a pairing-free one.

Yang et al. (2007) and Lo et al. (2007) in the following year constructed two efficient certificateless pairing-free encryption schemes and a mediated revocation-free encryption scheme, respectively, in the attempt to close the addressed open problem. Unfortunately, both these schemes lacked ciphertext integrity check and were defeated by partial decryption attacks Chow and Yap (2009). Later, Seo et al. (2013) proposed an efficient certificateless mediated encryption scheme that is pairing-free, claiming to have resolved the open problem. The authors, however, did not provide any formal proof of security about the scheme.

Tea et al. (2021) recently designed a novel mediated encryption scheme, utilizing an ElGamal variant as the primitive. Via the difficulty assumption of the computational Diffie-Hellman (CDH) problem, the scheme was proven to be IND-CCA secure in the random oracle model. Although the proposal did

not advance in line with SMC cryptography, it opened a new construction of mediated schemes. In this paper, we extend the mediated ElGamal scheme by Tea et al. (2021) into a mediated ElGamal IBE scheme. We prove that our mediated ElGamal IBE scheme is IND-ID-CCA secure under the difficult assumption of the CDH problem.

2 PRELIMINARIES

We layout related mathematical background and underlying primitive used in our work, primarily the CDH problem, the security-mediated IBE setting and its security model.

2.1 Computational Diffie-Hellman Problem and Pairing Function

Definition 2.1. (Computational Diffie-Hellman (CDH) Problem). *Let g be a primitive root and g^a, g^b be two non-zero elements in cyclic group of $\mathbb{Z}_p^*$. Given the set of $\{g, g^a, g^b\}$, the CDH problem is to find g^{ab}.*

Definition 2.2. (Pairing). *A pairing function $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ takes in input on two elements of groups $\mathbb{G}_1, \mathbb{G}_2$ and outputs an element in another group (a finite field) $\mathbb{G}_T$. Such map satisfies the following three properties:*

i. Linearity (Bilinearity). Given any $P, Q, R \in \mathbb{G}_1, \mathbb{G}_2$,

$$\begin{aligned} \hat{e}(P+R, Q) &= \hat{e}(P, Q) * \hat{e}(R, Q) \\ \hat{e}(P, Q+R) &= \hat{e}(P, Q) * \hat{e}(P, R). \end{aligned}$$

ii. Non-degenaracy. For all $P \in \mathbb{G}$, $\hat{e}(P, P) \neq 1$.

iii. Computability. The function $\hat{e}$ is efficiently computable.

2.2 Security-Mediated IBE Scheme

We define the mediated IBE scheme, similar to the conventional IBE setting given by Boneh and Franklin (2001).

i) *Setup*: On input of security parameter 1^n, generates public system parameters (Params), master secret key (msk) and master public key (mpk).

ii) *Extraction*: On input of user's ID, Params and msk, generates decryption keys by splitting msk into $(K_{\mathsf{user}}, K_{\mathsf{sem}})$ pair.

iii) *Encryption*: This algorithm involves interaction between sender and receiver. Sender takes in Params, mpk, receiver's ID, and message m as input, encrypts message into ciphertext c.

iv) *Decryption*: This algorithm involves interaction between the receiver and SEM. Receiver firstly relays received ciphertext c to SEM for partial decryption $m_1 = \textsc{Dec}\,(ID, c, K_{\mathsf{sem}})$ meanwhile partially decrypting his own part $m_2 = \textsc{Dec}\,(ID, c, K_{\mathsf{user}})$. Finally, the receiver combines both m_1, m_2 to recover message $m = m_1 * m_2$, which $*$ represents the necessary operation according to different scheme settings.

2.3 Security Model of Security-Mediated IBE Scheme

Here we describe the IND-ID-CCA security game corresponds to the mediated IBE scheme defined above.

i) **Setup.** On input of security parameter 1^n, challenger $\mathcal{B}$ runs *Setup* that generates $\{\mathsf{Params}, \mathsf{mpk}, \mathsf{msk}\}$. $\mathcal{B}$ provides adversary $\mathcal{A}$ with $\{\mathsf{Params}, \mathsf{mpk}\}$.

ii) **Phase 1:**

(a) **Extraction.** $\mathcal{A}$ queries the extraction of the decryption keys for the identity ID of his choice. $\mathcal{B}$ responds correspondingly.

