# AN APPROACH TO REMOVE KEY ESCROW PROBLEM IN IDENTITY-BASED ENCRYPTION FROM PAIRING

***Dissertation submitted to Jawaharlal Nehru University***
***in partial fulfillment of the requirements***
***for the award of the degree of***

**MASTER OF TECHNOLOGY**
**IN**
**COMPUTER SCIENCE AND TECHNOLOGY**
**BY**

**MAHENDER KUMAR**
**13/10/MT/016**

Under
the Supervision of
**Prof. C. P. KATTI**

**SCHOOL OF COMPUTER & SYSTEMS SCIENCES**
**JAWAHARLAL NEHRU UNIVERSITY**
**NEW DELHI-110067**
**INDIA**
**2015**

SCHOOL OF COMPUTER & SYSTEMS SCIENCES
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI, 110067 (INDIA)

# CERTIFICATE

This is to certify that the dissertation entitled **"An Approach to Remove Key Escrow Problem in Identity-based Encryption From Pairing"** is being submitted by **Mr. Mahender Kumar,** to **School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi-110067, India** in the partial fulfillment of the requirements for the award of the degree of **Master of Technology** in **Computer Science and Technology**. This work has been carried out by him in the School of Computer and Systems Sciences under the supervision of Prof. **C. P. Katti**. The matter personified in the dissertation has not been submitted for the award of any other degree or diploma.

Katti
Prof. C. P. Katti
(Supervisor)

Katti
Dean, SC&SS
Jawaharlal Nehru University
New Delhi, India

Dean
School of Computer & Systems Sciences
Jawaharlal Nehru University
New Delhi-110067

**SCHOOL OF COMPUTER & SYSTEMS SCIENCES**
**JAWAHARLAL NEHRU UNIVERSITY**
**NEW DELHI, 110067 (INDIA)**

# DECLARATION

This to certify that the dissertation work entitled **"An Approach to Remove Key Escrow Problem in Identity-based Encryption From Pairing"** in partial fulfillment for the requirements for the degree of **"Master of Technology** in **Computer Science and Technology"** and submitted to School of Computer & Systems Sciences, Jawaharlal Nehru University, New Delhi-110067, India, is the authentic record of my own work carried out during the time of Master of Technology under the supervision of **Prof. C. P. Katti**. This dissertation comprises only my original work.

The matter personified in the dissertation has not been submitted for the award of any other degree or diploma.

Mahenderkumar

MAHENDER KUMAR

**(Student)**

*Dedicated to*

*My*

*Beloved Late Grandfather….*

# ACKNOWLEDGEMENT

I would never have been able to finish my dissertation without the guidance of my research Lab members, help from friends, and support from my family and relatives.

First of all, I would like to express my deepest gratitude to my honorable supervisor and Dean, **Prof. C. P. Katti**, School of Computer and System Science, Jawaharlal Nehru University, New Delhi, for his excellent guidance, caring, patience, and encouragement. He has been an inspiration to me and a great teacher. I would like to very gratefulness towards him for providing me with an excellent atmosphere for doing research. It would have been impossible to complete this dissertation without his valuable support and patience.

Secondly, I would like to thank **Prof. P.C. Saxena (Emeritus)**, School of Computer and System Science, Jawaharlal Nehru University, New Delhi, who let me experience the research of freshwater mussels in the field and practical issues beyond the textbooks, patiently corrected my writing and motivationally supported my research.

Thirdly, I would like thank to my lab mates for fruitful support and encouragement thought-out my M. Tech dissertation. Impressive thanks to **Mr. J. K. Verma, R. A. Haidri, Mr. Ashok Kumar, Mr. Vineet Anand, Mr. Pankaj Kumar** and **Mr. Kunal Bhaskar** for providing me for their indiscipline suggestion and motivation with an excellent atmosphere for doing research.

Additionally, I would like to thank my friends and colleague for their support and care helped me overcome setbacks and stay focused on my graduate study. Special thanks to **Mr. Kashif Nawaz, Mr. Ravi Shankar Soni, Mr. Bhaskar Prasad, Mr. Mayank Gupta, Ms. Neha Anand, Mr. Tarun Kumar Gupta, Mr. Sunil Kumar, Mr.**

**Sudhakar, Ms. Ruby Rani, Ms. Monika Yadav** and all friends. I greatly value their friendship and I deeply appreciate their belief in me. I am also grateful to **Mr. Asif Ali**, who always willing to help, cheering me up and stood by me through the good and wrong time.

I shall ever remain obliged to the faculty members and staff of SC &SS, Jawaharlal Nehru University, New Delhi for their cooperation and the kindness extended to me during the completion of this dissertation work. I would like to thank **Robin canteen and their Staff**, for their "*Tea, Pakodas and Lunch*", where the most part of my thesis work was discussed with my friends. I am very thankful for their good serving nature.

I sincerely obliged to my Grandfather **Late. Sh. Nathi Singh**, whose Spiritual-Talk, Patience, Simplicity, Consistency, Encouragement, Pleasing, Bliss and Love have been continuously reflecting to me though my family, I would also like to thank my **Parents, elder Brother and Sister-In-Law, two elder Sisters, two Brother-In-Law, and Nephew** and **Nieces**. They were always supporting me, motivating me and encouraging me with their best wishes and Love.

*Mahender Kumar…*

# ABSTRACT

One of the main problems in the cryptosystem is the key distribution over an unsafe network. ID-based encryption has many advantages over the public key cryptosystem in key distribution, but they also suffer from the inherited key escrow problem. Several approaches are proposed to remove the key escrow problem. In multiple authority approaches, private key generation is distributed to the multiple authorities. In CL-PKC scheme, user-chosen secret information keeping Key privacy, is also a simple and efficient solution, but does not preserve the advantage ID-based encryption. Nevertheless, these approaches solve the key escrow problem, but it becomes a new problem to the democratic world. Since, PKG or government has no control over the user's private key. They may not take action against the unlawful use of private keys.

To overcome above issues, we proposed an efficient democratic identity-based encryption model that attains a balance between the government and the users where the government and people enjoying their rights. The government has rights to monitor the user's unlawful message. On the other hand; the user has the right to privacy on their lawful message. In proposed scheme, also to PKG one more entity (PKPO) is used. User's partial key is escrowed at PKG and the partial key is escrowed at PKPO. PKPO is introducing to provide the privacy service to the user by providing their signature in a confused manner. Only that user who has a secret info can unlock and extract his private key. Furthermore, we proof that proposed scheme is secure against an IND-CCA of Type I and Type II adversary attacks with their proofs in Theorem 1. It is assumed that two entities never collude each other.  Otherwise, the malicious entity will be caught by the judge on a legal complaint by guanine entity.  Finally, we shows that the proposed model is very efficient for low consumption devices because it take less computation cost on the client side, and overload shifted to the server side (PKG) on the cloud. As a result, the present scheme is environment-friendly, practically applicable and instantly prepared to use.

