A New Efficient Identity-Based Encryption Without Pairing

Majid Salimi

Faculty of computer Engineering, University of Isfahan, Isfahan, Iran
MajidSalimi3@gmail.com

Abstract. So far, most of the Identity-Based Encryption (IBE) schemes have been realized by employing bilinear pairings, lattices, trapdoor discrete logarithm, or based on the quadratic residue problem. Among the IBE schemes, only pairing-based methods seem to be practical. Previously published non-pairing-based schemes are generally inefficient in encryption, decryption, key generation, ciphertext size or key size. In this paper, we propose an IBE scheme based on a hybrid of Diffie-Hellman and RSA-like hardness assumption. The computational cost of the proposed scheme is lower than the previous schemes and the ciphertext size for an l-bit plaintext is only $2l$ bits. The proposed scheme is similar to the well-known ElGamal encryption algorithm; therefore it might be used in applications such as oblivious computation.
Keywords: Identity-based encryption, Discrete logarithm problem, Trapdoor Decisional Diffie Hellman problem.

1 Introduction

In public-key cryptography, the binding between a public key and the owner of the corresponding private key is provided by a digital certificate signed by a trusted Certificate Authority (CA). This setting not only requires universal trust of users in CAs, but also imposes heavy computations on users; whenever Alice wants to employ Bob's public key, for example for encryption or signature verification, she must first obtain Bob's certificate and then validate the CA's signature on it. To avoid these costs, Shamir proposed the idea of identity-based cryptography in 1984 [5].

Identity-based cryptosystems do not need any CA or certificate, because in these systems the user's public key is constructed as a function of his identity such as his name, email address or telephone number. That is why identity-based systems are very cost-effective. The corresponding private key is generated by a Key Generation Center (KGC) by applying its master secret key. To use identity-based cryptography, the user is authenticated by the KGC and gives his unique identity to receive the corresponding private key through a secure channel. In this manner, to send an encrypted message to Bob, or to verify his signature on a message, other users require only Bob's identity and the KGC's public key.
The IBE schemes ID-Based cryptography has many uses, such as the Internet

of Things (IoT) [**?**]. In the IoT and the battery and processing power is limited. Because of the elimination of the PKI, an efficient IBE scheme can be a viable option for IoT.

1.1 Related Work

In 1984 Shamir presented an identity-based signature scheme based on RSA [5], but invention of practical identity-based encryption (IBE) remained as an open problem until 2001 when Boneh and Franklin presented an IBE based on bilinear pairings over elliptic curves [4].

In 2001, Cocks presented a new IBE scheme based on the quadratic residue problem [3]. In this scheme, the computational cost of encrypting an l-bit plaintext is l Jacobi symbols computation and l modular divisions. Based on the analysis performed in Table 1 of [2], computing 1024-bit Jacobi symbol on a specified processor lasts 0.135 ms and the cost of 1024-bit exponentiation in this processor is 5.0 ms. Based on these numbers, the cost of 1024-bit encryption with Cock's scheme is about 140 ms which is equivalent to the cost of computing about 28 1024-bit modular exponentiations. The cost of decryption level. Moreover, the ciphertext size is extremely long, i.e., the size of ciphertext for an l-bit plaintext is $2l.log_2N$, where N is the modulus of the computation. For example, a 128-bit plaintext is transformed to a 32768-byte ciphertext.

Gentry et al. [2] extended the Cock's scheme to solve the ciphertext size problem. They reduced the ciphertext size to $l + log_2N$ bits, but the computational cost of their scheme increased dramatically. For l-bit encryption with 1024-bit modulus, this scheme needs to produce l 1024-bit primes. The cost of a 1024-bit prime generation is reported about 123.6 ms in Table 1 of [2], so based on this number, generation of the required prime numbers for a 1024-bit plaintext is about $1024 \times 123.6(\text{ms}) \approx 126(\text{s})$; therefore, this scheme seems to be impractical.

Peterson and Srinivasan presented a new IBE scheme based on the Trapdoor Discrete Logarithm (TDL) problem [12]. The idea is that, computing a discrete logarithm in a special RSA modulus with knowing its factorization has a complexity equivalent to about the square root of the complexity of solving that without knowing the factorization of the modulus [12]. This algorithm is efficient in encryption and decryption but the system setup and key-extraction in this scheme are too costly. For example for 80-bit security, the pre-computing cost of system setup is about 2^{48} bit operations and the cost of each private key extraction is about 2^{26} operations. Hence, this scheme seems to be impractical. In 2019, Ramadan et al. proposed an IBE scheme under the RSA assumption providing equality test, their scheme is like Peterson-Srinivasan scheme and suffers from the same problem, which is costly key-extraction [12], [13].

Dodis et al. introduced the idea of bounded-collusion IBE scheme [29] and then Goldwasser et al. and Tessaro and Wilson proposed other bounded-collusion

IBE schemes [30, 28]. The bounded-collusion IBE schemes are secure if the adversary can obtain only private keys of t users and these schemes are not fully secure [28]. In 2003, Ding and Tsudik present a mediated RSA-based IBE scheme. The main Idea of their scheme was splitting an RSA private key between the user and a Security Mediator. Their scheme needs an online server and this server will be a bottleneck of the system [30].

Boneh et al. [14] and Agrawal et al. [15, 16] presented lattice-based IBE schemes; lattices, however, are more costly in computation, compared to modular exponentiation and bilinear pairing, and therefore, the lattice-based schemes are also too costly and inefficient in both computation and ciphertext size.