(b) **Decryption.** The following queries may be asked adaptively.

i. SEM-Decryption: $\mathcal{A}$ queries SEM-decryption for the $\{ID, C\}$ of his choice. $\mathcal{B}$ responds with the corresponding SEM's partial decryption to $\mathcal{A}$.

ii. Full Decryption: $\mathcal{A}$ queries full decryption for the $\{ID, C\}$ of his choice. $\mathcal{B}$ responds with decrypted message m to $\mathcal{A}$.

iii) **Challenge.** $\mathcal{A}$ produces two messages $\{m_0, m_1\}$ of equal length and an identity ID^* to be challenged. $\mathcal{B}$ randomly picks $l \in \{0, 1\}$ and outputs challenge ciphertext $C^* = \text{ENC}(\mathsf{Params}, \mathsf{mpk}, ID^*, m_l)$ to $\mathcal{A}$.

iv) **Phase 2.** $\mathcal{A}$ may repeat **Phase 1** as he wishes, except for the challenge ciphertext C^* and identity ID^*.

v) **Guess.** $\mathcal{A}$ outputs a guess of l', ending the simulation. $\mathcal{A}$ wins if $l' = l$.

Definition 2.3. ***(Indistinguishability against Chosen-Ciphertext Attack (IND-ID-CCA)****. An IBE scheme is IND-ID-CCA secure if the guessing advantage of a probabilistic polynomial-time (PPT)* $\mathcal{A}$*,* $\text{Adv}[\mathcal{A}]$ *is negligible. That is,*

$$\text{Adv}[\mathcal{A}] = \left| \Pr\left[\mathit{IBE}_{\mathcal{A}}^{\mathit{ind\text{-}id\text{-}cca}}(n) = 1 \right] - \frac{1}{2} \right| \leq \varepsilon.$$

3 SECURITY-MEDIATED ELGAMAL IBE SCHEME

The full mediated ElGamal IBE scheme is given in this section and hereafter we abbreviated it as mEGIBE for simplicity.

Algorithm 1 Key Generation of the mEGIBE.

Setup:

Input: Security parameter 1^n.

Output: System parameters $\{p, q, g, \hat{e}, \mathbb{G}, H_1, H_2, H_3, H_4, H_5\}$, user's master public key X_i and master secret key x_i.

1: On input of security parameter 1^n, generates two large primes p, q with $|p| = |q| = n$, a generator g such that $\langle g \rangle = \mathbb{Z}_p^*$.
2: Generates the following pairing and hash functions such that:
   (a) $\hat{e} : \mathbb{Z}_p^* \times \mathbb{Z}_p^* \to \mathbb{G}$, where $\mathbb{G} = \langle \hat{e}(g, g) \rangle$ of order q,
   (b) $H_1 : \{0,1\}^n \times \mathbb{Z}_p^* \to \mathbb{Z}_p^*$,
   (c) $H_2 : \mathbb{Z}_p^* \to \{0,1\}^{2n}$,
   (d) $H_3 : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^{2n}$,
   (e) $H_4 : \mathbb{Z}_p^* \times \{0,1\}^{2n} \times \mathbb{Z}_p^* \to \mathbb{Z}_p^*$,
   (f) $H_5 : \{0,1\}^* \to \mathbb{Z}_p^*$.
3: Next, chooses a random integer $x_i \in \mathbb{Z}_p^*$ and computes user i's master public key $X_i \equiv g^{x_i} \pmod{p}$.
4: Publishes system parameters $\{p, q, g, \hat{e}, \mathbb{G}, H_1, H_2, H_3, H_4, H_5\}$ and user i's master public key X_i. The master secret key x_i is kept securely.

Extraction:

Input: System parameters $\{p, q, g, \hat{e}, \mathbb{G}, H_1, H_2, H_3, H_4, H_5\}$, master secret key x_i, and user's identity ID_i.

Output: Decryption keys x_{user} and x_{sem}.

1: For user i with ID_i, randomly selects $x_{\mathsf{user}_i} \in \mathbb{Z}_p^*$ and computes $x_{\mathsf{sem}} \equiv x_i - x_{\mathsf{user}_i} \cdot H_5(ID_i) \pmod{p-1}$.
2: Distributes decryption keys x_{user_i} and x_{sem} respectively to user i and SEM.

Algorithm 2 Encryption and Decryption of the mEGIBE.

Encryption:

Input: System parameters $\{p, q, g, \hat{e}, \mathbb{G}, H_1, H_2, H_3, H_4, H_5\}$, user's master public key X_i, user's identity ID_i and message m.

Output: Ciphertext $(c_1, c_2, \gamma, \delta)$.