At the end, we conclude the thesis by intensify that proposed scheme shall apply in wireless network, email application.  Finally, we discussed the future scope of the thesis in the signature scheme and authenticated key agreement protocol.

# CONTENTS

**CHAPTER 1:**

CHAPTER 2:

CHAPTER 3:

**CHAPTER 4:**

**CHAPTER 5:**

**CHAPTER 6:**

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF SYMBOLS AND NOTATIONS

| | |
|---|---|
| m | Message |
| C | Ciphertext |
| $\mathbb{Z}_n$ | Set of integers modulo n |
| $(\mathbb{G}, \bullet)$ | Algebraic group with respect to the set $\mathbb{G}$ and the binary operation |
| $<\mathbb{G}\ \bullet, \blacksquare>$ | Cyclic Group with respective to the set $\mathbb{G}$ and two binary operation |
| $\mathbb{G}$ | Group $<\mathbb{G}, \bullet>$ |
| $\mathbb{H}$ | Subgroup of group $\mathbb{G}$ |
| $\mathbb{R}$ | Ring $< \mathbb{R}, \bullet, \blacksquare >$ |
| $\mathbb{F}$ | Field $< \mathbb{F}, \bullet, \blacksquare >$ |
| $\bullet$ and $\blacksquare$ | Binary operation |
| GF(p) | Finite field of order p |
| H (x) | Hash function with bit sequence w as input |
| $S_A$ (x) | Signature of entity A on message m |
| $e : G_1 \times G_2 \rightarrow G_T$ | Bilinear map |

| | |
|---|---|
| $v \oplus w$ | Bit by bit XOR operation of v and w |
| $\perp$ | Invalid output |
| U, P, Q | Points on an elliptic curve |
| $E_k$ (m) | Symmetric encryption of the message m under session key k |
| $D_k$ (C) | Symmetric decryption of the ciphertext c under session key k |
| k | Generic symmetric session key |
| x, y | Binary bit sequences |
| $\{x \parallel y\}$ | Concatenated bit sequences |
| $\{0, 1\}^n$ | Binary bit sequence of length n |
| $\{0, 1\}^*$ | Binary bit sequence of variable length |
| $usk_{Pr}$ | User's Private/secret Key generated by the user |
| $usk_{Pub}$ | User's Public Key computed by the user |
| $pkg_{Pr}$ | PKG's Private/secret Key |
| $pkg_{Pub}$ | PKG's Public Key |
| $pkPo_{Pr}$ | PKPO's Private key |
| $pkpo_{Pub}$ | PKPO's Public key |
| $D_{ID}$ | User's private key computed by PKG |
| $k$ | Security parameter |
| $\in$ | Belong to |
| *params* | Parameters that publically published to all |
| adv | Advantage for adversary to win game. |
| $\epsilon$ | Negligible function |
| $Cert_{ID}$ | certificate issued by PKG to user whose identity ID |

| | |
|---|---|
| CA | Certification authority |
| $\mathcal{A}_I$ | Type I IND-CPA adversary |
| $\mathcal{A}_{II}$ | Type II IND-CPA adversary |

# LIST OF ABBREVATIONS

| | |
|---|---|
| ID | User’s Identity |
| IBE | Identity based encryption |
| IBS | Identity based encryption |
| PKG | private Key Generation |
| HIBE | Hierarchical IBE |
| DLP | Discrete Logarithm problem |
| DDHP | Decision Diffie-Hellman problem |
| BDHP | Bilinear Diffie-Hellman problem |
| CDHP | Computational Diffie-Hellman problem |
| GDHP | Gap Diffie-Hellman problem |
| PKI | Public key infrastructures |
| IND-CPA | Indistinguisablity chosen ciphertext attack |
| IND-CCA | Indistinguisablity chosen palintext attack |
| ECC | Elliptic Curve Cryptography |
| RSA | Rivest Shamir Adleman Algorithm |
| DKG | Distributed key Generation |
| SSN | Social Security Number |
| ANO-IBE | Anonymous Identity-Based Encryption |
| ANO-CCA | Anonymous chosen ciphertext attack |

| | |
|---|---|
| CL-PKC | Certificate-less Public Key Cryptosystem |
| CB-PKC | Certificate-based Public Key Cryptosystem |
| SCS | Self Certified Scheme |
| V-IBE | Variant of Identity based encryption |
| M-IBE | Modified Identity based encryption free from key escrow problem |
| PKPO | Private Key Privacy organization |
| KeyGen | Key Generation |
| RO | Random Oracle |
| BasicM-IBE | Proposed Basic model of Identity-based encryption free from Key Escrow Problem |
| FullM-IBE | Proposed Full model of Identity-based encryption free from Key Escrow Problem secure against IND-CCA attack. |

Chapter

# 1 Introduction

In late 50's and 60's, only government and defense had a need for cryptography. Nowadays, the Internet becomes a primary need in our daily life. For example, exchange of information, text messages and video message over the network is very common. Today, every sixth person in the world is available on the internet. Each user wants to work on the web by e-commerce, e-mail, e-transaction, chat on social networking site, online shopping, etc. Thus, the world is becoming a virtual network where people can communicate over the internet. Since, information transmits over the insecure network. Therefore, confidentiality is the main issue while the message is sent from one device to another. A user who shares his information over the internet may wish that his information should not disclose to the unauthorized person. Apart from confidentiality, the user requires privacy and authenticity. The primary goal of cryptography is to provide the integrity and confidentiality [2]. Confidentiality and integrity can be achieved by encryption and digital signature respectively. The most common technique to encrypt the message is pubic key encryption [4, 16, 35]. But it needs PKI to managed certificates that certify that the public key is of the claimed user with identity ID. To remove this issue, Shamir [5] introduced ID-based signature that uses user's identity ID as a public key. Instead of generating the key-pair by the user, a third party known as private key generator generates the private key on request to the user with identity ID. Since then, there are many ID-based signature schemes those have been presented in [7, 18, 21, 23, 24, 29, 41, 43]. Most of them are based on the integer factorization including the Shamir's scheme [5] and GQ scheme [29] and the rest of them are based on bilinear pairing on elliptic curves. At a recent time, Boneh and Franklin [7] suggested an ID-based encryption scheme based on the bilinear maps on an elliptic curve. This scheme was the first practical ID-based encryption, but they did not implement the ID-based signature.

PKG issues the private key to the user and keeps one copy to itself. This problem is known as the key escrow problem. To avoid this issue, several schemes [22, 31, 39, 44, 45, 46, 47] have already been disused in chapter 3 with some drawbacks and advantages over the others.

