Journal of Computing and Security

January 2016, Volume 3, Number 1 (pp. 3–12)

http://www.jcomsec.org

A CCA2-Secure Incomparable Public Key Encryption Scheme

Bahman Rajabi [a], Ziba Eslami [a,b,*]

[a] *Department of Computer Science, Shahid Beheshti University, G.C.,Tehran,Iran*

[b] *Cyberspace Research Institute, Shahid Beheshti University, Tehran, Iran*

ARTICLE INFO.

Article history:
Received: 31 May 2016
Revised: 15 October 2016
Accepted: 05 December 2016
Published Online: 25 January 2017

Keywords:
Public Key Encryption, Anonymity, Key Privacy, Data Privacy, Incomparability

ABSTRACT

In 2003, Waters, Felten and Sahai introduced a novel cryptographic primitive called Incomparable Public Key cryptosystem to protect anonymity of message receivers in an untrusted network. In this setting, a receiver is allowed to create many anonymous identities for himself without divulging the fact that all these identities refer to the same receiver. Recently, Lee and Lim improved the solution of Waters et al. with a more efficient decryption process. Both of these schemes are based on the ELGamal encryption scheme and are CCA1-secure in the sense of data privacy and IK-CPA-secure in the sense of key privacy. In this paper, we employ the Cramer-Shoup encryption scheme to propose the first example of an incomparable public key encryption scheme which is CCA2-secure in data privacy and IK-CCA-secure in key privacy. Therefore, our scheme outperforms existing incomparable public key schemes in security properties.

1 Introduction

It is a well known fact that in the context of public key encryption, public keys are usually closely related to the identity of recipients. Therefore, a wide range of research in public key cryptography is done to devise cryptographic schemes which guarantee the anonymity of recipients against eavesdroppers as well as message senders [1–16]. One approach to achieve anonymity against eavesdroppers is using public key encryption schemes with ***key privacy*** property introduced in 2001 by Bellare et al.[17]. This property has important applications in building (receiver) anonymous channels, or privacy-enhanced authentication/signature schemes. Key privacy ensures that for a specific adversary, it is impossible to determine which of the keys has been used to encrypt a given ciphertext. Hence, key privacy provides some anonymity for receiver from the point of view of the adversary. However, this is not enough to guarantee recipient's anonymity against senders. In other words, if senders share the same public key to encrypt messages to be sent for an anonymous receiver, they can obtain some information to compromise the anonymity of that public key's owner. As an example, Bellare et al. showed that the basic ElGamal encryption achieves key privacy but if two recipients in a network use different cyclic groups then, anonymity can be compromised either due to the contents of messages or public keys themselves.

Another approach to guarantee the anonymity of recipients against eavesdroppers as well as traffic analysis is re-encryption of all the messages. In 2004, Golle et al. exploited the homomorphic property of ElGamal to propose a universal re-encryption which is a modification of the basic ElGamal encryption scheme [4]. The authors, suggested several applications of the universal re-encryption scheme including mix-nets and anonymizing bulletin board systems [3, 4]. The main

* Corresponding author.
Email addresses: b_rajabi@sbu.ac.ir (B. Rajabi), z_eslami@sbu.ac.ir (Z. Eslami)

ISSN: 2322-4460

idea of re-encryption is using a public key encryption scheme with key privacy property such that without any knowledge of the corresponding public key, a ciphertext C can be changed into another ciphertext C_1 and both of these ciphertexts can be decrypted to the same plaintext with the same secret key.

To achieve receiver's anonymity against senders, in 2003, Waters et al. proposed an '***incomparable***' public key scheme [15]. In this approach, a receiver can create as many anonymous identities (public keys) as desired and give them to different senders. It is even possible to give multiple anonymous identities to the same sender if the receiver is carrying on multiple independent conversations with that sender. This will prevent compromising receiver's anonymity against coalition of senders as well. In this approach, the recipient is allowed to issue several public keys, all related to his unique secret key. The main idea here is using a public key encryption scheme with the property of key privacy in such a way that it is computationally impossible to distinguish that two given public keys are related to the same recipient. To do so, Waters et al. introduced the notions of equivalent public keys, key incomparability and incomparable public key encryption scheme. Two public keys which correspond to the same secret key are called equivalent . A public key encryption has key incomparability if it is not possible to distinguish two equivalent public keys from two non-equivalent public keys. Finally a public key encryption scheme which has key incomparability is called incomparable public key encryption scheme.

We have grounds to prefer incomparable public key scheme over universal re-encryption scheme. In the universal re-encryption scheme , the senders know the (unique) public key of the intended recipient. Hence, a coalition of senders can be a threat to the anonymity of the recipient. Moreover, we need some trusted centers to change chiphertexts which leads to lose self-reliance in security. However, if each recipient is able to create a large number of anonymous identities such that no (feasible) entity is able to determine that those anonymous identities correspond to the same receiver, then all users in the system can be regarded as potential recipients of every ciphertext. This ability would further prevent them from aggregating the information they have about the receiver which is important in an environment where senders can find a little information about each receiver.

In Waters et al.'s scheme, every recipient in a multicast group must first completely decrypt a message in order to find out if it belongs to him or not. In 2011, Lee and Lim proposed a modification of Waters et al's scheme in which the number of computations that the recipient has to carry out is reduced while the security level remains unchanged. Here, the recipient can first determine whether a ciphertext is directed to him, and only if the direction is correct, he decrypts it [11]. Both Waters et al.'s and Lee and Lim's schemes are based on homomorphic encryption and therefore, their schemes can not achieve security level better than nonadaptive chosen ciphertext attacks ($CCA1$) in the sense of data privacy. On the other hand, both of these schemes are based on ElGamal encryption scheme which Bellare et al. [17] have shown to achieve indistinguishability of keys under chosen plaintext attacks ($IK-CPA$ security in the sense of key privacy). Therefor, both of these scheme has $IK-CPA$ security (see Lemma 1 of Section 6).