1: Sender who wishes to send message $m \in \{0,1\}^n$ to user i with ID_i performs the following computations:
   (a) Compute user's public key $Y_i = g^{H_5(ID_i)} \pmod p$,
   (b) Selects a random string $\sigma \in \{0,1\}^n$ and computes $r = H_1(\sigma \parallel Y_i)$,
   (c) Set $M = \sigma \parallel m$, computes $c_1 \equiv g^r \pmod p$ and $c_2 = M \oplus H_2(X_i^r)$,
   (d) Computes $\gamma = H_3(\sigma \parallel m)$ and $\delta = [H_4(c_1, c_2, Y_i)]^r \pmod p$.
2: Sends ciphertext $(c_1, c_2, \gamma, \delta)$ to user i.

Decryption:

Input: System parameters $\{p, q, g, \hat{e}, \mathbb{G}, H_1, H_2, H_3, H_4, H_5\}$, user's master public key X_i, user's identity ID_i, user's decryption key x_{user_i}, SEM's decryption key x_{sem} and ciphertext $(c_1, c_2, \gamma, \delta)$.

Output: Message m.

SEM-Decryption:

1: User i upon receiving ciphertext $C = (c_1, c_2, \gamma, \delta)$, relays it to SEM.
2: SEM validates whether $\hat{e}(g, \delta) = \hat{e}\left(c_1, H_4\left(c_1, c_2, g^{H_5(ID_i)}\right)\right)$ and replies with partial decryption $c_1^{x_{\mathsf{sem}}}$ to user i if it does. Otherwise, it rejects ciphertext C.

User-Decryption:

1: User i receives partial decryption from SEM, and computes $M' = c_2 \oplus H_2\left(c_1^{x_{\mathsf{sem}}} \cdot c_1^{x_{\mathsf{user}_i} \cdot H_5(ID_i)}\right)$.
2: Checks whether $\gamma = H_3(M')$ and parse σ and message m from $\sigma \parallel m$ if it does. Otherwise, it rejects ciphertext C.
3: Lastly, computes $r' = H_1\left(\sigma \parallel g^{H_5(ID_i)}\right)$ and verifies whether $c_1 = g^{r'} \pmod p$. If it does, then m is the valid message. Otherwise, it rejects ciphertext C.

Proof of correctness. The ciphertext validity check by SEM is true since

$$\begin{aligned}\hat{e}\left(g,\delta\right) &= \hat{e}\left(g,\left[H_4\left(c_1,c_2,g^{H_5(ID_i)}\right)\right]^r\right)\\ &= \hat{e}\left(g^r,H_4\left(c_1,c_2,Y_i\right)\right)\\ &= \hat{e}\left(c_1,H_4\left(c_1,c_2,Y_i\right)\right).\end{aligned}$$

Next, the full decryption by user can be verified effectively as

$$\begin{aligned}c_1^{x_{\mathsf{sem}}}\cdot c_1^{x_{\mathsf{user}_i}\cdot H_5(ID_i)} &= c_1^{x_{\mathsf{sem}}+x_{\mathsf{user}_i}\cdot H_5(ID_i)}\\ &= \left(g^r\right)^{x_i}\\ &= X_i^r,\end{aligned}$$

which leads to $M = c_2 \oplus H_2\left(X_i^r\right)$. Then, the extraction of $\sigma \parallel m$ allows the message m retrieval and the final integrity check of $c_1 = g^{H_1(\sigma\|Y_i)}$.

4 SECURITY ANALYSIS OF SECURITY-MEDIATED ELGAMAL IBE SCHEME

We now analyze the proof of security of mEGIBE. Intuitively, we extend the proof from the mEG scheme by Tea et al. (2021). That is, if there exists an IND-ID-CCA adversary $\mathcal{A}'$ that has the advantage over the mEG-IBE, then there exists an IND-CCA adversary $\mathcal{A}$ that has the advantage over the mEG in (Tea et al., 2021).

Theorem 4.1. *Let mEGIBE be the proposed mediated ElGamal IBE scheme as described in section 3, and $\mathcal{A}'$ be a probabilistic polynomial-time (PPT) adversary. Then the proposed mediated ElGamal IBE scheme is secure indistinguishably against chosen-ciphertext attack (IND-ID-CCA) in the random oracle model if solving the computational Diffie-Hellman (CDH) problem is difficult. That is,*

$$\Pr\left[\mathsf{mEGIBE}_{\mathcal{A}}^{\mathsf{ind\text{-}id\text{-}cca}}(n)=1\right] \leq \frac{1}{2}+\frac{\varepsilon_1 q_{H_5}^2}{q_{H_5}-q_{\mathsf{Ext}}},$$

where ε_1 denotes the negligible function, q_{H_5} and q_{Ext} represent the number of H_5 and extraction queries, respectively.