## 1.1 Problem Statements

In modern cryptographic algorithm [4, 5], the user generates his key-pair that use public key and private key. The public key is publically available to everyone in the network and is used to encrypt the message. Alternate to the public key, the private key keeps as a secret to him and is used to decrypt the message. The PKI manages each user's public key along with his identity. No one other than the user can ever decrypt the message. So, the user has guaranteed to obtain his secret information in communication. At the same time, encryption can also be used by the user to encrypt the criminal activity. So we can say that there are two main problems with the public key encryption. First one is, it requires PKI to manage the certification that certifies that the public key is of the claimed user with identity ID. And second is, a malicious user can encrypt the criminal information.

To solve the problem of certificate management in PKC, ID-based encryption scheme [5, 7, 18] tackle the first problem. Instead of using the public key, it uses the user's identity as the public key. Therefore, there is no need of certificate to certify that the public key is the real public key of the user with identity ID. There is a trusted third authority called as Private Key Generator, who generate the private key corresponds to the identity ID; one copy sends to the user and other copy stored in its storage. In future, PKG may decrypt the doubtful message encrypted by the user over the public communication. Thus, the second issue with PKC can solved by monitoring the suspicious message by the PKG with the help of the copy of user's private key stored in its storage.

Additionally, malicious PKG may also decrypt the encrypted message as PKG generates the user's private key. Thus, with advantages over the public key encryption system, ID-based encryption suffers from two concerned issues: 1) key escrow problem and 2) secure key issuing between the PKG and the user. These issues motivate the need for an efficient model to avoid such problems. To remove key escrow problem, several approaches [22, 31, 39, 44, 45, 46, 47] have proposed either by secure key issuing or by user-chosen information. Nevertheless, the advantages come with some drawbacks. Each approach comes with new disadvantages. However, every scheme solved the key escrow problem in a different manner. But it becomes a new issue in the crypto world. We realize from these schemes discussed above that each scheme give full control over their private key. Indeed, to give full control over the private key to the user is also a disadvantage. User's Privacy has induced two new disadvantages; PKG or government has no control over the user's private key, and they take no action against user's unlawful message. Today, one of the hot topics in cryptography area is to balance control on the Private Key for both the user and the PKG. Thus, the user has a right to privacy on their lawful message, and Government has rights to monitor the unlawful message of the user.

## 1.2 Recent Solutions

In the previous section, we have seen the limitations of PKE and IBE. Here, we discussed the existing solution of key escrow problem in IBE and existing solution of PKC.

### 1.2.1 Existing Solutions to key escrow Problem

Earlier several kinds of researchers on ID-based encryption scheme have proposed that avoids key escrow problem. Boneh-Franklin [7] is one of them that use the technique of threshold cryptography [50] to distribute the master key to multiple PKG instead of one, discussed in chapter 4. Due to massive infrastructure to managing multiple PKG, this scheme did not work so much efficient. At the same time, HIBE scheme [9, 12] attempts to solve the issue, but the problem remains the same of extensive infrastructure to manage the multiple PKG. In 2003, Gentry [45] introduce the certificate-based Public Key encryption, but it needs a certificate authority to certify the identity of the user and manages those certificates. To tackle this issue, certificate-less public key encryption [46] was introduced which provides implicit authentication to the public key with user-chosen information. A new variant of IBE [31] was proposed which uses a combination of the key issue by PKG and some information chosen by the user as a private key.

However, the most unattractive property of all solution is each scheme is proposed to tackle the problem of key escrow with different techniques. Each scheme is supposed to have the advantage over others. By removing the key escrow problem in existing scheme, PKG in each scheme have no control over the private key. Thus, the user may get the chance to use privately in some criminal activity since no any authority to monitor the user communication.

### 1.2.2 Existing solution to key escrow

In 1993, the U. S. government declared Escrow encryption standard [51]. This scheme based on the particular tamper-resistant hardware encryption device known as Clipper chip. This chip has two properties [64]:

1. SKIPJACK algorithm provides the secret encryption.
2. Provide "backdoor" for law enforcement to monitor the unlawful commutation.

Since then, key escrow is less attractive because the issue with this scheme is how to balance these two properties in a single approach. As we seen in ID based encryption, user's private key completely depends on the trusted third party. In 1995, Shamir [63] indicate:

> "*Nowadays even if escrowed agent is reliable, In future, other dishonest agencies may replace it, these dishonest agencies will likely decrypt escrowed key of all user suddenly and monitor user's communication for their own stake*"

Many approaches explore the problem. Shamir [40] introduced partial key escrow approaches, Micali and Ney [53] put forward shared random function and key escrow scheme, and another improved scheme [32] which is more advantageous than previous one.

## 1.3 Motivation

To remove the key escrow problem from the ID-based encryption, several schemes has been discussing in Chapter 3. HIBE [9, 12] and threshold key issuing [7] needs extra infrastructure for storage and computation time. So it consumes the lot of machine cycle and slower than other existing scheme. Certificate-based encryption [45] has the disadvantage of key revocation of certificate. Thus, it requires a large amount of storage for the certificate and computational time for verifying those certificates. Therefore, this scheme lost the advantage of identity-based encryption. Similar to certificate-based, Certificate-less encryption [46] using the user-chosen information, but only have implicit authentication with the public key. The sender will never assure that the receiver's public key is the original public key until communication is successful. Contrast to an existing scheme, VIBE [31] used the user-chosen secret information and used the combination of his confidential information and partial private generated by PKG to encrypt the message. As a result, encryption algorithm becomes more complex. Thus, there will be a need to construct an identity-based encryption that is partial key escrow problem and monitor the unlawful communication.

## 1.4 Objective of thesis

The main aim of the thesis is to construct an efficient democratic model for the identity-based cryptosystem. Here, democratic means, a model that attain a balance between the government and the peoples. Moreover, we construct a simple model for modifying the identity-based encryption to the democratic one [62], where, the government and people enjoying their rights:

1. Government rights to monitor: To provide the "back door" for authorized agency so that they can intercept the doubtful communication message and attack criminal activities
2. User's right to privacy: To provide the right to the user that they cannot compromise their privacy.

## 1.5 Structure of thesis

The structure of this thesis is organized as follows:

**Chapter 2** introduces required mathematical backgrounds such as definitions of modular arithmetic, algebraic groups, finite fields, and number-theoretic assumptions. Definition of Elliptic curve and why the elliptic curve needed in cryptography are introduced to give an enough strong background to understand our proposed model. Cryptographically Hash function and random oracle are introduced to understand the one-way hash function. Bilinear maps are presented as a response to different variants of the Diffie-Hellman assumption. The notion of bilinear maps then allows deriving the Bilinear Diffie-Hellman problem (BDH), Computational Diffie-Hellman problem (CDH) and Gap Diffie-Hellman problem (GDH) which guarantees the security of our future constructions. The chapter concludes with basic definitions and notations on cryptography.