In this paper, we propose the first example of an incomparable public key encryption scheme which is secure against adaptive chosen ciphertext attacks ($CCA2$ secure) in data privacy and achieves indistinguishability of keys under adaptive chosen ciphertext attacks ($IK-CCA$) in key privacy. Our scheme is based on Cramer-Shoup encryption scheme [18, 19] and its decryption procedure is done in two steps as in [11]. Therefore, the proposed scheme outperforms existing incomparable public key encryption schemes either in terms of computation or security.

The paper is organized as follows. In Section 2, we provide notations, assumptions and definitions required in the rest of the paper. In Section 3, we review related work including existing incomparable public key encryption schemes together with a description of Cramer Shoup scheme. A general construction for an $HPKE$ with key incomparability is given is Section 4. Section 5 presents our incomparable public key encryption scheme and Section 6 provides an analysis of the scheme. Comparison with the existing literature is done in Section 7 and finally conclusions are provided in Section 8.

2 Preliminaries

In this section, we briefly provide the notations, definitions and assumptions used throughout the paper.

2.1 *Notations*

In Table 1, we use λ, pk, sk, m and c to denote the security parameter, the public key, the secret key, the plaintext and the ciphertext in an encryption scheme, respectively. By $(c_1, c_2, \ldots) \longleftarrow A(b_1, b_2, \ldots)$ we mean a deterministic algorithm A which takes $b_1, b_2, \ldots$ as input and produces $c_1, c_2, \ldots$ as output. In the case of a probabilistic algorithm we use the notation $\longleftarrow^R$.

Note that to encrypt a message m in an $HPKE$, we use its PKE to encrypt a random key K and then

Table 1. Table of notations.

Scheme	Abbreviation	Algorithms
Public Key Encryption Scheme	PKE	$(pk, sk) \longleftarrow KEYGEN(\lambda)$ $c \longleftarrow^R ENC(m, pk)$ $m \longleftarrow DEC(c, sk)$
Symmetric Key Encryption Scheme	SKE	$s \longleftarrow KEYGEN(\lambda)$ $c \longleftarrow^R ENC(m, s)$ $m \longleftarrow DEC(c, s)$
Hybrid Public Key Encryption Scheme From PKE, SKE	$HPKE$	$(pk, sk) \longleftarrow KEYGEN(\lambda)$ $(c_1, c_2) \longleftarrow^R ENC(m, pk) = (ENC_{PKE}(K, pk), ENC_{SKE}(m, K)$ $m \longleftarrow DEC((c_1, c_2), sk) = DEC_{SKE}(c_2, DEC_{PKE}(c_1, pk))$

encrypt m using SKE with the key K. A PKE which is used in the structure of a $HPKE$ is called ***key encapsulation mechanism*** (KEM) of that $HPKE$.

2.2 *Privacy notions*

In this paper, the following notions of privacy are considered.

2.2.1 *Notions of data privacy*

Data privacy of an encryption scheme has been well documented in the literature [8, 20]. We consider three types of adversarial attacks: chosen plaintext attacks (CPA), nonadaptive chosen chiphertext attacks $(CCA1)$ and adaptive chosen ciphertext attacks $(CCA2)$. In this section, we provide a breif description of these notions.

Consider the following experiment for any public key encryption scheme $\Pi = (Gen, Enc, Dec)$:

The CPA indistinguishability experiment $EXP^{CPA}_{\mathcal{A},\Pi}(\lambda)$:

(1) A key pair, public key pk and secret key sk is generated by running $Gen(1^\lambda)$.
(2) The adversary $\mathcal{A}$ is given input 1^λ and access to $Enc_{PKE}(-, pk)$. $\mathcal{A}$ outputs a pair of messages m_0, m_1.
(3) A random bit $b \leftarrow \{0, 1\}$ is chosen, and then a ciphertext $c \leftarrow Enc_{pk}(m_b)$ is computed and given to $\mathcal{A}$.
(4) The adversary $\mathcal{A}$ continues to have access to $Enc_{PKE}(-, pk)$, and outputs a bit b'.
(5) The output of the experiment is defined to be 1 if $b' = b$, and 0 otherwise. (In case $EXP^{CPA}_{\mathcal{A},\Pi}(\lambda) = 1$, we say that $\mathcal{A}$ succeeded .)

Definition 1. (CPA security) A public key encryption scheme $\Pi = (Gen, Enc, Dec)$ has indistinguishable encryptions under a chosen-plaintext attack (or is CPA-secure) if for all probabilistic polynomial-time adversaries $\mathcal{A}$ there exists a negligible function $negl$ such that

$$Pr[EXP^{CPA}_{\mathcal{A},\Pi}(\lambda) = 1] < 1/2 + negl(n),$$

where the probability is taken over the random coins used by $\mathcal{A}$, as well as the random coins used in the experiment.

If in step 2 of the above experiminent, the adversary $\mathcal{A}$ has access to the decryption oracle DEC_{sk}, we come up with the experiment $EXP^{CCA1}_{\mathcal{A},\Pi}(\lambda)$ and can define $CCA1$ security in the same manner. $CCA2$ security is achieved based on $EXP^{CCA2}_{\mathcal{A},\Pi}(\lambda)$ in which the adversary $\mathcal{A}$ has oracle access to DEC_{sk} in steps 2 and 4.

The relation between these security notions are as follows:

$$CPA \leq CCA1 \leq CCA2$$

In other words, if a scheme is $CCA2$ secure, then it is also $CCA1$ secure and if a scheme is $CCA1$ secure, then it is also CPA secure.

2.2.2 *Notions of key privacy*

The notion of key privacy in the public key encryption was introduced by Bellare et al.[17]. Key privacy provides anonymity of public keys. In the formalization of key privacy in the public key encryption setting, the adversary has access to a pair of public keys pk_0, pk_1 and a ciphertext c which is the encryption of an arbitrary message under one of these public keys. The possession of c should not give the adversary any advantage in determining which one of the keys has been used to create c.