Proof. The ultimate objective of $\mathcal{B}$ is to solve the CDH problem (i.e., to find g^{ab}) given the CDH instances of $\left(g, g^a, g^b\right)$ of cyclic group $\left\{\mathbb{Z}_p^*, p, g\right\}$. We hence extend the simulation of the game between $\mathcal{B}, \mathcal{A}$ and $\mathcal{A}'$.

1. **Setup:** Challenger $\mathcal{B}$ initially takes on security parameter 1^n as input and runs ***Setup*** to output system parameters $\{p, q, g, \hat{e}, \mathbb{G}, H_1, H_2, H_3, H_4, H_5\}$ and public keys $\{X, Y\}$. It sets $X = g^a$ where $a = x$. These system parameters and public keys are sent to $\mathcal{A}$. Note that $\mathcal{B}$ does not know the secret integer x. $\mathcal{A}$ then further transmits it (except public key Y) to $\mathcal{A}'$, and controls H_5 as a random oracle.

2. H_1, H_2, H_3, H_4**-queries:** As $\mathcal{A}$ has previously interacted with $\mathcal{B}$, all the corresponding H_1, H_2, H_3, H_4-queries that had been made and kept by $\mathcal{A}$ will be used in interacting further with $\mathcal{A}'$. We briefly outline these queries for short references as follows:

   (a) H_1-query: It has the form of (w_i, W_i), this query corresponds to the $w_i = \sigma_i \parallel Y_i$ and $W_i = H_1(w_i)$.

   (b) H_2-query: It has the form of (u_i, U_i), this query corresponds to the $u_i = X_i^r$ and $U_i = H_2(u_i)$.

   (c) H_3-query: It has the form of (v_i, V_i), this query corresponds to the $v_i = \sigma_i \parallel m_i$ and $V_i = H_3(v_i)$.

   (d) H_4-query: It has the form of (z_i, Z_i), this query corresponds to the $z_i = (c_1, c_2, Y_i)$ and $Z_i = H_4(z_i)$.

   Whenever there are new queries during the interaction, $\mathcal{A}$ will pass them to $\mathcal{B}$ and replies whatever $\mathcal{B}$ replies to him.

3. H_5**-query**: Before initializing the H_5-list, $\mathcal{A}$ randomly fixes an index $j \in \{1, ..., q_{H_5}\}$ for the **Challenge** phase later. Now, when $\mathcal{A}'$ queries H_5 on ID_i, $\mathcal{A}$ proceeds as follows:

   (a) If $ID_i \neq ID_j$, $\mathcal{A}$ picks a random integer $k_i \in \mathbb{Z}_p^*$, sets $Y_i = g^{k_i}$ and replies as $Y_i = g^{H_5(ID_i)}$ to $\mathcal{A}'$. Finally, $\mathcal{A}$ adds this new tuple of (ID_i, Y_i, k_i) to the H_5-list.

   (b) If $ID_i = ID_j$, $\mathcal{A}$ sets $Y_j = g^{H_5(ID_j)} = Y$ and replies it to $\mathcal{A}'$. $\mathcal{A}$ also adds this new tuple of $(ID_j, Y_j, -)$ to the H_5-list.

4. **Phase 1:**

   (a) **Extraction query:** When $\mathcal{A}'$ queries the decryption key for ID_i, $\mathcal{A}$ search the H_5-list above for the ID_i to obtain $Y_i = g^{H_5(ID_i)}$. Note that both ID_i and Y_i correspond to the entries in H_5-tuples above.

      i. If $ID_i = ID_j$, then $\mathcal{A}$ aborts the game and the attack against mediated ElGamal (mEG) scheme failed.

      ii. Otherwise if $ID_i \neq ID_j$, $\mathcal{A}$ samples a random integer $D_i \in \mathbb{Z}_p^*$, computes $d_i = D_i k_i^{-1}$ and replies d_i as the decryption key to $\mathcal{A}'$. Lastly, it stores the new tuple (ID_i, Y_i, k_i, d_i) to the L_{key}-list.

   Since the decryption key can be viewed as a random integer in $\mathbb{Z}_p^*$, it does not matter whether $\mathcal{A}'$ query to the extraction of decryption key of SEM's or user's part.