**Chapter 3** highlights the building blocks of cryptography for the design of our solution. The chapter starts with a review on traditional public key infrastructures (PKI). In the following section, Identity-based encryption (IBE) is discussed. Then, we describe a comparison of IBE with PKI, the security aspect of IBE including IND-CPA, IND-CCA and ANO-IBE explained, related work is elaborated. Finally, Chapter 3 concludes with the comparison of the different existing scheme.

**Chapter 4** describes the design of a model of ID-based encryption scheme free from key escrow problem. We start by defining a model that describes the current solution to avoid the key escrow. With the help of this model, the current security threats are uncovered along with possible adversaries and realistic assumptions on these adversaries. Consecutively, different cryptographic design goals are defined to resolve the earlier described security threats. The design goals serve as a guideline to construct a practical algorithm based on the cryptographic building blocks from Chapter 3. The end of Chapter 4 we explore our proposed model to implement the ID based encryption, ID-based signature and authenticated ID-based key exchange protocol.

**Chapter 5** proofs the security of our model implemented on ID-based encryption, ID-based signature and authenticated key agreement protocol and analyze the performance in terms of point addition, exponentiation, and scalar multiplication compared with existing protocol. Finally, we end the chapter by claim that our scheme is fair IBE scheme.

**Chapter 6** concludes this thesis with a summary of earlier research results along with the limitations of our current solution. Finally, we close the thesis by highlighting that might be subject of future work.

Chapter

# Mathematical Backgrounds 2

This chapter briefly covers the mathematical background to understand cryptographic algorithms presented in the later section. This chapter represents the fundamental of cryptography concepts.

Note that this chapter only covers the cryptographic fundamentals required to understand the remainder of the thesis. Definitions and theorems always provided without t proof. For a more in-depth discussion on algebraic topics in this chapter, the reader is referred to [14]. More information on elliptic curves, Diffie-Hellman assumptions, and pairing-based cryptography can found in [1].

## 2.1 Mathematics of Cryptography

In this chapter, we will ready to understand the mathematics description by discussing the various mathematical tools and properties of cryptography. Some useful functions like field, ring, group, bilinear pairing, elliptic curve, etc. will be discussed here.

### 2.1.1 Modular Arithmetic

For any given positive integer n and any nonnegative integer x, if we divide p by n then we get an integer quotient q and an integer remainder r that satisfied the following equation:

$$x = qn + r;\ 0 \leq r \leq n;\ q = x/n$$

where m is the largest integer less than or equal to m. The remainder r is also known as residue or x mod n. The integer n is known as the modulus. So, for any integer x, we can write:

$$x = x/n * n + (x \bmod n)$$

Example: 13 mod 11 = 2 and -13 mod 11=9.

Two integer x and y are said to be congruent modulo n; if (x mod n) = (y mod n). It can write x=b (mod n).

Properties of modular arithmetic

1. [(x mod n ) + (y mod n)]mod n = (x + y) mod n.
2. [(x mod n ) - (y mod n)]mod n = (x - y) mod n.
3. [(x mod n ) * (y mod n)]mod n = (x * y) mod n.

**Set of Residues ($\mathbb{Z}_n$):** For the set $\mathbb{Z}_n$ as the set of nonnegative integer less than n. Suppose n is the modulo operation then the set $\mathbb{Z}_n$ = {0, 1, 2….n-1}. We can denote the residue class modulo n as [1], [2]….. [n-1], where [i]= {x:x is an integer, x = i(mod n)}

Example: $\mathbb{Z}_n$= {0, 1, 2,….(n-1)}

$\mathbb{Z}_6$= {0, 1, 2, 3, 4, 5}

$\mathbb{Z}_{13}$={0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}

**Additive Inverse**: Suppose x and y are two number in $\mathbb{Z}_n$ then it is called the additive inverse of one another if x + y = 0 (mod n). For example: in $\mathbb{Z}_{13,}$ 13-7 = 6 is additive inverse of 7, so in generalized way for $\mathbb{Z}_n$, x = n - y

**Multiplicative Inverse**:Two number x and y are multiplicative inverse toeach other if, m * n =1 (mod n) for example, in $\mathbb{Z}_{13,}$ the multiplicative inverse of6 is 11 because 6*11= 1 (mod 13). The integer x in $\mathbb{Z}_n$ has a multiplicative inverse exist only if gcd (x, n) = 1. For example, 4 have no multiplicative inverse in $Z_{14}$because gcd (14, 4) ≠ 1.

Some more sets:

- **$\mathbb{Z}^*_n$**: It is the subset of $\mathbb{Z}_n$ and contains only those integers for which multiplicative inverse exist. In $\mathbb{Z}_n$each member contains additive inverse but only a few members contain multiplicative inverse. Example:

$\mathbb{Z}_6$ = {0; 1; 2; 3; 4; 5} $\mathbb{Z}^*6$= {1; 5}

$\mathbb{Z}_{15}$= {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}  $\mathbb{Z}^*_{15}$= {1, 2, 4, 7, 8, 11, 13, 14}

- **$\mathbb{Z}_p$**: It is similar to $\mathbb{Z}_n$ where n is prime number p. $\mathbb{Z}_p$ contains all integers between 0 to p-1. Each element that belongs to $\mathbb{Z}_p$ has an additive inverse, and all elements have multiplicative inverse excluding 0. Example:

  $\mathbb{Z}_{13}$= {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}

- **$\mathbb{Z}^*_P$**: It is similar to $\mathbb{Z}^*_{n,}$ where n is prime number p and the subset of $\mathbb{Z}_p$. In $\mathbb{Z}_{p,}$ only some elements have multiplicative inverse but in $\mathbb{Z}^*_P$ all member have multiplicative inverse excluding0. $\mathbb{Z}^*_P$ contain all integer from 1 to p-1. Example:

  $\mathbb{Z}^*_{13}$= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}

### 2.1.1 Mathematics of Symmetric Key Cryptography

The requirements of cryptography are the sets of integers and different operations performed on those sets. The different operations applied to the elements of the combination of the different set are called an algebraic structure. In this section, we will briefly discuss three common algebraic structures: groups, rings, and fields.

- **Group**: A group is the set of an element that contains the operation of binary "●" and satisfies four operations. The properties are as follows:
  1. Closure: If x and y are the elements of $\mathbb{G}$ then z = x ● y is also the element of $\mathbb{G}$ that means if we apply any operation to any element of group $\mathbb{G}$ then result willalso belong to group $\mathbb{G}$.
  2. Associativity: If x, y, and z are the element of group $\mathbb{G}$, then

     $$(x \bullet y) \bullet z = x \bullet (y \bullet z).$$

  3. Existence of identity element: For all x in group $\mathbb{G}$ there exist an identity element 'e' such that

     $$e \bullet x = x \bullet e = x.$$

4. Existence of inverse: For each x in group$\mathbb{G}$ there exist an element ‘e’ such that

$$e \bullet x = x \bullet e = e.$$

The commutative group is a group that satisfies above four operations and commutative operation, also called **abelian group**.