Boneh et al. showed that an anonymous IBE scheme, in which the ciphertext does not reveal any information about the receiver's identity, can be used in the keyword-searchable encryption [11, 10]. In asymmetric searchable encryption schemes, keywords, which are identities in the IBE scheme, can be used to encrypt the database. In these systems, anyone can encrypt his data using the server's public key and his keyword (as identities). The resulted ciphertext is stored in the database and the keyword would be the key of database. Finally, the server uses his master secret key to derive the secret key corresponding to the searching keywords and gives it to the authorized users. Authorized users can search the database based on the keywords and decrypt the results.

In 2016, Park et al. present a new IBE scheme based on Trapdoor Diffie Hellman on large RSA modulus [24]. In just a few days Hanzlik and Kluczniak attacked their scheme and break the security of their scheme [25]. They showed that just two instances of the hard problem of their scheme($\tilde{q}$-TSDH problem) yield the attacker to obtain factorization of N [25]. The proposed IBE scheme in this paper, uses the Trapdoor Diffie Hellman problem in a different way and we proved that our hard problem is secure.

1.2 Our Contribution

In this paper, we propose an IBE scheme based on an RSA-modulus TDDH assumption, a hybrid of Diffie-Hellman and RSA-like hardness assumption. Compared with previously published IBE schemes, the proposed scheme is very efficient in all different stages of an IBE scheme including setup, key-extraction, encryption and decryption. In particular, the decryption cost of Boneh-Boyen and Sakay-Kasahara schemes is 10 times higher than encryption, which can make the servers vulnerable to DDoS attacks. Notably, the proposed scheme reduces the computation cost of the decryption phase more than 10 times by omitting bilinear pairing from it. This property makes the proposed scheme suitable for the Internet of Things (IoT). The security of the proposed scheme is proved formally in the random oracle model.

1.3 Paper Organization

The rest of this paper is organized as follows. In Section 2, the required notations, the general IBE model and the required security definitions are introduced. In Section 3, we describe the proposed identity-based encryption scheme, and then in Section 4, the security of this scheme is investigated. Section 5 compares the performance of the proposed scheme with related schemes, and finally, the paper is concluded in Section 6.

2 Preliminaries

In this section, the required notations, the general form of IBE schemes, and the underlying security model are introduced.

2.1 Notations

The notations applied throughout this article are listed in Table 1.

Table 1. Notations.

Symbol	**Description**
$H_1(\cdot), H_2(\cdot), H_3(\cdot)$	Three cryptographic hash functions
ID_i	Identity of user i
$msk = \{s_1, s_2, p, q\}$	KGC's master secret keys
$P_{pub} = \{g_1, g_2, g_3\}$	KGC's public keys
M	Plaintext
e_i	Public key of user i which is calculated directly from his identity
d_i	Private key of user i
N	Product of two large prime numbers
g	A generator of group $\mathbb{G}$
r	A random element in $\mathbb{G}$
PP	The set of public parameters $\{P_{pub}, N, H_1(\cdot), H_2(\cdot), H_3(\cdot), g\}$
$CT = (C_1, C_2)$	Ciphertext

2.2 The general IBE model

An IBE scheme is specified by the following four algorithms [4]:

1. ***Setup(λ):*** Generates KGC's master secret key msk and a set of public parameters PP based on a security parameter λ. The parameter λ shows, for example, the size of modulus of computation which defines a specific level of security.

2. ***KeyGen***(msk, PP, ID_i)***:*** Takes the master secret key msk, the set of public parameters PP and the identity of user i as input, and generates the private key d_i for the user with identity ID_i.

3. ***Encrypt***(PP, M, ID_i)***:*** Takes the set of public parameters PP, a message M and the identity of user i, as input. The *Encrypt* function returns the ciphertext CT which is the encrypted form of M under public key e_i, which is obtained from ID_i.

4. ***Decrypt***(CT, PP, d_i)***:*** Takes a ciphertext CT encrypted by the public key e_i, the set of public parameters PP and the private key d_i as inputs. Its output is the message M.

2.3 Security Model

Here, we adopt the INDistinguishable IDentity Chosen Ciphertext Adversary (*IND-ID-CCA*) security model, which was first introduced by Boneh and Franklin for the IBE systems [4]. In this model, an adversary and a challenger play the following game.

1. **Setup.** In this phase the challenger is given as input an instance of a hard problem and then compute public parameters, and finally gives them to the adversary.

2. **Phase 1.** The adversary adaptively asks the challenger to answer at most n queries where each query is one of the following types.
   (a) Key generation query(ID_i): The adversary asks the challenger for generating private key d_i, for any arbitrary ID_i.
   (b) Decryption query(CT_j, ID_i): The adversary asks the challenger to decrypt the ciphertext CT_j, for any arbitrary identity ID_i.
3. **Challenge.** The adversary submits two equal length messages M_0, M_1 and an arbitrary identity ID^* that it wishes to attack. Then the challenger selects a random bit $\gamma \in \{0, 1\}$, computes the ciphertext $CT^* = Encrypt(PP, M_\gamma, ID^*)$, and then sends it back to the adversary as a challenge.
4. **Phase 2.** This phase is the same as Phase 1, in which the adversary issues several additional queries except that the adversary is not allowed to ask the challenger to decrypt CT^* for ID^*.
5. **Guess.** The attacker guesses $\acute{\gamma} \in \{0, 1\}$, and it will win if $\acute{\gamma} = \gamma$.