Let Π be a PKE scheme and $\mathcal{A}_{CPA}$ be the adversary who is given two public keys. Consider the key privacy experiment $EXP^{IK-CPA}_{\mathcal{A}_{CPA},\Pi}(\lambda)$, played between $\mathcal{A}_{CPA}$ and a challenger, as follow:

- Two key pairs $(pk_0, sk_0), (pk_1, sk_1)$ are generated by running $KEYGEN(\lambda)$.
- The adversary $\mathcal{A}_{CPA}$ is given input λ, pk_0 and pk_1. It outputs a message m.
- The challenger chooses a random bit $b \in \{0, 1\}$, computes the ciphertext $c = ENC_{PKE}(m, pk_b)$ and gives c to $\mathcal{A}_{CPA}$.
- The adversary tries to guess which public key was used to encrypt m and he outputs a bit b'. The output of the experiment is defined to be 1 if $b' = b$, and 0 otherwise.

We define the advantage of the adversary via

$$Adv^{IK-CPA}_{\mathcal{A}_{CPA},\Pi}(\lambda) = |Pr[EXP^{IK-CPA}_{\mathcal{A}_{CPA},\Pi}(\lambda) = 1] - Pr[EXP^{IK-CPA}_{\mathcal{A}_{CPA},\Pi}(\lambda) = 0]|.$$

Definition 2. **($IK - CPA$ security)** The scheme Π has indistinguishability of keys under chosen-plaintext attacks (or is $IK - CPA$ secure) if the function $Adv^{IK-CPA}_{\mathcal{A}_{CPA},\Pi}(\lambda)$ is negligible for polynomial time adversary $\mathcal{A}_{CPA}$ whose time complexity is polynomial in λ.

Consider the experiment $EXP^{IK-CCA}_{\mathcal{A}_{CCA},\Pi}(\lambda)$ the same as $EXP^{IK-CPA}_{\mathcal{A}_{CPA},\Pi}(\lambda)$ except that in this case, the adversary $\mathcal{A}_{CCA}$ has access to decryption oracles DEC_{SK_0} and DEC_{SK_1} as well. It is mandated that $\mathcal{A}_{CCA}$ never queries DEC_{sk_0} or DEC_{sk_1} on the challenge ciphertext c.

Definition 3. **($IK - CCA$ security)** The scheme Π has indistinguishability of keys under chosen-cipherext attacks (or is $IK - CCA$ secure) if the function $Adv^{IK-CCA}_{\mathcal{A}_{CCA},\Pi}(\lambda))$ is negligible for any polynomial time adversary $\mathcal{A}_{CCA}$ whose time complexity is polynomial in λ.

It is therefore easy to see that we have

$$IK - CPA \leq IK - CCA.$$

In other words, if a scheme is $IK - CCA$ secure, then it is also $IK - CPA$ secure.

2.3 *Hardness Assumptions*

Security of our proposed scheme is based on the following assumptions.

Target Collision Resistant (TCR) Hash Function assumption: A family of hash functions is said to be collision resistant if upon drawing a function H at random from the family $\mathcal{H}$, it is infeasible for an adversary to find two different inputs x and y such that $H(x) = H(y)$.

A weaker notion is that of a *target collision resistant* family of hash functions. In this case, it should be infeasible for an adversary to choose an input x, draw a random hash function H, and then find a different input y such that $H(x) = H(y)$. Such hash function families are also called "universal one-way hash functions".

Decisional Diffie–Hellman (DDH) assumption Consider a (multiplicative) cyclic group G of order q and generator g. The DDH assumption states that, given g^a and g^b for uniformly and independently chosen $a, b \in Z_q$, the value g^{ab} "looks like" a random element in G. This intuitive notion is formally stated by saying that the following two probability distributions are computationally indistinguishable (in the security parameter $n = log(q)$).

- $T_1 = (g, g^a, g^b, g^{ab})$, where a and b are randomly and independently chosen from Z_q (T_1 is also known as DH-quadruple).
- $T_2 = (g, g^a, g^b, c)$, where a and b are randomly and independently chosen from Z_q and c is a random element in G.

3 Related Work

In this section, we first review existing incomparable public key encryption schemes,i.e. Waters et al.'s [15] and Lee and Lim's [11] schemes. We then provide details of the Cramer-Shoup's PKE which we later use for peresenting our proposed incomparable public key encryption scheme.

3.1 Waters et al's Public Key Encryption Scheme

Waters et al. PKE scheme is based on ELGamal PKE and consists of four algorithms (SETUP, KEYGEN, ENC, DEC) as follows ([15]).

SETUP: On input security parameter λ, the setup algorithm SETUP generates a secure safe prime p, where $p = 2q + 1$ for a prime number q, sets a cryptographically secure hash function H, sets a secure symmetric encryption scheme E, and outputs the system parameter $I = (p, q, H, E)$.

KEYGEN:

- On input system parameter I, generates a at random in Z^*_q, and sets the private key $sk = a$.
- On input private key sk, generates a generator $g \in Z^*_p$ of the order q at random, and sets a public key $pk = (g, g^a)$.

To obtain another equivalent public key for sk, the

recipient computes another generator $h = g^x$ for randomly chosen $1 < x < q$, issues the public key (h, h^a).

ENC: On input public key $pk = (g, g^a)$ and a message m, the encryption algorithm E chooses $r \in Z_q^*$ at random and outputs the ciphertext

$$C = ENC_{PKE}(m, pk) = (g^r, g^{ar}.K, H(r), E_{SKE}((r, (g, g^a), m), K)).$$

DEC: On input private key sk and a ciphertext (c_1, c_2, c_3, c_4), the decryption algorithm DEC

* Computes $K = c_3/g^a$.
* Computes $D_{SKE}(c_4, K) = (r', pk', m)$.
* Checks the equalities $H(r') = c_4$, $g^{r'} = c_1 \ mod \ p$ and the validity of the public key pk'.
* Outputs m as the plaintext only if all the checks are correct.

3.2 Lee and Lim's Public Key Encryption Scheme

Lee and Lim's PKE scheme consists of four algorithms (SETUP, KEYGEN, ENC, DEC) as follows ([11]).

SETUP: On input security parameter λ, the setup algorithm SETUP generates a secure safe prime p, where $p = 2q + 1$ for a prime number q, sets a cryptographically secure hash function H, sets a secure symmetric encryption scheme E, and outputs the system parameter $I = (p, q, H, E)$.

KEYGEN:

- On input system parameter I, generates a, b at random in Z_q^*, and sets the private key $sk = (a, b)$.
- On input private key sk, generates a generator $g \in Z_p^*$ of the order q at random, and sets a public key $pk = (g, g^a, g^b)$.