5. Commutativity: If x, y belongs to group$\mathbb{G}$, then $x \bullet y = y \bullet x$.

- **Finite Group**: A group is called finite if it contains the finite number of elements otherwise it is called infinite group.
- **Order of Group**: The number of a unique element present in the group is known as an order of the group. If the number of an element is finite, then it is called finite order otherwise infinite order.
- **Subgroup**: A subset $\mathbb{H}$is called the subgroup of group $\mathbb{G}$ if $\mathbb{H}$ itself is a groupwith respect to the operation on $\mathbb{G}$, in other words if $\mathbb{G} = <x, \bullet>$ is a group and $\mathbb{H} = <y, \bullet>$ is a group under the same operation and y is non-empty subset of x, then$\mathbb{H}$ is called subgroup of $\mathbb{G}$.This definition yields:
    1. If x and yare the members of both group then $z = x \bullet y$ is also the member of both groups.
    2. Same identity element exists for both.
    3. If x belong to both groups, then the inverse of x also belongs to both groups.
    4. Each group is itself a subgroup.
- **Cyclic Subgroup**: A subgroup is said to be cyclic if it can be generated by the power of an element of the group.

$$x^n \rightarrow (x \bullet x \bullet x \bullet x \bullet x \bullet x \bullet \ldots.. \bullet x)(\text{n times})$$

- **Cyclic Group**: It is the group that contains its own Cyclic Subgroup. The Component that can generate cyclic subgroup can also produce the whole group itself, that component is called generator of the group. If g is a generator then, the elements in finite cyclic group can be written as

$$\{e, g^1, g^2, \ldots g^{n-1}\}, \text{ where } g^n = e.$$

Note: A cyclic group can have many generators.

Example: The group $\mathbb{G}$ = <$Z^*$ ,∗, ×> is a cyclic group with two generators, $g = 3$ and $g = 7$.

- **Ring**: A ring $\mathbb{R}$= <{…}, **●, ■**> is a type of algebraic structure that have two operations. First operation (●) satisfies the all five properties of abelian group that is
    1. Closure
    2. Associativity
    3. Commutativity
    4. Existence of inverse
    5. Existence of identity

  Second operation (**■)** is distributed over first operation and satisfies only the two properties:
    1. Closure
    2. Associativity

    The ring which second operation satisfies commutative property is called commutative ring.

- **Field**: A field $\mathbb{F}$=< {…}, **●, ■** > istype of commutative rings in which the second operation satisfies all five properties that defined for the first operation except that the identity of the first operation has no inverse.
- **Finite Fields**: A field with the finite number of an element is called the finite field. Galois demonstrated that the field has finite number of component must be $p^k$, where p is a prime number and k area positive integer. A Galois field, GF($p^n$), is a finite field with $p^n$ elements. While n=1, we have GF($p$) field. This field can be the set $\mathbb{Z}_p$, {0, 1, …, p − 1}, with two arithmetic operations.

## 2.2 Elliptic-Curve Cryptosystem

The properties and functions of elliptic curves have been studied in mathematics for 150 years. In 1985 Victor Miller [56] and Neal Koblitz [55] suggested an elliptic curve as a mathematical tool in cryptography known as the elliptic –curve cryptosystem.

### 2.2.1 Definition of Elliptic Curve

An Elliptic-curve over a finite field is a smooth non-singular cubic projective curve of genus 1 defined over k with distinguished k rational points. By, non-singular means all 3 roots of EC must be distinct. Over any field F, an irreducible projective curve is a compact manifold that is topological as a sphere with handles. The number of handles is the genus.

### 2.2.2 General form of Elliptic Curve

Any elliptic curve can be defined by following equation:

$$y^2= x^3+ ax + b,$$

Where x is not a continuous point that is chosen from particular field GF(P) or GF ($2^k$). The figure 3.1 shows the elliptic curve [8] of equation $y^2= x^3- x + 1$.

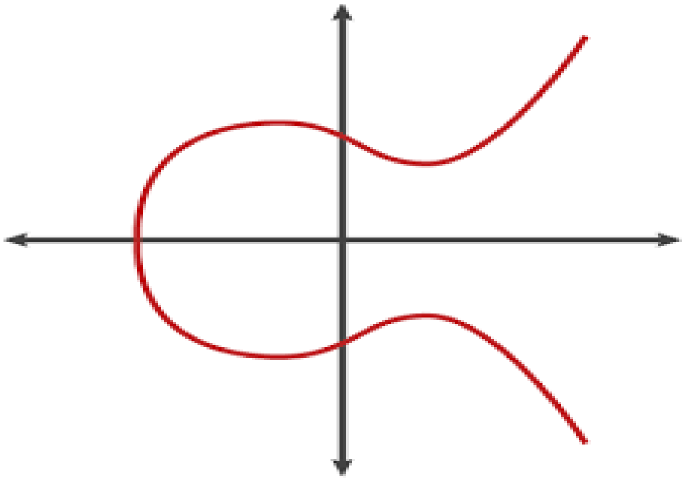

Figure 2. 1 Graphical representation of elliptic curve $y^2= x^3- x + 1$

Let E be an elliptic curve over F defined by Weierstrass equation as follow:

$$y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6$$

If P is a rational point on elliptic curve E and l is a line through P with rational slope, it is not necessarily true that l intersects E in another rational point. However, if P and Q are two rational points on elliptic curve E, then the line PQ intersects E in a third rational point R. This permits us to generate many new rational points from old ones.

And also, it permits us to define a group operation on E(k) for any elliptic curve defined over any field k.

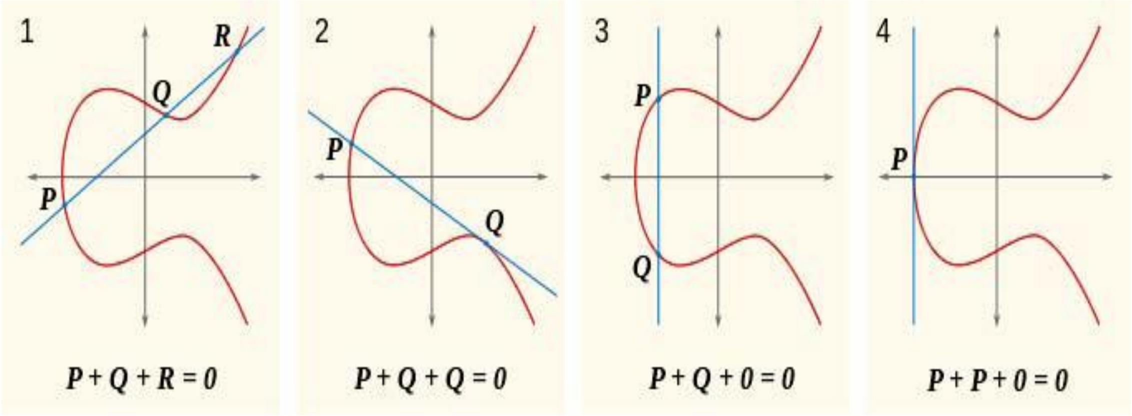

Figure 2. 2 Rational points on line over elliptic curve.