   (b) **Decryption query:**

      i. **SEM-decryption query:** $\mathcal{A}'$ queries the SEM-decryption of (ID_i, C_i) of his choice. $\mathcal{A}$ firstly checks if such query has been made before and replies it accordingly. Otherwise, it searches through the H_1, H_4, H_5, and L_{key} lists based on the several possible cases as in Table 1.

Queried ID	**Existence of Tuples**	**Validity / Correctness**	**Computations**
$ID_i \neq ID_j$	$(w_i, W_i), (z_i, Z_i),$ $(ID_i, Y_i, k_i),$ (ID_i, Y_i, k_i, d_i)	$c_1 = g^W$; $\hat{e}(g, \gamma) = \hat{e}(c_1, Z)$; $Y_i = g^{k_i}$	Return $\left(\frac{X}{Y_i^{d_i}}\right)^W$
$ID_i \neq ID_j$	$(w_i, W_i), (z_i, Z_i),$ (ID_i, Y_i, k_i)	$c_1 = g^W$; $\hat{e}(g, \gamma) = \hat{e}(c_1, Z)$; $Y_i = g^{k_i}$	Extract d_i for ID_i Return $\left(\frac{X}{Y_i^{d_i}}\right)^W$
$ID_i \neq ID_j$	$(w_i, W_i),$ (z_i, Z_i)	$c_1 = g^W$; $\hat{e}(g, \gamma) = \hat{e}(c_1, Z)$	Run H_5-query for ID_i Extract d_i for ID_i Return $\left(\frac{X}{Y_i^{d_i}}\right)^W$
$ID_i \neq ID_j$	None	—	$\perp$
$ID_i = ID_j$	—	$Y_i = Y$	Relays (ID_i, C_i) to $\mathcal{B}_1$ and returns to Λ_1' with decryption that $\mathcal{B}_1$ returns to it

Table 1: SEM-Decryption Query of mEGIBE Simulation.

Observe that

$$X = g^x = g^{x_{\mathsf{sem}} + x_{\mathsf{user}_i} \cdot H_5(ID_i)} = g^{x_{\mathsf{sem}}} \cdot g^{d_i k_i} = g^{x_{\mathsf{sem}}} \cdot Y_i^{d_i}.$$

Then, $g^{x_{\mathsf{sem}}} = \frac{X}{Y_i^{d_i}}$ and

$$\left(\frac{X}{Y_i^{d_i}}\right)^W = (g^{x_{\mathsf{sem}}})^W = \left(g^W\right)^{x_{\mathsf{sem}}} = c_1^{x_{\mathsf{sem}}}$$

is a valid SEM's partial decryption from the $\mathcal{A}'$'s view in the simulation.

ii. **Full-decryption query:** $\mathcal{A}'$ queries the full decryption of the ciphertext $(ID_i, C_i = (c_1, c_2, \gamma, \delta))$ of his choice. $\mathcal{A}$ firstly search through all the H and L_{key} lists for the existence of these tuples $(w_i, W_i), (u_i, U_i), (v_i, V_i), (z_i, Z_i)$ and (ID_i, Y_i, k_i).

Queried ID	**Existence of Tuples**	**Validity / Correctness**	**Computations**
$ID_i \neq ID_j$	$(w_i, W_i), (u_i, U_i),$ $(v_i, V_i), (z_i, Z_i),$ (ID_i, Y_i, k_i)	$w = \sigma \parallel Y_i$; $u = X^W$; $v = \sigma \parallel m$; $c_1 = g^W$; $c_2 = v \oplus U$; $\hat{e}(g, \delta) = \hat{e}(c_1, Z)$; $Y_i = g^{k_i}$	Return m
$ID_i \neq ID_j$	$(w_i, W_i), (v_i, V_i),$ $(z_i, Z_i),$ (ID_i, Y_i, k_i)	$w = \sigma \parallel Y_i$; $v = \sigma \parallel m$; $c_1 = g^W$; $\hat{e}(g, \delta) = \hat{e}(c_1, Z)$; $Y_i = g^{k_i}$	Extract σ from $w = \sigma \parallel Y_i$ Extract m from $v = \sigma \parallel m$ Return m
$ID_i \neq ID_j$	$(w_i, W_i), (z_i, Z_i),$ (ID_i, Y_i, k_i)	$w = \sigma \parallel Y_i$; $c_1 = g^W$; $\hat{e}(g, \delta) = \hat{e}(c_1, Z)$; $Y_i = g^{k_i}$	Extract σ from $w = \sigma \parallel Y_i$ Compute $u = X^W$ Send H_2-query for (u, U)
$ID_i \neq ID_j$	$(w_i, W_i),$ (ID_i, Y_i, k_i)	$w = \sigma \parallel Y_i$; $c_1 = g^W$; $Y_i = g^{k_i}$	Revert v from $c_2 = v \oplus U$ Extract m from $v = \sigma \parallel m$ Return m
$ID_i \neq ID_j$	(w_i, W_i)	$c_1 = g^W$	Run H_5-query for Y_i Extract σ from $w = \sigma \parallel Y_i$ Compute $u = X^W$ Send H_2-query for (u, U) Reverts v from $c_2 = v \oplus U$ Extract m from $v = \sigma \parallel m$ Return m
$ID_i \neq ID_j$	None	—	$\perp$
$ID_i = ID_j$	—	$Y_i = Y$	Relays (ID_i, C_i) to $\mathcal{B}_1$ and returns to Λ_1' with decryption that $\mathcal{B}_1$ returns to it