The adversary in the above game is called an *IND-ID-CCA* adversary. The advantage of IND-ID-CCA adversary A in attacking the IBE scheme $\mathcal{E}$ in the above game is defined as:

$$Adv_A^{\mathcal{E}} = |Pr[\gamma = \acute{\gamma}] - 1/2| \tag{1}$$

Definition 1. *The IBE scheme $\mathcal{E}$ is secure against the IND-ID-CCA attack, if any $t-$time adversary A that makes at most n key generation query and decryption query does not have any advantage greater than ϵ (i.e. $(Adv_A^{\mathcal{E}} > \epsilon)$). Then we say that the scheme is (t, ϵ, n)-IND-ID-CCA secure [9].*

Definition 2. *A bilinear pairing is a function $e : \mathbb{G}_1 \times \mathbb{G}_2 = \mathbb{G}_t$, where $\mathbb{G}_1$ and $\mathbb{G}_2$ are cyclic groups of prime order p, and $\mathbb{G}_t$ is the target group of the same order. Assume $g_2 \in \mathbb{G}_2$ and $g_1 \in \mathbb{G}_1$ are the generators of their respective groups. The function e must satisfy the following conditions [7]:*
Non-degeneracy:

$$\forall a, b \in \mathbb{Z}_p, e({g_1}^a, {g_2}^b) \neq 1. \tag{2}$$

Bilinearity:

$$\forall a, b \in \mathbb{Z}_p, e({g_1}^a, {g_2}^b) = e(g_1, g_2)^{ab} \in \mathbb{G}_t. \tag{3}$$

To prove the security of our scheme, we need to define the following assumptions, as well.

Definition 3. *Decisional Diffie-Hellman (DDH): Let N be a safe RSA modulus, $\mathbb{G}$ a subgroup of $\mathbb{Z}_N^*$ of order p and g a generator of $\mathbb{G}$. Given $g, g^x \bmod N$ and $g^r \bmod N$ in $\mathbb{G}$, the adversary A cannot distinguish $g^{xr} \bmod N$ from T, where x, T and r are chosen uniformly at random.*

Adversary A that outputs a bit $\gamma \in \{0,1\}$ has advantage $Adv_A^{DDH} = \epsilon$ in solving the DDH problem in $\mathbb{G}$ if

$$|Pr[A(g, g^x, g^r, g^{xr}) = 0] - Pr[A(g, g^x, g^r, T) = 0]| \geq \epsilon \tag{4}$$

One variant of the Decisional Diffie Hellman assumption (DDH) is the Trapdoor Decisional Diffie Hellman (TDDH). In this paper, we use a modified version of the TDDH assumption for the security proof of the proposed IBE scheme.

Definition 4. *RSA-based Trapdoor Decisional Diffie-Hellman (TDDH-(g, X, d)) [21]: Let N be a safe RSA modulus and $\mathbb{G}$ be a subgroup of order pq of $\mathbb{Z}_N^*$. Jacobi symbol of all of elements of $\mathbb{G}$ is one, thus $\mathbb{G}$ is cyclic. Given $g \in \mathbb{G}$, $X = g^x \bmod N \in \mathbb{G}$ and $d = x^{-1} \bmod q \in \mathbb{G}$, the user i can distinguish $(g, g^r \bmod N, g^x \bmod N, g^{xr} \bmod N)$ from $(g, g^r \bmod N, g^x \bmod N, T)$, where T, x and r are chosen uniformly at random and g is a generator of $\mathbb{G}$ and $x^{-1} \bmod q$ is a trapdoor of the problem.*

This means that the DDH problem in subgroup $\mathbb{G}$ will be easy when the trapdoor of the problem is given and will remain hard without the trapdoor. Thus, if an adversary obtains a trapdoor for TDDH, it means that it can solve the DDH problem [21]. Seurin proved that the TDDH assumption is equivalent to the e-th root problem in RSA modulus $N = \acute{p}\acute{q}$ [21].

Definition 5. *$\tilde{n}$-RSA-based Trapdoor Decisional Diffie-Hellman ($\tilde{n}$-TDDH): Let N be a safe RSA modulus and $\mathbb{G}$ be a subgroup of order pq of $\mathbb{Z}_N^*$. Jacobi symbol of all of the elements of $\mathbb{G}$ is one, thus $\mathbb{G}$ is cyclic. Given $(N, g^{ps_1}, g^{ps_2}, g_3 =$*

$g^{p^2}, \{h_{i_1} = y_{i_1}p + k_{i_1}q, h_{i_2} = y_{i_2}p + qk_{i_2}, d_i = (s_1y_{i_1} + s_2y_{i_2})^{-1} \bmod q\}_{i=1}^{n})$ *and* $h_{c_1} = y_{c_1}p + qk_{c_1}, h_{c_2} = y_{c_2}p + qk_{c_2}$ *as input distinguishing a tuple* $(g_3, {g_3}^r, {g_3}^{(y_{c_1}s_1y_{c_2}s_2)}, {g_3}^{r(y_{c_1}s_1y_{c_2}s_2)})$ *from* $(g_3, {g_3}^r, {g_3}^{(y_{c_1}s_1y_{c_2}s_2)}, T)$ *with un-negligible probability is intractable, where* T *is a random integer. The advantage of adversary* A *(i.e.* Adv_A^{n-TDDH}*) is defined as*

$$|Pr[A(g_3, {g_3}^r, {g_3}^{x_c}, {g_3}^{rx_c}) = 0] - Pr[A(g_3, {g_3}^r, {g_3}^{x_c}, T)]| \geq \epsilon, \tag{5}$$

where $x_c = s_1y_{c_1} + s_2y_{c_2}$.