Figure 3.2 denote the sum of three rational points on line l over elliptic curve E should be zero [17]. Here, zero is a point at infinity. The evaluation of P+Q=R is purely algebraic. The coordinates of R are rational functions of the coordinates of P and Q, and can be computed over any field. By adding a point to itself repeatedly, we can compute 2P=P+P, 3P=P+P + P, and in general, nP=P+⋯+P for any positive n. We also define 0P=0 and (−n)P=−(nP). Thus, we can perform scalar multiplication by any integer n.

There are three particular applications that we will explore in detail:

1. Factoring integers
2. Primarily Proving
3. Cryptography

### 2.2.3 Why Elliptic Curve Cryptography?

Elliptic curve cryptography is one of the robust and faster Public Key encryption technique based on EC theory. The main advantage of using Elliptic curve cryptography is to make smaller, faster, and efficient keys of cryptography. With the help of Elliptic-Curve, it generates a pair of keys. As compared to RSA requires 1024 bit key, it gives security level with only 164-bit key. Indeed, ECC helps to make equivalent security with less computation power and battery resource usage. So, it is most applicable for mobile applications. In every ten years, key size becomes double so traditional methods can'tbe used due to large bit key.

| Organization | RSA key length(in bits) |
|---|---|
| ICICI Bank | 1024 |
| Amazon | 2048 |
| eBay | 2048 |
| Online SBI | 2048 |
| Facebook | 1024 |
| Canara Bank | 2048 |

Table 2. 1 RSA Key length of Some Organization

Table 2.1 shows some currently used RSA key length by some organization. If key size increase then definitely it increases the security, but it causes the serious problem. If we double the RSA key length, then decryption will be 8 times slower. Table 2.1 gives the RSA key length of some organization and Table 2.2 gives the security level of ECC and RSA. Ciphertext size also becomes large. The speed of encryption also infected with large key length, which is slower by the factor of 4. Table 3.2 gives the security level of ECC and RSA scheme. From the Table 3.1 and 3.2, it is clear that ECC takes less key length so as compared to RSAit is more efficient.

| Key Size (in bits) | 80 | 112 | 120 | 128 | 256 |
|---|---|---|---|---|---|
| **ECC** | 160 | 185 | 237 | 256 | 512 |
| **RSA** | 1024 | 2048 | 2560 | 3072 | 15360 |

Table 2. 2 RSA and ECC key Sizes [65]

Application of ECC: ECC takes low power and low key length, so any application that takes less power and more security where ECC is used. Some areas are as follows:

1. Wireless communication devices
2. Online transactions
3. Mobile devices

4. Smart cards
5. Web servers

## 2.3 Cryptographic Hash Function

Cryptographic hash function [57] is a function that takes variable length string as an input and gives Fixed-length string i.e. message digest. $H:\{0, 1\}^{*} \rightarrow \{0, 1\}^{k}$ where k is the length of a message digest. Let's take a function f(x) = y that maps x to the image y. x is called pre-image of y. The output is called hash value or message digests. Here we use y = H(x) that denotes, applying hash function into variable length message x and that gives fixed length digest y. Hash function should follow some characteristics:

1. x should be variable length and y is fixed length.
2. For given x, it's easy to compute y but vice versa should be very tough that means hash function should be the one-way function.
3. Two messages doesn't have same message digest.
4. The hash function must be easy to compute.

Suppose x and y are messages then H(x) = H(y) is infeasible. Today, Hash function isused in various cryptographic techniques like message authentication code (MAC), digital signature, Random sequence generator used in key agreements, authentication protocol, etc. Hash function needs to satisfy the four main properties:

1. Pre-image Resistance: Given a digest y = H(x), it is computationally infeasible to compute x. That is, the computational cost of getting the input x must be $\geq 2_{k}$, where H(x) = y and |y| = k.
2. The hash function for which pre-image can't be solved efficiently is called pre-image resistance.
3. Second pre-image resistance: Given message x it is computationally infeasible to compute different message $x_0$ that have same message digest, i.e. $H(x) = H(x_0)$ is infeasible to compute. It is called second pre-image resistance.
4. Collision resistance: It is impossible to find two messages with the same message digest. That means if x and $x_0$two different message then $H(x) = H(x_0)$ is impossible. This property is known as collision resistance.

Hash functions are useful in wide range of practical applications. For example, hash functions act as one-way functions in password databases to lighten sensitivity of the stored content. Besides, hash functions are also a valuable tool for data authentication and integrity checking.

### 2.3.1 Random Oracle model

Random Oracle model was introduced in 1993 by bellare and rogaway [58]. A Random Oracle is a theoretical black box that gives a uniformly Random chosen result from its output domain for each unique query. A Random Oracle is deterministic, i.e. given a particular input it will always produce the same output.

The behavior of this model is given as:

1. When any new message comes then, Oracle creates the fixed size of digest for that message and save the message and digest in Oracle record.
2. When any message exists and digest exists for that message, then Oracle simply puts the message digest on their record.
3. The digest for any new information is independently chosen from the previous digest.

In an ideal model hash functions can be considered Random Oracles. That is, the output of the hash function would look like perfect Random bit sequences if and only if Hash function is ideal. Therefore, hash functions are often examined Random Oracles in security proofs. Such security proofs are called proven secure in the Random Oracle model. The next step of these security proofs is replacing the Random Oracle accesses by the computation of an appropriately chosen (hash) function [58]. Algorithms that are not requiring such a system in their security proof are said tobe proven secure in the standard model.

### 2.3.2 Pigeonhole principle

The pigeonhole principle can understand Random Oracle model. It states that if we have n pigeonholes and n+1 pigeons then 2 pigeons are occupied in at least one pigeonhole. In the generalized way, if tm+1 pigeons occupy m pigeonholes, then at least one pigeonhole is occupied by t+1 pigeons. Because the main idea of hashing yields the digest should be shorter than the message, according to the principle there can be a collision. In other words, there must be some digest that corresponds to more than one message, so the relationship between messages and possible digests is many to one.

## 2.4 Pairing-Based Cryptography

The main idea of pairing-based cryptography [1, 3] is to map between two important groups. It allows a new scheme based on the reduction of one problem to another that means reduction of problem that is hard from one group to the problem that is easier as compared to first one in another group.