To obtain another equivalent public key for sk, the recipient computes another generator $h = g^x$ for randomly chosen $1 < x < q$, issues the public key (h, h^a).

ENC: On input public key $pk = (g, g^a)$ and a message m, the encryption algorithm E chooses $r \in Z_q^*$ at random and outputs the ciphertext

$$C = ENC_{PKE}(m, pk) = (g^r, g^{ar}, g^{br}.K, H(r), E_K(r, (g, g^a), m)).$$

DEC: On input private key sk and a ciphertext $(c_1, c_2, c_3, c_4, c_5)$, the decryption algorithm DEC, first checks the equality $c_1^a - c_2 = 0 \ mod \ p$, only if the equality holds, proceeds as follows:

* Computes $K = c_3/g^a$.
* Computes $D_{SKE}(c_5, K) = (r', pk', m)$.
* Checks the equalities $H(r') = c_4$, $g^{r'} = c_1 \ mod \ p$ and $g^{ar'} = c_2 \ mod \ p$ and the validity of the public key pk'.
* Outputs m as the plaintext only if all the checks are correct.

3.3 Cramer-Shoup's Public Key Encryption Scheme

Assume that G is a group with a subgroup $\widehat{G}$ of prime order q such that the DDH problem is hard in $\widehat{G}$. Also assume that H is a TCR hash function from G^3 to Z_q. The Cramer-Shoup PKE's algorithms are as follows ([18, 19]).

KEYGEN: The key generation algorithm on input of the security parameter λ, chooses random elements $x_1, x_2, y_1, y_2, z \in Z_q$ and $g_1, g_2 \in \widehat{G}$ such that $g_2 = g_1^w$ for some $w \in Z_q$. It computes $c = g_1^{x_1} g_2^{x_2}$, $d = g_1^{y_1} g_2^{y_2}$, $h = g_1^z$ and sets $sk = (x_1, x_2, y_1, y_2, z)$ and $pk = (g_1, g_2, c, d, h)$.

ENC: To encrypt a message $m \in G$, the encryption algorithm chooses $r \in Z_q$ at random. It computes $u_1 = g_1^r$, $u_2 = g_2^r$, $e = mh^r$, $\alpha = H(u_1, u_2, e)$ and $v = c^r d^{r\alpha}$. The ciphertext is (u_1, u_2, e, v).

DEC: To decrypt a chiphertext (u_1, u_2, e, v), the decryption algorithm computes $\alpha = H(u_1, u_2, e)$, and tests if $u_1^{x_1 + \alpha y_1} u_2^{x_2 + \alpha y_2} = v$. If this condition does not hold, the decryption algorithm outputs "reject"; otherwise, it outputs $m = \frac{e}{u_1^z}$.

4 General Construction of $HPKE$ with Key Incomparability

The aim of this section is to show that constructing an $HPKE$ with key incomparability and desired security properties, reduces to choosing a KEM with desired security properties and proving that this KEM has key incomparability property. Note that the key component of such an $HPKE$ is its KEM. This is because first, public keys in the $HPKE$ are generetaed by the key generation algorithm in its KEM. So, if there is a feasible secure approach for the KEM to create as many public keys as desired related to a secret key, then the approach can be used for the $HPKE$, too. Second, if a proper SKE is used together with the KEM to build the $HPKE$, then the resulting $HPKE$ inherits security properties (key privacy and data privacy) of the KEM as well. We state the following two lemmas to prove this. In subsequent sections, we prove that under certain hardness assumptions, the Cramer-Shoup PKE achieves key incomparibility and can be used as a KEM with this general construction.

Lemma 1. *If a PKE achieves some key privacy security, then any $HPKE$ that uses this PKE as its KEM, achieves the same key privacy security.*

Proof. Let Π be the $HPKE$ that uses Π', Π'' as its KEM and SKE, respectively. We show that if there

exists a polynomial time adversary $\mathcal{A}$ who can compromise key privacy of Π with a non-negligible advantage, then we can construct a polynomial time distinguisher $\mathcal{D}$ for compromising key privacy of Π' as well. Suppose that the distinguisher is given two public keys pk_0, pk_1, a message m and a chiphertext c_b. Its goal is to determine whether c_b is the encryption of m under the public key pk_0 or pk_1 in the setting Π'. $\mathcal{D}$ emulates the key privacy experiment for $\mathcal{A}$ in the manner described below and observes outputs of $\mathcal{D}$. If $\mathcal{A}$ outputs 0 then $\mathcal{D}$ guesses that c_b must be encryption of m under the public key pk_0, while if $\mathcal{A}$ outputs 1 then $\mathcal{D}$ guesses that c_b is the encryption of m under the public key pk_1 . In detail:
Distinguisher $\mathcal{D}$: $\mathcal{D}$ is given as input a quadruple (m, pk_0, pk_1, c_b).

- The distinguisher $\mathcal{D}$ runs $KEYGEN(\lambda)$ and gives pk_0, pk_1 to $\mathcal{A}$ in the setting Π .
- $\mathcal{A}$ generates a message m' and gives it to $\mathcal{D}$.
- $\mathcal{D}$ computes $c' = ENC_{SKE}(m', m)$ in setting Π'' ($\mathcal{D}$ uses m as a key) and gives $c = (c_b, c')$ to $\mathcal{A}$.
- $\mathcal{A}$ uses his ability to guess whether c is the encryption of m' under the public key pk_0 or pk_1 in the setting Π and outputs a bit b'.
- If $b' = 0$, $\mathcal{D}$ outputs "c_b is the encryption of m under the public key pk_0 ". Otherwise, $\mathcal{D}$ outputs "c_b is the encryption of m under the public key pk_1"

Note that what $\mathcal{D}$ outputs is in the setting Π'. From the above it is obvious that for $i = 0, 1$ we have: $c_b = Enc_{PKE}(m, pk_i)$ in setting Π' if and only if $c = ENC(m', pk_i)$ in setting Π.