The $d_i = (s_1y_{i_1} + s_2y_{i_2})^{-1} \bmod q$ is a trapdoor for user i and by using it the user i can compute ${g_3}^r \bmod N$ from ${g_3}^{r(y_1s_1+y_2s_2)} \bmod N$. Note that the adversary cannot obtain y_{i_1} and y_{i_2} from h_{i_1} and h_{i_2}. Furthermore, in the proposed scheme h_{i_1} and h_{i_2} are just random integers (hashed values) and are useless to the adversary. Assume that instead of $d_i = (y_{i_1}s_1 + y_{i_2}s_2)^{-1} \bmod q$ the adversary has access to $y_{i_1}s_1 + y_{i_2}s_2$, then by accessing to $3n$ equation $y_{i_1}s_1 + y_{i_2}s_2$, $y_{i_1}p + k_{i_1}q$ and $y_{i_2}p + k_{i_2}q$ for $i = 1, \ldots, n$, the adversary just have $3n$ equation and $\frac{4n}{3}+4$ unknown parameter. These equations are underdefined, and they have many possible solutions. Thus by accessing n private key the adversary cannot solve these set of equations to find p, q, s_1 and s_2. The security of $\tilde{n}$-TDDH is discussed in Theorem 1.

3 The proposed identity-based encryption scheme

In this section, we describe our new identity-based encryption scheme. The description mainly follows the general model introduced in Section 2.

3.1 Setup(λ)

The KGC first produces two $(\lambda/2)$-bit safe primes $\acute{p}$ and $\acute{q}$, where $\acute{p} = 2p+1$ and $\acute{q} = 2q + 1$. Afterwards, the KGC produces the safe RSA modulus as $N = \acute{p}\acute{q}$. In this stage, the KGC chooses two random integers $s_1, s_2 \in \mathbb{Z}_N^*$. and a random generator g of $\mathbb{G}$ with order pq and then calculates its two public keys as:

$$P_{pub} = (g_1, g_2, g_3) = (g^{ps_1} \bmod N, g^{ps_2} \bmod N, g^{p^2} \bmod N). \tag{6}$$

The KGC presents a hash function $H_1 : \{0,1\}^* \longrightarrow \{0,1\}^\lambda$ and $H_2 : \{0,1\}^* \longrightarrow \{0,1\}^\lambda$, (i.e. the output of $H_1(\cdot)$ and $H_2(\cdot)$ are a λ-bit integers).
The KGC also present a hash function $H_3 : \{0,1\}^\lambda \to \{0,1\}^\eta$, where $\eta = log_2 M$.
The public key of user i with identity ID_i is computed as:

$$e_i = {g_1}^{h_{i_1}} \times g_2^{h_{i_2}} \bmod N \tag{7}$$

where $h_{i_1} = H_1(ID_i)$ and $h_{i_2} = H_2(ID_i)$. Any integer bigger than $pq - p - q$ (Frobenius number) can be expressed by linear combination of p and q with positive coefficient [22]. So, we have

$$h_{i_1} = y_{i_1}p + k_{i_1}q \tag{8}$$

$$h_{i_2} = y_{i_2}p + k_{i_2}q \tag{9}$$

Finally the set of public parameters is announced as $PP = \{N, g, P_{pub}, H_1(\cdot), H_2(\cdot), H_3(\cdot)\}$.

3.2 $KeyGen(msk, PP, ID_i)$

User i sends his identity to the KGC who first computes $h_{i_1} = H_1(ID_i)$ and $h_{i_2} = H_2(ID_i)$ then it computes

$$\begin{aligned} y_{i_1} = p^{-1}h_{i_1} \bmod q =& p^{-1}(y_{i_1}p + k_{i_1}q) \bmod q \\ =& y_{i_1}pp^{-1} \bmod q, \end{aligned} \tag{10}$$

$$\begin{aligned} y_{i_2} = p^{-1}h_{i_2} \bmod q =& p^{-1}(y_{i_2}p + k_{i_2}q) \bmod q \\ =& y_{i_2}pp^{-1} \bmod q. \end{aligned} \tag{11}$$

Finally the KGC produces the corresponding private key d_i and sends it to user i through a secure channel.

$$d_i = (y_{i_1}s_1 + y_{i_2}s_2)^{-1} \bmod q \tag{12}$$

3.3 $Encrypt(PP, m, e_i)$

The message $m \in \mathbb{Z}_N^*$ is encrypted using the public key of user i and also a random r chosen from $\mathbb{Z}_N^*$ as below:

$$F = {g_3}^r \bmod N \tag{13}$$

$$C_1 = m \oplus H_3(F) \tag{14}$$

$$C_2 = {e_i}^r \bmod N = ({g_1}^{y_{i_1}} \times g_2^{y_{i_2}})^r = {g_3}^{(y_{i_1}s_1 + y_{i_2}s_2)r} \bmod N \tag{15}$$

Then, the ciphertext $CT = (C_1, C_2)$ is sent to the user i.