### 2.4.1 Bilinear Maps

Bilinear Map [6] allows mapping between different groups. Let $\mathbb{G}_1$ is the cyclic additive group with generator p. The bilinear map is also called pairing because it allows a pair of the element from $\mathbb{G}_1$ and $\mathbb{G}_2$ to another group $\mathbb{G}_3$.Suppose $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_3$ are cyclic groups with large prime order q. Generally $\mathbb{G}_1$, $\mathbb{G}_2$ are the additive group and $G_3$are the multiplicative group. A bilinear pairing isdescribed as e: $\mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_3$ that satisfy the bilinear property:

$$e(x.P, y.Q) = e(P, Q)^{x.y} \text{ for all } P \in G_1,\ Q \in G_2 \text{ and all } x, y \in \mathbb{Z}.$$

It means if P is the generator of $\mathbb{G}_1$ and Q is the generator of $\mathbb{G}_2$ then e(P,Q) is the generator of $\mathbb{G}_3$. The mapping is called computable if there exist some algorithm that can efficiently compute e(P, Q) for $P,Q \in \mathbb{G}_1$. If $\mathbb{G}_1 = \mathbb{G}_2$ then pairing is called symmetric otherwise pairing is known as asymmetric. If $\mathbb{G}_1 = \mathbb{G}_2 = \mathbb{G}_3$ then pairing is called self-bilinear map.

### 2.4.2 Bilinear pairing

Suppose$\mathbb{G}_1$ is a cyclic additive group, $\mathbb{G}_2$ is a cyclic multiplicative group of the same order q, and P is generator of $\mathbb{Z}_q$. A bilinear pairing is a map e: $\mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_3$ that satisfies the following properties:

1. *Bilinearity*: For every P, Q, R $\in \mathbb{G}_1$,

$e(P, Q+R) = e(P, Q)e(P, R)$ or

$e(P+Q, R) = e(P, R)e(Q, R)$

for any $x, y \in Z_q$

$$e(x.P, y.Q) = e(P, Q)^{xy} = e(x.y.P, Q)$$
$$= e(P, xyQ) = e(x.P, Q)^{y} = e(P, x.Q)^{y}$$
$$= e(y.P, Q)^{x} = e(P, y.Q)^{x}$$

2. *Non-Degeneracy*: If everything maps to identity then it is undesirable, if P is generator of $\mathbb{G}_1$ then e(P, P) is generator of $\mathbb{G}_2$ that means if there exist $P \in \mathbb{G}_1$ such that $e(P, P) \neq 1$, where 1 is identity element of $\mathbb{G}_2$.

3. *Computability:* There must exist an algorithm that can efficiently compute e(P, Q) for every $P,Q \in \mathbb{G}_1$.

Here, $\mathbb{G}_1$ and $\mathbb{G}_2$ as an additive notation and $\mathbb{G}_3$ with a multiplicative notation. In general, $G_1$ and $G_2$ are the groups of points on an elliptic curve and $G_3$ will denote the multiplicative subgroup of the finite field. The map e will be Tate pairing or Weil pairing on an elliptic curve over a finite field.

Now, we ready to describe some mathematical problems.

1. **Discrete Logarithm (DLP) problem**: Given two Random integer $P \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$, compute an integer x, such that $Q = x.P$, where $x \in \mathbb{Z}_q$.

   *DLP Assumption*: DLP is hard to solve.

2. **Bilinear Diffie-Hellman (BDH) problem**: Given $x, y, z \in \mathbb{Z}_q$, and $<P, x.P, y.P, z.P>$ compute $e(P,P)^{x.y.z} \in \mathbb{G}_{3.}$

   *BDH assumption*: BDH is hard to solve.

3. **Computational Diffie-Hellman (CDH) problem:** Given $x, y \in \mathbb{Z}_q$ and $<P, x.P, y.P>$, compute xyP.

   *CDH Assumption*: CDH is hard to solve.

4. **Decision Diffie-Hellman (DDH) problem [11]:** Given x, y, z $\in \mathbb{Z}_q$, and <P, x.P, y.P, z.P> check whether z = x.y mod q.

   *DDH Assumption*: DDH is hard to solve.

5. **Gap Diffie-Hellman (GDH) problem**: A group of problem where DDHP is easy while CDHP is hard.

Our model described in next chapter considers the GDHP group where DDHP is easy, but CDHP is difficult to compute. One of the good examples of GDHP group is the bilinear pairing described above.

## 2.5 Cryptographic Definitions

This section defines basic cryptographic aspects, and their notation applied throughout the remaining part of the thesis.

### 2.5.1 Terminology

1. Confidentiality: The assurance of an entity's information is protected from open to unauthorized users.
2. Integrity: Unauthorized users did not modify the assurance to entity information.
3. Authentication: The assurance to an entity that another entity effectively has a claimed identity.
4. Authenticity: The assurance to entity information comes from the claimed entity.
5. Non-repudiation: The assurance to an entity of authenticity and integrity of information that undeniably links the originating entity as the source of information.

### 2.5.2 Symmetric cryptosystem

Symmetric cryptosystem algorithm requires a secret shared key between two entities to achieve confidentiality. For encryption, this cryptosystem has two polynomial time algorithm

1. An Encryption algorithm C← (m, k) that encodes the plaintext m to a Ciphertext under symmetric key k such that only parties having key K can derive m from C.

2. A Decryption algorithm m ← (C, k) that decodes a Ciphertext C from plaintext m under symmetric key k

### 2.5.3 Asymmetric cryptosystem

In the asymmetric cryptosystem, each entity has a key pair ( $usk_{Pr}$, $usk_{Pub}$). One key is Private Key denoted as $usk_{Pr}$ is known to the user only, Public Key $usk_{Pub}$ made available to every entity. The Private Key is mathematically linked with the corresponding Public Key. Because the Public Key is derived from the Private Key using the one-way function e.g. in the ElGamal encryption scheme [59] the Public Key is calculated as $usk_{Pr} = g^{usk_{Pub}}$ in a group $\mathbb{Z}_p$ for some large prime p. Similar to the symmetric cryptosystem, this system also consist of two polynomial time algorithm:

1. An Encryption algorithm C← (m, $usk_{Pub}$) encodes the plaintext m with the Public Key $usk_{Pub}$ yields to a Ciphertext C such that only parties having the corresponding Private Key $usk_{Pr}$ can derive m from C.
2. A Decryption algorithm m ← (C, $usk_{Pr}$) that decodes a Ciphertext C from plaintext m with his Private Key $usk_{Pr}$.

## 2.6 Summary

This chapter summarized the basic mathematics concept required to understand the model described in Chapter 3. The first part of this chapter introduced basic mathematics of cryptography includes the modular arithmetic, Group, ring, Field, and algebraic structures. Elliptic curve cryptography includes elliptic curve definition, the graphical view, and application, are introduced in next section. Also discussed Hash function and Random Oracle model, and conclude with differences between security under Random Oracle assumptions and security in the standard model. From the discrete logarithm assumption, several variants of the Diffie-Hellman problem were introduced, consequently leading to the Gap Diffie-Hellman assumption. The Bilinear Diffie-Hellman assumption defined as a computationally infeasible problem for the construction of cryptographic protocols relying on bilinear maps. Finally, this chapter concluded with basic cryptographic terminology.