Since the output of $\mathcal{D}$ is based on the output of $\mathcal{A}$ and $\mathcal{D}$ makes a true guess if and only if $\mathcal{A}$ makes a true guess, then the advantage of $\mathcal{A}$ in the key privacy experiment for the setting Π is equal to the advantage of $\mathcal{D}$ in the key privacy experiment for the setting Π'. Hence, if Π' (as a KEM) achieves $IK - CCA$ (or $IK - CPA$) key privacy security level then Π (as a $HPKE$ based on Π') preserves $IK - CCA$ (or $IK - CPA$) as well. □

Analogous to Lemma 1, we can relate the data privacy properties of a $HPKE$ to the key privacy of its KEM.

Lemma 2. *If KEM and SKE are $CCA2$ secure, then so is $HPKE$.(Theorem 5 [19])*

5 Our Proposed Scheme

We make novel use of the Cramer-Shoup cryptosystem and its properties to realize an incomparable public key encryption scheme. In our scheme, every recipient in the multicast group can identify the ciphertext directed to him in the multicast address more efficiently. The details of each algorithm of our PKE scheme are as follows:

SETUP: On input of the security parameter λ, the setup algorithm:

- Generates a group G with a subgroup $\widehat{G}$ of prime order q such that the DDH problem is hard in $\widehat{G}$. For example $\widehat{G}$ could be the subgroup of quadratic residues in Z_p^* for the prime number p. Here, we have $p = 2q + 1$ where q is also prime [21].
- Sets a family of TCR hash functions and chooses a random function H of this family.
- Sets a semantically secure symmetric encryption scheme $\mathcal{E} = \{E, D\}$.
- Outputs the system parameters $I = (p, G, H, \mathcal{E})$. These parameters are common to all recipients.

KEYGEN: The key generation algorithm consists of two phases defined as follows.

phase 1: On input of the system parameters I, the algorithm chooses $x_1, x_2, y_1, y_2, z \in Z_q^*$ at random, and sets the secret key $sk = (x_1, x_2, y_1, y_2, z)$.

phase 2: On input of the secret key sk, the algorithm generates random elements g_1 of order q and for some $w \in Z_q$, and sets a public key $pk = (g_1, g_2, c, d, h)$ such that $c = g_1^{x_1} g_2^{x_2}$, $d = g_1^{y_1} g_2^{y_2}$, $h = g_1^z$. To obtain another equivalent public key for sk, the recipient chooses $g_1' \in G$ of order q and $g_2' = g_1^{w'}$ for some $w' \in Z_q$ different from (g_1, g_2) and sets $pk' = (g_1', g_2', c', d', h')$ where $c' = (g_1')^{x_1}(g_2')^{x_2}$, $d' = (g_1')^{y_1}(g_2')^{y_2}$, $h' = (g_1')^z$.

The output is the multiple key $(sk, pk, pk', ...)$. Every recipient runs this algorithm to generate his secret key and some public keys (as many as needed).

ENC: On input of the public key $pk = (g_1, g_2, c, d, h)$ and a message m, the encryption algorithm chooses $r \in Z_q$ and $K \in G$ at random and outputs the chiphertext

$$C = (g_1^r, g_2^r, h^r K, c^r d^{r\alpha}, E_{SKE}((pk, r, m), K))$$

where $\alpha = H(g_1^r, g_2^r, h^r K)$.

DEC: On input of the secret key sk and a ciphertext $C = (c_1, c_2, c_3, c_4, c_5)$ the decryption algorithm DEC proceeds as follow:

step 1 Computes $\alpha = H(c_1, c_2, c_3)$. Then checks if $c_1^{x_1 + y_1\alpha} c_2^{x_2 + y_2\alpha} = c_4$ or not. If the equality does not hold, unhands the message, otherwise continues to the next step.

step 2

- Computes $K = \frac{c_3}{c_1^z}$.
- Computes $D_{SKE}(c_5, K) = (pk, r, m)$ where D is the decryption algorithm of $\mathcal{E}$.

All of the members in a multicast group share the same system parameters generated by the algo-

rithm **SETUP**. The recipient with the secret key (x_1, x_2, y_1, y_2, z) generates $g_1, g_2 \in G$ at random and stores the public key $(g_1, g_2, g_1^{x_1} g_2^{x_2}, g_1^{y_1} g_2^{y_2}, g_1^z)$ in his public key table to record it as being valid. To construct another public key from the secret key (x_1, x_2, y_1, y_2, z), the recipient generates another pair $s_1, s_2 \in \widehat{G}$, issues the public key $(s_1, s_2, s_1^{x_1} s_2^{x_2}, s_1^{y_1} s_2^{y_2}, s_1^z)$ with another identity and adds this public key in his public key table. When a recipient see a ciphertext in the address of the multicast group, he checks the first step of **DEC** to determine if the message belongs to him or not. If the equality doesn't hold, he rejects the message otherwise he saves that message to decrypt it in proper time.

6 Analysis of Our Scheme

In this section, we show that our proposed scheme achieves key incomparability, $CCA2$-security in data privacy and $IK-CCA$-security in key privacy. The security of our scheme is based on the hardness of the DDH problem.

6.1 Key Incomparability

Theorem 1. *If the DDH assumption holds in $\widehat{G}$, then our proposed scheme Π achieves key-incomparability.*

Proof. We show that if there exists a polynomial time adversary $\mathcal{A}$ who can distinguish whether two given public keys are related to the same secret key or not, then this adversary can be used to solve an instance of the DDH problem, thus contradicting the assumption of the theorem.
Consider the following experiment which is designed based on definitions in [11, 15].
public key incomparability experiment $PubK^{inc}_{\mathcal{A},\Pi}(\lambda)$

- The challenger runs $SETUP(\lambda)$ and generates two secret keys $sk = (x_1, x_2, y_1, y_2, z)$ and $sk' = (x_1', x_2', y_1', y_2', z')$ such that $sk \neq sk'$.
- The challenger generates two public keys pk, pk_0 related to the secret key sk and gives pk to the adversary $\mathcal{A}$.
- The challenger generates a public key pk_1 related to secret key sk' .
- The challenger chooses a random bit $t \in \{0, 1\}$ and gives pk_t to $\mathcal{A}$.
- $\mathcal{A}$ tries to find t and outputs a bit t'. $\mathcal{A}$ wins this game if $t' = t$.