3.4 $Decrypt(CT, PP, d_i)$

Upon receiving a ciphertext $CT = (C_1, C_2)$, the user with identity i uses his private key d_i to decrypt the ciphertext CT as:

$$F = {C_2}^{d_i} = {g_3}^r \bmod N, \tag{16}$$

$$m = C_1 \oplus H_3(F) \tag{17}$$

4 Security Analysis

In this section, we prove the security of the proposed scheme based on the IND-ID-CCA model [9] in the following two theorems. In Theorem 1 we will show that the adversary that has access to n private keys cannot obtain any information about p, q and master secret of KGC (i.e. s_1 and s_2). The security of this scheme relies on the intractability of factorizing the large RSA modulus N. In other words, if the attacker manages to obtain p and q, then it can compute the private key of all users. In Theorem 2, the security of the proposed IBE scheme is proved *IND-ID-CCA* model.

Theorem 1. *Let $N = \acute{p}\acute{q}$ be a λ-bit safe RSA modulus with unknown factorization and $\mathbb{G}$ be a cyclic subgroup of order p, and $d_i = {x_i}^{-1} \bmod N$. Given several instances of private keys of proposed scheme (d_i), computing the factorization of modulus N is infeasible if the value of x_i's be unknown and for any two instance x_i and x_j the value of $x_i - x_j$ and x_i/x_j be unknown to the attacker.*

Proof. Suppose there is an adversary A that has access to several instances of d_i's which satisfy the Theorem 1 conditions. If the adversary can compute a private key d_k, which not satisfy Theorem 1 conditions, then we show that there exists an algorithm $\acute{A}$ that can factor the modulus of computations with approximately the same running time. This algorithm consists of three phases as below:
Initialization. Given N, algorithm $\acute{A}$ as the challenger generates n random integers $d_i \in_R \mathbb{Z}_q^*$ and sends it to attacker. Note that the challenger does not know the factorization of N and there exists a unique element x_i such that $x_i = {d_i}^{-1} \bmod q$.
Challenge. The algorithm $\acute{A}$ asks the attacker to choose one of d_i's and then compute a private key d_k which does not met conditions of Theorem 1 related to one of the challenge private keys.
Response. If the adversary computes d_k and $x_i - x_k$ and gives them to $\acute{A}$, the algorithm $\acute{A}$ can factor N as follow:

$$d_k d_i(x_i) = d_k \bmod q \tag{18}$$
$$d_i d_k(x_k) = d_i \bmod q \tag{19}$$
$$d_k d_i(x_i - x_k) = d_k - d_i \bmod q \tag{20}$$
$$d_k d_i(x_i - x_k) - d_k + d_i = 0 \bmod q \tag{21}$$

which is multiple of q.
If the adversary computes d_k and $v = x_k/x_i$ and gives them to $\acute{A}$, the algorithm $\acute{A}$ can factor N as follow:

$$d_i(x_i) = 1 \bmod q \tag{22}$$
$$d_k(x_i v) = 1 \bmod q \tag{23}$$
$$d_i x_i - d_k x_i v = 0 \bmod q \tag{24}$$
$$x_i = d_i - d_k v \bmod q \tag{25}$$
$$d_i x_i = 0 \bmod q. \tag{26}$$

■

In other words, in spite of having the finite regular set of private keys, without knowing the factorization of modulus of computations, it is intractable to compute a another valid private key. Note that the h_{i_1} and h_{i_2} are just random integers, so the adversary can not obtain any useful information from them.
The Park et al scheme does not satisfy the conditions of Theorem 1, because the difference of public keys of users i and j is known [24]. This flaw is the main idea of Hanzlik and Kluczniak attack [25].

Theorem 2. *Let N be a safe RSA modulus and $\mathbb{G}$ be a subgroup of $\mathbb{Z}_N^*$. If the $\tilde{n}$-TDDH problem holds in $\mathbb{G}$, then, the proposed IBE scheme is secure against IND-ID-CCA adversary.*

Proof. Assume we have an adversary A that breaks the proposed IBE scheme with advantage ϵ, then we show that there exists an algorithm B that solves the $\tilde{n}$-TDDH problem with the advantage $\frac{\epsilon}{e(n+1)}$, and approximately the same running time. We use IND-ID-CCA model for our proof with random oracle [4]. The algorithm B is given an instance of $\tilde{n}$-TDDH problem as input. The goal of algorithm B is to distinguish $(g, g^x \bmod N, g^r \bmod N, g^{rx} \bmod N)$ from $(g, g^x \bmod N, g^r \bmod N, T)$, where T is a random integer. Algorithm B interacts with adversary A in the following game.