Table 2: Full-Decryption Query of mEGIBE Simulation.

This decryption query may be asked adaptively and as many times as adversary $\mathcal{A}'$ wishes.

5. **Challenge:** $\mathcal{A}'$ terminates **Phase 1** and outputs a public identity ID^* and two distinct messages of equal length $\{m_0, m_1\} \in \{0,1\}^n$ to $\mathcal{A}$.

   (a) If $ID^* \neq ID_j$, $\mathcal{A}$ aborts the game and the attack against mEG failed.

   (b) Otherwise if $ID^* = ID_j$, $\mathcal{A}$ relays the messages m_0, m_1 to $\mathcal{B}$. $\mathcal{B}$ randomly selects bit $l \in \{0,1\}$, $\sigma^* \in \{0,1\}^n$, $R_1 \in \{0,1\}^{2n}$, and $R_2 \in \mathbb{Z}_p^*$. Next, it outputs challenge ciphertext C^* where

$$C^* = \left(g^b, R_1, \gamma, R_2\right),$$

   where g^b is taken from the CDH instance. The challenge ciphertext could be treated as the encryption of message $m_l \in \{m_0, m_1\}$ such that

   i. $b = H_1(\sigma^* \parallel Y)$,
   ii. $R_1 = M \oplus H_2\left(X^b\right)$,
   iii. $\gamma = H_3(\sigma^* \parallel m_l)$,
   iv. $R_2 = \left[H_4\left(g^b, R_1, Y\right)\right]^b \pmod p$.

   Hence, the challenge ciphertext C^* is correct and valid in the $\mathcal{A}'$'s view as long as it does not query the following to random oracles:

$$\begin{aligned} w &= \sigma^* \parallel Y \\ u &= X^b \\ v &= \sigma^* \parallel m_l \\ z &= \left(g^b, R_1, Y\right). \end{aligned}$$

6. **Phase 2:** $\mathcal{A}'$ is allowed to repeat **Phase 1** except with the targeted identity $ID^* = ID_j$ and challenge ciphertext (ID_j, C^*).

7. **Guess:** $\mathcal{A}'$ finally output its guess of l' to $\mathcal{A}$, ending the IND-ID-CCA game. Λ_1 next passes this l' as its guess to $\mathcal{B}$, ending its IND-CCA games too. $\mathcal{A}'$ wins the game if $l' = l$. $\mathcal{B}$ randomly selects one of the queries $\left((w_1, W_1), ..., \left(w_{q_{H_1}}, W_{q_{H_1}}\right)\right)$ from H_1-list and computes

$$X^b \equiv (g^a)^b \equiv g^{ab} \pmod p,$$

with $X = g^a$ set by $\mathcal{B}$ using the CDH instance, which outputs the solution to the CDH problem.

We now examine the advantage of the extended game simulation described above. If $b = H_1(\sigma^* \parallel Y)$ does exists in the stored H_1-list, then $\mathcal{A}'$ wins with the correct output of $l' = l$. This implies the success of the IND-ID-CCA game over mEGIBE, that is:

$$\mathrm{Adv}\left[\mathcal{A}'\right] = \left|\Pr\left[\mathsf{mEGIBE}_{\mathcal{A}'}^{\text{ind-id-cca}}(n) = 1\right] - \frac{1}{2}\right| \leq \frac{\varepsilon'}{q_{H_1}} \tag{1}$$

with ε' indicates the advantage of $\mathcal{A}'$. We further analyze the advantage of $\mathcal{A}$ by considering the case where the game does not abort until the guessing stage. For such, there are two conditions during simulation:

1. $\mathcal{A}'$ queries ID_j in the **Extraction** query. We denote this event as $\mathbb{E}_1$. Since j is chosen randomly from $\{1, ..., q_{H_5}\}$ and suppose $\mathcal{A}'$ makes at maximum q_{Ext} number of extraction queries. Then,
$$\Pr\left[\mathbb{E}_1\right] \leq \frac{q_{\mathsf{Ext}}}{q_{H_5}}. \tag{2}$$
2. $\mathcal{A}'$ outputs challenge identity $ID^* \neq ID_j$ in the **Challenge** phase. We denote this event as $\mathbb{E}_2$. This occurs only when the extraction query does not abort and the game continues. Thus,
$$\Pr\left[\mathbb{E}_2|\mathbb{E}_1\right] = \frac{q_{H_5} - 1}{q_{H_5}} \tag{3}$$

Therefore, the probability that $\mathcal{A}$ does not abort in the simulation is then

$$\begin{aligned}\Pr\left[\mathcal{A} \text{ does not Abort}\right] &= \Pr\left[\neg\mathbb{E}_1 \wedge \neg\mathbb{E}_2\right] \\ &= \Pr\left[\neg\mathbb{E}_2|\neg\mathbb{E}_1\right] \cdot \Pr\left[\neg\mathbb{E}_1\right] \\ &\geq \frac{1}{q_{H_5}} \cdot \left(1 - \frac{q_{\mathsf{Ext}}}{q_{H_5}}\right).\end{aligned}$$

Suppose we let $\mathrm{Adv}\left[\mathcal{A}\right]$ be the advantage of $\mathcal{A}$ winning the IND-CCA game against mEG, such that

$$\mathrm{Adv}\left[\mathcal{A}\right] = \left|\Pr\left[\mathsf{mEG}_{\mathcal{A}}^{\text{ind-cca}}(n) = 1\right] - \frac{1}{2}\right| \geq \varepsilon. \tag{4}$$

Also, the event of $\mathcal{A}$ outputs $l' = l$ is identical to the event of $\mathcal{A}'$ outputting the same $l' = l$. Hence, the following probabilities are equivalent.

$$\begin{aligned} & \Pr\left[\mathcal{A} \text{ outputs } l' = l \mid \mathcal{A} \text{ does not Abort}\right] \\ = \; & \Pr\left[\mathcal{A}' \text{ outputs } l' = l \mid \mathcal{A} \text{ does not Abort}\right] \\ = \; & \mathrm{Adv}\left[\mathcal{A}'\right]. \end{aligned}$$

Combining all the probabilities, it is easy to see that from (1), $\mathrm{Adv}\left[\mathcal{A}\right]$ is then

$$\begin{aligned} \mathrm{Adv}\left[\mathcal{A}\right] &= \Pr\left[\mathcal{A} \text{ outputs } l' = l \wedge \mathcal{A} \text{ does not Abort}\right] \\ &= \Pr\left[\mathcal{A} \text{ outputs } l' = l \mid \mathcal{A} \text{ does not Abort}\right] \cdot \Pr\left[\mathcal{A} \text{ does not Abort}\right] \\ &\geq \mathrm{Adv}\left[\mathcal{A}'\right] \cdot \frac{1}{q_{H_5}} \cdot \left(1 - \frac{q_{\mathsf{Ext}}}{q_{H_5}}\right) \\ &\geq \frac{\varepsilon'}{q_{H_1}} \cdot \frac{1}{q_{H_5}} \cdot \left(1 - \frac{q_{\mathsf{Ext}}}{q_{H_5}}\right) \\ &\geq \frac{\varepsilon'}{q_{H_1} q_{H_5}} \cdot \left(1 - \frac{q_{\mathsf{Ext}}}{q_{H_5}}\right). \end{aligned}$$

Extending from (4) such that $\mathrm{Adv}\left[\mathcal{A}\right] \geq \varepsilon$ and re-arranging the expression, we obtain

$$\begin{aligned} \frac{\varepsilon'}{q_{H_1} q_{H_5}} \cdot \left(1 - \frac{q_{\mathsf{Ext}}}{q_{H_5}}\right) &\leq \varepsilon \\ \varepsilon' \cdot \left(1 - \frac{q_{\mathsf{Ext}}}{q_{H_5}}\right) &\leq \varepsilon q_{H_1} q_{H_5} \\ \varepsilon' &\leq \frac{\varepsilon q_{H_1} q_{H_5}^2}{q_{H_5} - q_{\mathsf{Ext}}}. \end{aligned}$$