We set $PubK^{inc}_{\mathcal{A},\Pi}(\lambda) = 1$ if $\mathcal{A}$ wins this game. Let ε be the advantage of $\mathcal{A}$ for the above game, then we have

$$\varepsilon = Pr(PubK^{inc}_{\mathcal{A},\Pi}(\lambda) = 1) - \frac{1}{2}$$

In the game $PubK^{inc}_{\mathcal{A},\Pi}(\lambda)$, the challenger has to choose $sk \neq sk'$. We set $E_i = \{(sk, sk') | sk_i \neq sk_i'\}$, $1 \leq i \leq 5$ where sk_i and sk_i' are the ith component of sk and sk', respectively. Then a selected pair (sk, sk') in the above game will be at least in one of E_is. Assume that $\mathcal{A}$ is an adversary who can win the game $PubK^{inc}_{\mathcal{A},\prod}$ with a non-negligible advantage ε . Then at least for one of E_is, $\mathcal{A}$ has an advantage more than $\frac{\varepsilon}{5}$. We now show how $\mathcal{A}$ can be used to construct a probabilistic polynomial time distinguisher $\mathcal{D}$ for the DDH problem. □

Suppose that the distinguisher is given a quadruple $T = (g, g^a, g^b, g^c)$ as input and its goal is to determine whether T is a Diffie-Hellman (DH) quadruple or not. $\mathcal{D}$ emulates the game $PubK^{inc}_{\mathcal{A},\Pi}(\lambda)$ for $\mathcal{A}$ in the manner described below, and observes what $\mathcal{D}$ outputs. In each case, if $\mathcal{A}$ outputs 0 then $\mathcal{D}$ guesses that T must be a DH quadruple, while if $\mathcal{A}$ outputs 1 then $\mathcal{D}$ guesses that T is not a DH quadruple. In detail:
Distinguisher $\mathcal{D}$: A quadruple $T = (g, g^a, g^b, g^c)$ is given as input .

- $\mathcal{D}$ runs SETUP(λ), chooses $I \in \{1, 2, 3, 4, 5\}$ at random and follows the case I.
  case 1 $(x_1 \neq x_1')$: In this case, $\mathcal{D}$ chooses $x_2, y_1, y_2, z \in Z_q$, $g_2 \in G$ at random and gives
  $$(g, g_2, g^a g_2^{x_2}, g^{y_1} g_2^{y_2}, g^z)$$
  to $\mathcal{A}$ as pk (indeed $\mathcal{D}$ sets $sk = (a, x_2, y_1, y_2, z)$ without knowing the value of a). $\mathcal{D}$ chooses $r \in Z_q$ at random and gives
  $$(g^b, g_2^r, g^c g_2^{rx_2}, g^{by_1} g_2^{ry_2}, g^{bz})$$
  to $\mathcal{A}$ as pk_t.
  case 2 $(x_2 \neq x_2')$: In this case, $\mathcal{D}$ chooses $x_1, y_1, y_2, z \in Z_q$, $g_2 \in G$ at random and gives
  $$(g_2, g, g^{x_1} g^a, g_2^{y_1} g^{y_2}, g_2^z)$$
  to $\mathcal{A}$ as pk (indeed $\mathcal{D}$ sets $sk = (x_1, a, y_1, y_2, z)$). $\mathcal{D}$ chooses $r \in Z_q$ at random and gives
  $$(g_2^r, g^b, g_2^{rx_1} g^c, g_2^{ry_1} g^{by_2}, g_2^{bz})$$
  to $\mathcal{A}$ as pk_t.
  case 3 $(y_1 \neq y_1')$: In this case, $\mathcal{D}$ chooses $x_1, x_2, y_2, z \in Z_q$, $g_2 \in G$ at random and gives
  $$(g, g_2, g^{x_1} g_2^{x_2}, g^a g_2^{y_2}, g^z)$$
  to $\mathcal{A}$ as pk (indeed $\mathcal{D}$ sets $sk = (x_1, x_2, a, y_2, z)$). $\mathcal{D}$ chooses $r \in Z_q$ at random and gives
  $$(g^b, g_2^r, g^{bx_1} g_2^{bx_2}, g^c g_2^{ry_2}, g^{bz})$$
  to $\mathcal{A}$ as pk_t.
  case 4 $(y_2 \neq y_2')$: In this case, $\mathcal{D}$ chooses $x_1, x_2, y_1, z \in Z_q$, $g_2 \in G$ at random and gives
  $$(g_2, g, g^{x_1} g^{x_2}, g_2^{y_1} g^a, g_2^z)$$

to $\mathcal{A}$ as pk (indeed $\mathcal{D}$ sets $sk = (x_1, x_2, y_1, a, z)$). $\mathcal{D}$ chooses $r \in Z_q$ at random and gives

$$(g_2^r, g^b, g_2^{rx_1} g^{bx_2}, g_2^{ry_1} g^c, g_2^{bz})$$

to $\mathcal{A}$ as pk_t.

case 5 ($z \neq z'$) In this case, $\mathcal{D}$ chooses $x_1, x_2, y_1, y_2 \in Z_q$, $g_2 \in G$ at random and gives

$$(g, g_2, g^{x_1} g_2^{x_2}, g^{y_1} g_2^{y_2}, g^a)$$

to $\mathcal{A}$ as pk (indeed $\mathcal{D}$ sets $sk = (x_1, x_2, y_1, y_2, a)$). $\mathcal{D}$ chooses $r \in Z_q$ at random and gives

$$(g^b, g_2^r, g^{bx_1} g_2^{rx_2}, g^{by_1} g_2^{ry_2}, g^c)$$

to $\mathcal{A}$ as pk_t.

- $\mathcal{A}$ outputs a bit t.
- If $t = 0$, $\mathcal{D}$ outputs "T is a DH quadruple" and if $t = 1$, $\mathcal{D}$ outputs "T is not a DH quadruple".