1. ***Setup.*** In this phase the challenger is given an instance of the $\tilde{n}$-TDDH as input; that is $(N, g^{ps_1}, g^{ps_2}, g_3 = g^{p^2}, \{h_{i_1} = y_{i_1}p + k_{i_1}q, h_{i_2} = y_{i_2}p + qk_{i_2}, d_i = (s_1 y_{i_1} + s_2 y_{i_2})^{-1} \bmod q\}_{i=1}^{n})$ and $h_{c_1} = y_{c_1}p + qk_{c_1}, h_{c_2} = y_{c_2}p + qk_{c_2}$. We will use two random oracles $O_1(\cdot)$ and $O_2(\cdot)$, the random oracle $O_1(\cdot)$ simulates the two hash functions $H_1(\cdot)$ and $H_2(\cdot)$ and the random oracle $O_3(\cdot)$ simulates the hash function $H_3(\cdot)$. These random oracles are controlled by the challenger, and the adversary can obtain the hash value for any arbitrary identity (or elements of $\mathbb{Z}_N^*$) by asking it from these random oracles.
   Random Oracle $O_1(ID_i)$***.*** The identity of user i is given to random oracle $O_1(\cdot)$ as an input to generate a hash value. Upon receiving the identity of user i as a new query, the challenger first searches its database, if it finds a tuple matching the identity of user i, it will return that tuple to the adversary. If, however, there is no matching tuple in its database, it will act as following: the challenger generates a random bit $b_i \in \{0, 1\}$ and if $b_i = 0$ then it sends h_{c_1} and h_{c_2} to adversary, else if $b_i = 1$, then it chooses a unique random integer $j \in \{1, \ldots, n\}$ and returns h_{j_1} and h_{j_2} to the adversary. Finally, it saves the identity as well as h_{j_1}, h_{j_2} and the bit b_i to its database.
   Random Oracle $O_2(F)$***.*** The hash function $H_3(\cdot)$ is simulated with the random oracle $O_2(\cdot)$ which is given F as an input and generates a random integer as a hashed value. The challenger first searches its database; if it finds a tuple matching the identity of user i, it will return that tuple to the adversary. If, however, there is no matching tuple in its database, the challenger generates a random integer Z for simulating $H_3(F)$ and sends it as output. Then it saves the F and Z to its database. Finally, the challenger sends $\{N, g, g^{ps_1}, g^{ps_2}, g_3, O_1(\cdot), O_2(\cdot)\}$ as public parameter PP to the adversary.

2. ***Phase 1.*** The adversary A is allowed to makes at most n queries, where each one is a Key generation query(ID_i) or Decryption query(CT, ID_i) where ID_i and CT is specified by A. Then algorithm B sends a private key d_i corresponding to the specified ID_i or plaintext m and sends it back to A.
   KeyGen query(ID_i, PP). The challenger searches its database to find respective tuple if $b_i = 0$ it rejects the query and game is finished. Otherwise,

it sends the respective private key which equals to $(y_{i_1}s_1 + y_{i_2}s_2)^{-1} \bmod q$, to the adversary.
Decryption query(CT, ID_i, PP). The challenger will searches the database $O_1(\cdot)$ to find a tuple which contains ID_i, if $b_i = 1$ then it extract the value of d_i from it and computes $O_2({C_2}^{d_i}) \oplus C_1$ and sends it as a plaintext to the adversary. If, however, $b_i = 0$, the challenger rejects the query.

3. ***Challenge.*** The adversary A chooses two messages $m_0, m_1 \in \mathbb{Z}_N^*$ with the same size, as well as, an arbitrary ID^*, which it wants to attack and sends them to algorithm B. Then, the algorithm B searches the database $O_1(\cdot)$ to find the respective tuple, if the bit $b^* = 1$ then it rejects the query and the game will be finished. Otherwise if $b^* = 0$ then it generates a random bit $\gamma \in \{0, 1\}$, and finally calculates and sends $CT^* = (O_2({g_3}^r) \oplus M_\gamma, T)$ as the challenge to adversary A.
Assume the adversary is able to decrypt CT^* so if $T = {g_3}^{r(y_{c_1}s_1 + y_{c_2}s_2)} \bmod N$ it can obtain a bit γ, but if T be a random element of $\mathbb{G}$ the adversary can not obtain γ.
4. ***Phase 2.*** This phase is the same as phase 1, except that the adversary is not allowed to ask the challenger to decrypt CT^* or to generate a private key for ID^*.
5. ***Guess.*** A outputs a guess $\acute{\gamma} \in \{0, 1\}$. If $\gamma = \acute{\gamma}$, then $T = {g_3}^{r(s_1h_{i_1} + s_2h_{i_2})} \bmod N$ and if the $\gamma \neq \acute{\gamma}$, T is a random integer.

Analysis. This game will be successfully finished if the game is not terminated in challenge phase and all of the adversaries Key generation queries and Decryption queries are not rejected. Let probability of $b = 1$ be δ, then the probability of not rejecting none of adversaries n queries is $(\delta)^n$ and the probability of not terminating the game in challenge phase (i.e. $b^* = 0$) is $1 - \delta$. Thus with probability $\delta^n(1 - \delta)$ the game will be finished successfully. This probability is maximized at $\delta_{opt} = \frac{n}{n+1}$. If adversary A, by following the mentioned steps, can obtain the $Adv_A^{\mathcal{E}} = \epsilon$ in time t and by at most n queries and can break this scheme, then the algorithm B is able to break a $\tilde{n}$-TDDH problem with advantage $\epsilon\delta^n(1 - \delta) = \epsilon\ (\frac{n}{n+1})^n(1 - \frac{n}{n+1}) \approx \frac{\epsilon}{e(n+1)}$ and an additional constant time complexity. ■

5 Performance Analysis

It has been shown that pairing-based schemes for identity-based encryption are more efficient than other schemes [7]. Thus, in this section, we try to compare the performance of our proposed IBE scheme with pairing-based IBE schemes. Previous papers do not implement their schemes, and they just mention the number of required operations. So, for comparison, the actual implementation does not help us, and we just mention how many operations our schemes require.