Finally, combining everything together,

$$\begin{aligned} \Pr\left[\mathsf{mEGIBE}_{\mathcal{A}'}^{\mathsf{ind\text{-}id\text{-}cca}}(n) = 1\right] &\leq \frac{1}{2} + \frac{\varepsilon'}{q_{H_1}} \\ &\leq \frac{1}{2} + \frac{1}{q_{H_1}} \cdot \left(\frac{\varepsilon q_{H_1} q_{H_5}^2}{q_{H_5} - q_{\mathsf{Ext}}}\right) \\ &= \frac{1}{2} + \frac{\varepsilon q_{H_5}^2}{q_{H_5} - q_{\mathsf{Ext}}}. \end{aligned}$$

These complete the proof of security of the proposed mediated ElGamal IBE scheme. □

5 PERFORMANCE ANALYSIS

The computational efficiency is layout in Table 3. We argue that our scheme is reasonably efficient in computational as major operations are exponentiation common in most practical cryptographic schemes. Although pairing computation is relatively costly, we only involve it in the SEM's ciphertext integrity checking and not on the user's side.

Operation	**X-OR**	**Subtraction/ Multiplication**	**Exponentiation**	**Hashing**	**Pairing**
Setup	0	0	1	0	0
Extraction	0	2	0	1	0
Encryption	1	0	4	5	0
SEM-Decryption	0	0	2	2	2
User-Decryption	1	2	3	4	0

Table 3: Computational Efficiency of The Proposed Mediated ElGamal IBE Scheme.

6 CONCLUSION

In this paper, we extended the mediated ElGamal scheme into a mediated ElGamal IBE scheme and proved that our scheme is IND-ID-CCA secure via the hardness assumption of the CDH problem. The setting of our proposed scheme can be altered efficiently into an elliptic curve and pairing-based with the hardness assumptions of the elliptic curve Diffie-Hellman (ECDH) and bilinear Diffie-Hellman (BDH) problems, respectively. Our current proposal does not advance compared to the certificateless type, which will be considered in our future work.

ACKNOWLEDGMENTS

The Universiti Putra Malaysia Grant partially supported the present research with Project Number GP-IPS/2018/9657300. In addition, the first author would like to further express appreciation to the Institute for Mathematical Research (INSPEM), Universiti Putra Malaysia (UPM), and the Ministry of Higher Education (MoHE) for allowing conducting this research.

REFERENCES

Boneh, D., Ding, X., Tsudik, G., and Wong, C. M. (2001). A Method for Fast Revocation of Public Key Certificates and Security Capabilities. In *Proceedings of the 10th Conference on USENIX Security Symposium - Volume 10*, SSYM'01. USENIX Association.

Boneh, D. and Franklin, M. (2001). Identity-Based Encryption from The Weil Pairing. In Kilian, J., editor, *Advances in Cryptology – CRYPTO 2001*, pages 213–229. Springer Berlin Heidelberg.

Chow, S. and Yap, W.-S. (2009). Partial decryption attacks in security-mediated certificateless encryption. *IET Information Security*, 3:148–151.

Chow, S. S. M., Boyd, C., and Nieto, J. M. G. (2006). Security-Mediated Certificateless Cryptography. In Yung, M., Dodis, Y., Kiayias, A., and Malkin, T., editors, *Public Key Cryptography - PKC 2006*, pages 508–524, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ding, X. and Tsudik, G. (2003). Simple Identity-Based Cryptography with Mediated RSA. In *Topics in Cryptology - CT-RSA 2003*, pages 193–210. Springer Berlin Heidelberg.

Lo, C. M., Hwang, T., and Li, C. (2007). Revocation-Free Public-Key Encryption Based on Security-Mediated Public-Key Infrastructure. *IET Inf. Secur.*, 1(3):134–141.

Seo, S.-H., Nabeel, M., Ding, X., and Bertino, E. (2013). An Efficient Certificateless Cryptography Scheme without Pairing. In *Proceedings of the*

Third ACM Conference on Data and Application Security and Privacy, CODASPY '13, page 181–184, New York, NY, USA. Association for Computing Machinery.

Tea, B. C., Kamel Ariffin, M. R., Abd. Ghafar, A. H., and Asbullah, M. A. (2021). A Security-Mediated Encryption Scheme Based on ElGamal Variant. *Mathematics*, 9(21).

Yang, C., Wang, F., and Wang, X. (2007). Efficient Mediated Certificates Public-Key Encryption Scheme without Pairings. In *21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07)*, volume 1, pages 109–112.