For a given T, $\mathcal{D}(T) = 1$ if and only if $\mathcal{D}$ truly guess whether T is a DH quadruple or not. Since $\mathcal{D}$ outputs T is a DH quadruple if and only if $\mathcal{A}$ outputs 0, then for the case i, we have

$$Pr(\mathcal{D}\ outputs\ "T\ is\ a\ DH\ quadruple"|I = i) = Pr(\mathcal{A}\ outputs\ 0|E_i),$$

where the probability is conditioned on the event that case i is selected by $\mathcal{D}$. But for each of the above cases we have pk and pk_t are related to the same secret key if and only if $c = ab$ if and only if T is a DH quadruple. Then

$$pr(\mathcal{D}(T) = 1|I = i) = pr(PubK^{inc}_{\mathcal{A},\Pi}(\lambda) = 1|E_i).$$

On the other side, we have

$$\begin{aligned}
&Pr(\mathcal{D}(T) = 1) - \frac{1}{2} \\
&= \sum_{i=1}^{5} Pr(I = i) Pr(\mathcal{D}(T) = 1|I = i) \ - \frac{1}{2} \\
&= \sum_{i=1}^{5} Pr(I = i) Pr(PubK^{inc}_{\mathcal{A},\Pi}(\lambda) = 1|E_i) \ - \frac{1}{2} \\
&= \frac{1}{5} \sum_{i=1}^{5} Pr(PubK^{inc}_{\mathcal{A},\Pi}(\lambda) = 1|E_i) \ - \frac{1}{2} \\
&= \frac{1}{5} \sum_{i=1}^{5} (Pr(PubK^{inc}_{\mathcal{A},\Pi}(\lambda) = 1|E_i) \ - \frac{1}{2}) \geq \frac{1}{5} \times \frac{\varepsilon}{5} \\
&= \frac{\varepsilon}{25}.
\end{aligned}$$

Therefore, if ϵ is non-negligible (*i.e.*, our proposed scheme does not achieve key-incomparability), the distinguisher $\mathcal{D}$ can solve the DDH problem with non-negligible advantage $\frac{\varepsilon}{25}$ which is clearly a contradiction to the DDH assumption in G.

6.2 Key Privacy

Key privacy plays an important role in providing anonymity of public keys. Hence, we have to show that our proposed scheme achieves some key privacy security levels. The aim of this section is to prove that our proposed scheme is $IK - CCA$ secure. To do so, we note that our proposed scheme is a hybrid scheme based on Cramer-Shoup encryption scheme as a KEM. Then, we use Lemma 1 and the following theorem which has been proved by Bellare et al. in [17] to establish our claim.

Theorem 2. *Let G be a group with a subgroup $\widehat{G}$ of prime order q and let CS be the associated Cramer-Shoup scheme. If the DDH problem is hard in $\widehat{G}$, then CS is anonymous in the sense of $IK-CCA$. (theorem 3.2 [17])*

Therefore, the next theorem follows directly from Lemma 1 and Theorem 2.

Theorem 3. *If the DDH problem is hard in G, then the proposed scheme is $IK - CCA$-secure in the sense of key privacy.*

6.3 Data Privacy

In our scheme, we have adopted the idea of Waters et al.'s scheme to Cramer-Shoup's encryption scheme. This is achieved through the following lemma of Cramer and Shoup.

Lemma 3. *If the DDH assumption holds for G and the TCR assumption holds for H, then Cramer-Shoup encryption scheme is $CCA2$ secure.(Theorem 2 [19])*

We used Cramer and Shoup's PKE as the KEM of our $HPKE$ and a family of semantically secure SKE then the following theorem results from the Lemmas 3, 2.

Theorem 4. *Our proposed scheme is $CCA2$ secure.*

7 Comparison

In this section, we compare the proposed scheme with the existing incomparable public key encryption schemes. The comparison is done in terms of security properties as well as their efficiency in the decryption phase. As for key privacy, both Waters et al. and Lee and Lim's schemes are $IK - CPA$-secure, however, the proposed scheme performs better in this respect and is $IK - CCA$-secure. The proposed scheme outperforms Waters et al. and Lee and Lim's schemes in data privacy and achieves $CCA2$ security. The proposed scheme performs efficient decryption the same as that of Lee and Lim. The results are summarized in Table 2.

Table 2. Comparison of the existing incomparable public key encryption schemes.

Scheme	Incomparability	Efficient Decryption	Key privacy	Data Privacy
Waters et al. [15]	Yes	No	$IK-CPA$	$CCA1$
Lee and Lim[11]	Yes	Yes	$IK-CPA$	$CCA1$
Our Scheme	Yes	Yes	$IK-CCA$	$CCA2$

8 Conclusion

Existing incomparable public key cryptosystems proposed by Waters et al. and Lee and Lim are based on the ElGamal encryption scheme. Both of these schemes have $IK-CPA$ (indistinguishability of keys under chosen plaintext attacks) in key privacy. On the other hand, The homomorphic property of the ElGamal implies that these schemes can not achieve security level better than $CCA1$. In this paper, we employ the Cramer-Shoup encryption scheme to propose an incomparable public key encryption scheme which achieves $CCA2$ security in data privacy and has $IK-CCA$ (indistinguishability of keys under adaptive chosen ciphertext attacks) in key privacy.

References

[1] Hung-Yu Chien. Improved Anonymous Multi-receiver Identity-Based Encryption. *The Computer Journal*, 55(4):439–446, 2012. doi: 10.1093/comjnl/bxr086. URL `http://comjnl.oxfordjournals.org/content/55/4/439.abstract`.

[2] Chun-I Fan and Yi-Fan Tseng. Anonymous multi-receiver identity-based authenticated encryption with cca security. *Symmetry*, 7 (4):1856–1881, 2015. ISSN 2073-8994. doi: 10.3390/sym7041856. URL `http://www.mdpi.com/2073-8994/7/4/1856`.

[3] Philippe Golle, Stanislaw Jarecki, and Ilya Mironov. *Cryptographic Primitives Enforcing Communication and Storage Complexity*, pages 120–135. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. ISBN 978-3-540-36504-4. doi: 10.1007/3-540-36504-4_9. URL `http://dx.doi.org/10.1007/3-540-36504-4_9`.