Table 2 shows the computational costs of various operations in different groups, where $\mathbb{G}_1$ and $\mathbb{G}_2$ are domain bilinear groups, and $\mathbb{G}_t$ is the target

bilinear group. The RSA group generated by a composite modulus N is denoted by $\mathbb{Z}_N^*$ and $\mathbb{G}$ is a subgroup of $\mathbb{Z}_N^*$ with order q. We used point multiplication in the elliptic curve at 80-bit security as a unit for comparison. The computational cost of computing one pairing is 100 times more than the point multiplication in the elliptic curve and the cost of computing two pairings, that is called ratio pairing, is 1.2 times of the cost for one pairing [7]. There are two types of pairing implementations: MNT and SS curve, which have different domains and target groups [7]. The cost of general exponentiation in 80-bit security in $\mathbb{G}$ is 240 times higher than a multiplication in $\mathbb{G}$ [20, 18]. On the other hand, the cost of elliptic curve point multiplication at 80-bit security is 29 times higher than multiplications in $\mathbb{G}$ [20, 18]. As a result, the cost of general exponentiation in $\mathbb{G}$ is about 8 times higher than point multiplication in the elliptic curve. Also, the cost of fixed base exponentiation in $\mathbb{G}$ is 0.2 cost of general exponentiation [17]. To compare the efficiency of the proposed scheme, we add our results to the previously reported ones in [7]. Table 3 shows the comparison of our approach to BF [4], SK [8], and BB [9, 1] IBE schemes in terms of the type and number of operations. Finally, Table 4 compares the mentioned schemes in terms of the computational cost of encryption, decryption and private key extraction. Note that, we do not use the running time reported in [13], because they reported (in section 5.2) running time of point multiplication in the elliptic curve about 6 times more than modular exponentiation (in same security level), which does not make sense.

The computational cost of encryption of BB-IBE (with MNT pairing) is better than the proposed scheme, but the decryption cost of our approach is way better than previous schemes. That is 15 times faster than the BB approach. The ciphertext size of our approach for l-bit plaintext is only $2l$ bits.

Table 2. Relative Timings (arbitrary unit)* in 80-bit security

	Operation	$\mathbb{G}_1$	$\mathbb{G}_2$	$\mathbb{G}_t$	$\mathbb{G}$
Elliptic curve with SS paring	fix-base exp.	2	2	2	-
@ 80-bit Security	general exp.	10	10	10	-
	single pairing	-	-	100	-
	ratio pairings	-	-	120	-
Elliptic curve with MNT paring	fix-base exp.	0.2	8	2	-
@ 80-bit Security	general exp.	1	40	10	-
	single pairing	-	-	100	-
	ratio pairings	-	-	120	-
Modular exponentiation in $\mathbb{G}$	fix-base exp.	-	-	-	1.65
	fix-base exp.	-	-	-	8.27

Unit = point multiplication time on random curve E/F_q by random scalar in $\mathbb{Z}_p$, for prime $q \approx 2^{171}$ and $p \approx 2^{160}$.

Table 3. Number and type of cryptography operations

		BF-IBE	SK-IBE	BB-IBE	Our
Private key extraction	# fix-base exp.	-	$\mathbb{G}_2$	$\mathbb{G}_2\mathbb{G}_2$	-
	# general exp.	$\mathbb{G}_2$	-	-	-
Encryption	# fix-base exp.	$\mathbb{G}_{\mathbb{1}}\mathbb{G}_{\mathbb{1}}$	$\mathbb{G}_{\mathbb{1}}\mathbb{G}_{\mathbb{1}}\mathbb{G}_2$	$\mathbb{G}_1\mathbb{G}_1\mathbb{G}_1\mathbb{G}_t$	$\mathbb{G}\mathbb{G}$
	# pairings	$\mathbb{G}_t$	-	-	-
Decryption	# fix-base exp	$\mathbb{G}_1$	$\mathbb{G}_{\mathbb{1}}\mathbb{G}_{\mathbb{1}}$	$\mathbb{G}_1\mathbb{G}_t$	-
	# pairings	$\mathbb{G}_t$	$\mathbb{G}_t$	-	-
	# pairing ratios	-	-	$\mathbb{G}_t$	-
	# general exp.	-	-		$\mathbb{G}$

Table 4. Relative Timings (arbitrary unit)*

		Encryption	Decryption	Key Extraction
BF-IBE	SS	104	102	10
	MNT	100.4	100.2	40
SK-IBE	SS	6	104	2
	MNT	8.4	100.4	8
BB-IBE	SS	8	124	4
	MNT	2.6	122.2	16
Our		**3.3**	**8.27**	**Negligible**

Unit = point multiplication time on random curve E/F_q by random scalar in $\mathbb{Z}_p$, for prime $q \approx 2^{171}$ and $p \approx 2^{160}$.

6 Conclusion

In this paper, we presented a new IBE scheme based on the RSA modulus. The computation cost of this approach is two exponentiations for encryption and one exponentiation for decryption. Thus, the proposed scheme is more efficient than previous schemes. Especially, it is very cost-effective in key extraction and decryption. Furthermore, the proposed scheme converts l-bit plaintext to $2l$-bit ciphertext. This scheme is similar to the well-known ElGamal encryption algorithm, so it has some advantages over pairing-based approaches; for example, the proposed scheme could be used as a part of oblivious computation based protocols.

References

1. D. Boneh and X. Boyen. Efficient Selective Identity-Based Encryption Without Random Oracles. Journal of Cryptology (JOC), 24(4), pages 659-693, 2011.
2. D. Boneh, C. Gentry and M. Hamburg. Space-efficient identity based encryption without pairings. In FOCS, pages 647-657, 2007.
3. C. Cocks. An identity based encryption scheme based on quadratic residues. In IMA International Conference on Cryptography and Coding, volume 2260 of Lecture Notes in Computer Science, pages 360-363, Springer, 2001.
4. D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. In Advances in Cryptology - CRYPTO 2001, volume 2139 of Lecture Notes in Computer Science, pages 213-229, 2001.
5. A. Shamir. Identity-based cryptosystems and signature schemes. In Advances in Cryptology - CRYPTO 1984, volume 196 of Lecture Notes in Computer Science, pages 47-53, 1984.
6. W. Chen, "An IBE-based security scheme on Internet of Things," 2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems, Hangzhou, pages 1046-1049, 2012.
7. X. Boyen. A tapestry of identity-based encryption: practical frameworks compared. International Journal of Applied Cryptography, 1(1), pages 3–21, 2008.
8. R. Sakai and M. Kasahara. ID based cryptosystems with pairing on elliptic curve. IACR Cryptology ePrint Archive, 2003:54, 2003.
9. D. Boneh and X. Boyen. Efficient selective-ID secure identity-based encryption without random oracles. In Advances in Cryptology - EUROCRYPT 2004, volume 3027 of Lecture Notes in Computer Science, pages 223-238, 2004.
10. D. Boneh, B. Waters: Conjunctive, subset, and range queries on encrypted data. In TCC, volume 4392 of Lecture Notes in Computer Science, pages 535-554, 2007.
11. D. Boneh, G. Di Crescenzo, R. Ostrovsky, G. Persiano: Public-key encryption with keyword search. In Advances in Cryptology - EUROCRYPT 2004, volume 3027 of Lecture Notes in Computer Science, pages 506-522, 2004.
12. K. G. Paterson, S. Srinivasan. On the relations between non-interactive key distribution, identity-based encryption and trapdoor discrete log groups. In Designs, Codes and Cryptography, 52(2), pages 219-241, 2009.

13. M. Ramadan, Y. Liao, F. Li, S. Zhou and H. Abdalla. IBEET-RSA: Identity-Based Encryption with Equality Test over RSA for Wireless Body Area Networks. In Mobile Networks and Applications, 25, pages 223–233, 2019.
14. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In STOC, pages 197-206, 2008.
15. S. Agrawal, D. Boneh, and X. Boyen. Efficient lattice (H)IBE in the standard model. In Advances in Cryptology - Eurocrypt 2010, volume 6110 of Lecture Notes in Computer Science, pages 553-572, 2010.
16. S. Agrawal, D. Boneh, and X. Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In Advances in Cryptology - CRYPTO 2010, volume 6223 of Lecture Notes in Computer Science, pages 98-115, 2010.
17. J. Brown, J. Manuel Gonzalez Neito and C. Boyd. Efficient CCA-Secure Public-Key Encryption Schemes from RSA-Related Assumptions. In INDOCRYPT, volume 4329 of Lecture Notes in Computer Science, pages 176-190, 2006.
18. N. Koblitz, A. J. Menezes, and S. A.Vanstone. The State of Elliptic Curve Cryptography. In Designs, Codes and Cryptography, 19(2) pages 173-193, 2000.
19. NIST, Recommendation for Key Management — Part 1: General SP 800-57, May 2006, http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2-Mar08-2007.pdf.
20. Y. Chung, K. Huang, F. Lai, and T. Chen. ID-based Digital Signature Scheme on the Elliptic Curve Cryptosystem. In Computer Standards & Interfaces, 29(6), pages 601-604, 2007.
21. Y. Seurin. New Constructions and Applications of Trapdoor DDH Groups. In PKC, volume 7778 of Lecture Notes in Computer Science, pages 443-460, 2013.
22. J. Sylvester. Mathematical questions, with their solutions. Educational Times 41. pp. 21. 1884.
23. G. Tsudik and S. Xu. Accumulating Composites and Improved Group Singing. In Advances in Cryptology - Asiacrypt 2003, volume 2894 of Lecture Notes in Computer Science, pages 269-286, 2003.
24. J. H. Park, K. Lee and D. H. Lee. Efficient Identity-Based Encryption and Public-Key Signature from Trapdoor Subgroups. In IACR Cryptology ePrint Archive, 2016:500, 2016.
25. L. Hanzlik and K. Kluczniak. Security Analysis of ePrint Report 2016/500 "Efficient Identity-Based Encryption and Public-Key Signature from Trapdoor Subgroups". In IACR Cryptology ePrint Archive, 2016:512, 2016.
26. A. Menezes, P. Van Oorschot, S. Vanstone. Handbook of applied cryptography, pages 65-66, 1996.
27. X. Ding and G. Tsudik. Simple Identity-Based Cryptography with Mediated RSA. In Cryptographers Track at the RSA Conference - CT-RSA 2003, volume 2612 of Lecture Notes in Computer Science, pages 193-210, 2003.
28. S. Tessaro and D. A. Wilson. Bounded-Collusion Identity-Based Encryption from Semantically-Secure Public-Key Encryption: Generic Constructions with Short Ciphertexts. In Public-Key Cryptography - PKC 2014, volume 8383 of Lecture Notes in Computer Science, pages 257-274, 2014.
29. Y. Dodis, J. Katz, S. Xu, and M. Yung. Key-insulated public key cryptosystems. In Advances in Cryptology - EUROCRYPT 2002, volume 2332 of Lecture Notes in Computer Science, pages 65-82, 2002.

30. S. Goldwasser, A. B. Lewko, and D. A. Wilson. Bounded-collusion IBE from key homomorphism. In TCC 2012: 9th Theory of Cryptography Conference, volume 7194 of Lecture Notes in Computer Science, pages 564-581, 2012.