[4] Philippe Golle, Markus Jakobsson, Ari Juels, and Paul Syverson. *Universal Re-encryption for Mixnets*, pages 163–178. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. ISBN 978-3-540-24660-2. doi: 10.1007/978-3-540-24660-2_14. URL `http://dx.doi.org/10.1007/978-3-540-24660-2_14`.

[5] Graeme Horsman and Lynne R. Conniss. An investigation of anonymous and spoof {SMS} resources used for the purposes of cyberstalking. *Digital Investigation*, 13: 80 – 93, 2015. ISSN 1742-2876. doi: http://dx.doi.org/10.1016/j.diin.2015.04.001. URL `http://www.sciencedirect.com/science/article/pii/S1742287615000419`.

[6] SK Hafizul Islam, Muhammad Khurram Khan, and Ali M. Al-Khouri. Anonymous and provably secure certificateless multireceiver encryption without bilinear pairing. *Security and Communication Networks*, 8(13):2214–2231, 2015. ISSN 1939-0122. doi: 10.1002/sec.1165. URL `http://dx.doi.org/10.1002/sec.1165`.

[7] Marzieh Ispareh and Behrouz Tork Ladani. A Conceptual Framework for Specification, Analysis, and Design of Anonymity Services. In *Proceedings of the 2009 EDBT/ICDT Workshops*, EDBT/ICDT '09, pages 131–138, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-650-2. doi: 10.1145/1698790.1698812. URL `http://doi.acm.org/10.1145/1698790.1698812`.

[8] Jonathan Katz and Yehuda Lindell. *Introduction to modern cryptography*. CRC press, 2014.

[9] Markulf Kohlweiss, Ueli Maurer, Cristina Onete, Björn Tackmann, and Daniele Venturi. *Anonymity-Preserving Public-Key Encryption: A Constructive Approach*, pages 19–39. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-39077-7. doi: 10.1007/978-3-642-39077-7_2. URL `http://dx.doi.org/10.1007/978-3-642-39077-7_2`.

[10] Guido Lang and Tamilla Mavlanova. Perceptions and Use of Anonymous Communication across Cultures. *Journal of International Technology and Information Management*, 24(1):4, 2015.

[11] Hyang-Sook Lee and Seongan Lim. An efficient incomparable public key encryption scheme. *Information Sciences*, 181(14):3066 – 3072, 2011. ISSN 0020-0255. doi: http://dx.doi.org/10.1016/j.ins.2011.03.009. URL `http://www.sciencedirect.com/science/article/pii/S002002551100137X`.

[12] Y. Sreenivasa Rao and Ratna Dutta. *Recipient Anonymous Ciphertext-Policy Attribute Based Encryption*, pages 329–344. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

ISBN 978-3-642-45204-8. doi: 10.1007/978-3-642-45204-8_25. URL http://dx.doi.org/10.1007/978-3-642-45204-8_25.

[13] Farahnaz Rezaeian Zadeh and Shohreh Ajoudanian. A Novel Solution for Author Attribution Problem in Anonymous E-mail. *Journal of Computing and Security*, 1(4), 2015. ISSN 2383-0417. URL http://jcomsec.org/index.php/JCS/article/view/95.

[14] Y. M. Tseng, Y. H. Huang, and H. J. Chang. CCA-secure Anonymous Multi-receiver ID-based Encryption. In *2012 26th International Conference on Advanced Information Networking and Applications Workshops*, pages 177–182, March 2012. doi: 10.1109/WAINA.2012.50.

[15] Brent R. Waters, Edward W. Felten, and Amit Sahai. Receiver Anonymity via Incomparable Public Keys. In *Proceedings of the 10th ACM Conference on Computer and Communications Security*, CCS '03, pages 112–121, New York, NY, USA, 2003. ACM. ISBN 1-58113-738-9. doi: 10.1145/948109.948127. URL http://doi.acm.org/10.1145/948109.948127.

[16] Jianhong Zhang and Jian Mao. An improved anonymous multi-receiver identity-based encryption scheme. *International Journal of Communication Systems*, 28(4):645–658, 2015. ISSN 1099-1131. doi: 10.1002/dac.2693. URL http://dx.doi.org/10.1002/dac.2693.

[17] Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. *Key-Privacy in Public-Key Encryption*, pages 566–582. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-540-45682-7. doi: 10.1007/3-540-45682-1_33. URL http://dx.doi.org/10.1007/3-540-45682-1_33.

[18] Ronald Cramer and Victor Shoup. *A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack*, pages 13–25. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998. ISBN 978-3-540-68462-6. doi: 10.1007/BFb0055717. URL http://dx.doi.org/10.1007/BFb0055717.

[19] Ronald Cramer and Victor Shoup. Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack. *SIAM Journal on Computing*, 33(1):167–226, 2003. doi: 10.1137/S0097539702403773. URL http://dx.doi.org/10.1137/S0097539702403773.

[20] Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. *Relations among notions of security for public-key encryption schemes*, pages 26–45. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998. ISBN 978-3-540-68462-6. doi: 10.1007/BFb0055718. URL http://dx.doi.org/10.1007/BFb0055718.

[21] Dan Boneh. *The Decision Diffie-Hellman problem*, pages 48–63. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998. ISBN 978-3-540-69113-6. doi: 10.1007/BFb0054851. URL http://dx.doi.org/10.1007/BFb0054851.

Bahman Rajabi received his B.S. degree in Mathematics in 2007 from Shahrood University, Semnan, Iran. In 2010, he received his M.S. degree in Mathematics from Shahid Beheshti University, Tehran, Iran. He is currently a Ph.D. student in the Department of Mathematic, Shahid Beheshti University. His research interests include cryptographic protocols and network security.

Ziba Eslami received her B.S., M.S., and Ph.D. in Applied Mathematics from Tehran University in Iran. She received her Ph.D. in 2000. From 1991 to 2000, she was a resident researcher in the Institute for Studies in Theoretical Physics and Mathematics (IPM), Iran. During the academic years 2000-2003, she was a Post Doctoral Fellow in IPM. She served as a non-resident researcher at IPM during 2003-2005. Currently, she is an associate professor in the Department of Computer Science at Shahid Beheshti University in Iran. Her research interests include design theory, combinatorial algorithms, cryptographic protocols, and steganography.