Chapter

# Cryptographic Building Blocks 3

This chapter overviews the fundamental building blocks of cryptography used to design the fair model of ID-based encryption free from key escrow problem. An introduction of Public Key Infrastructures is given and discusses their limitations. Then, explore the Identity-Based Encryption (IBE), which is an alternative to existing Public Key Infrastructures, along with its disadvantages and advantages. In the next section, some solutions describe regarding the key escrow problem. Different security definitions are also overview. Finally, Distributed Key Generation (DKG) is described as a possible solution to the inherent key escrow problem of IBE.

## 3.1 Public key encryption

A class of asymmetric cryptosystem which based on algorithm takes two different keys; one for encryption and another for decryption. The user generates the Key pair ($usk_{Pr}$, $usk_{Pub}$). Private Key is kept secret to himself on the other hand; the Public Key is publicly available to everyone. The pair of private and Public Key allows secure communication between two parties who never met before. To verify the linking of a Public Key to the corresponding user, there must be a trusted authority that prevents impersonation attack. All Public Keys are authenticated by the infrastructure known as Public Key infrastructure. Whenever needed, the sender requests the CA to issue the received Public Key. On receiving the request, the CA provides the receiver Public Key if and only if the sender identity is valid. Now, sender encrypt/sign the message with Public Key yields the cipher text and sends to the receiver over the unsecure medium. Consequently, a receiver decrypts/verifies the encrypted/ signed message with his Private

Key. One of the good things with this cryptosystem is that Private Key needs not to be transmitted or revealed to anyone. Thus, the security increased. It gives the facility of digital signature so that the user can never deny its existence. In security point of view, PKI believed in trusting the user's key rather than them. Suppose Eve has made the PKI who ensure, that his Public Key is identified as the Public Key of Alice. So, Eve can be able to modify all Alice's communication as he really has the Alice's Public Key corresponding to the Alice's Private Key. Thus, is required to have the Public Key system depend on the infrastructure that authenticates whether key pairs belong to claimed user. In the real world, this can be achieved with the help of the certificate authority.

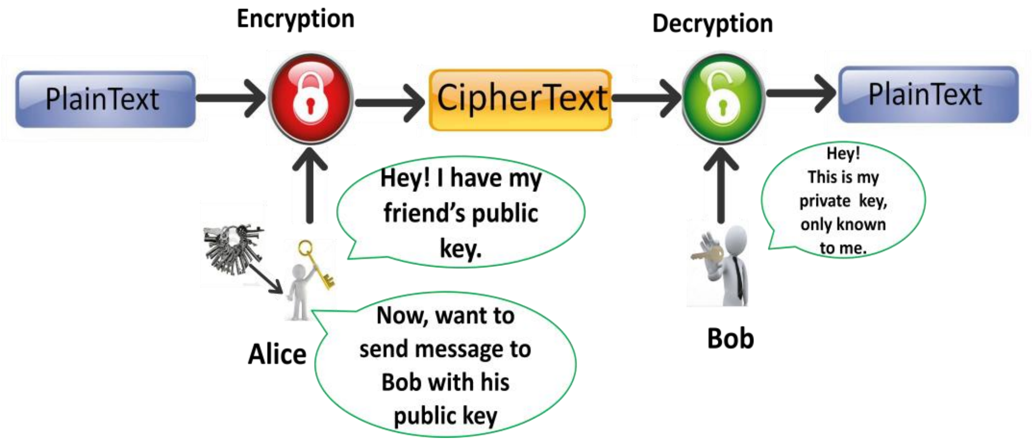
Encryption
Decryption
PlainText
CipherText
PlainText
Hey! I have my friend's public key.
Hey! This is my private key, only known to me.
Alice
Now, want to send message to Bob with his public key
Bob

Figure 3. 1 Public Key Cryptography

### 3.1.1 Certification Authority

In PKI system, users trust the trusted party called certification authority. CA guarantees Public Keys belong to the alleged user. X.509 determines the CA infrastructure [60]. Suppose Alice wish to authenticate herself with the CA using their key pair. On authenticated with CA, Alice sends her public to CA with the proof conveying that Alice has the corresponding Private Key. This proof of correct possession can achieve in the form of signature derived from the Private Key with the Public Key.

On getting proof of correct possession of Alice's Public Key, CA distributes a certificate certifying that $usk_{Pub}$ is really the Public Key of Alice. CA can sign the Alice's certificate with its Private Key so that illegal certificate may avoid. So, the third person who may have doubted the authenticity of Public Key, check the signature of the

CA with his Public Key on the certificate. In the real world, there are large amounts of user using the PKI system. So, there is a complex hierarchy of CA's who get permission to issuing the certificates on their Private Key. Compromised CA signed keys can break the whole system. So, this needs heavy requirement of CA infrastructure. Certificate of authenticated Public Key belongs to the malicious user signed with a compromised signed key.

A user's Private Key can be revoked by the CA if it lost or reveal to the third person. CA can solve this issue by publication of the revocation list (contain all compromised Public Keys) periodically. To deal successfully with the problem of Revocation List, the certificate includes an expiration date. A certificate should no longer trust after the certificate is expired.

## 3.2 Identity-based Cryptosystem

The concept of identity-based encryption first introduce by Adi Shamir [5], co-inventor of RSA system, in 1984. For encryption and signature verification, it uses user identity as a Public Key instead of the digital signature. User identity can be anything by which he/she can uniquely identify, such as email-id, phone number, SSN, etc. Shamir' innovation was to eliminate the need for generating and managing the users' certificate. This feature reduces the complexity of the cryptosystem. This makes it more efficient to provide cryptography for novice users. Shamir's scheme [5] based on the integer factorization of RSA. This scheme is built only for signature and verification. It becomes an open challenge for all researchers. Since then; many ID-based encryption schemes [22, 31, 39, 44, 45, 46, 47] were introduced. In 2001, Boneh and Franklin [7] was the first to propose the identity-based encryption scheme based on bilinear pairing. Moreover, after, Lynn [42] and cocks [34] were also two of several Identity-based encryption schemes.

### 3.2.1 Definition

This scheme is consists of four algorithms (setup, Extract, Encrypt, and Decrypt) as shown in figure 1.2 and runs as follows:

*IBE.Setup($1^k$)*: On input of a security parameter *k*, outputs a master secret $msk_{Pr}$ and public parameters *params*.

*IBE.Extract(params,* $msk_{Pub}$*, ID)***:** Takes public parameters *params*, the master secret $msk_{Pr}$, and an id as input and returns the Private Key $usk_{Pr}$ corresponding to ID.

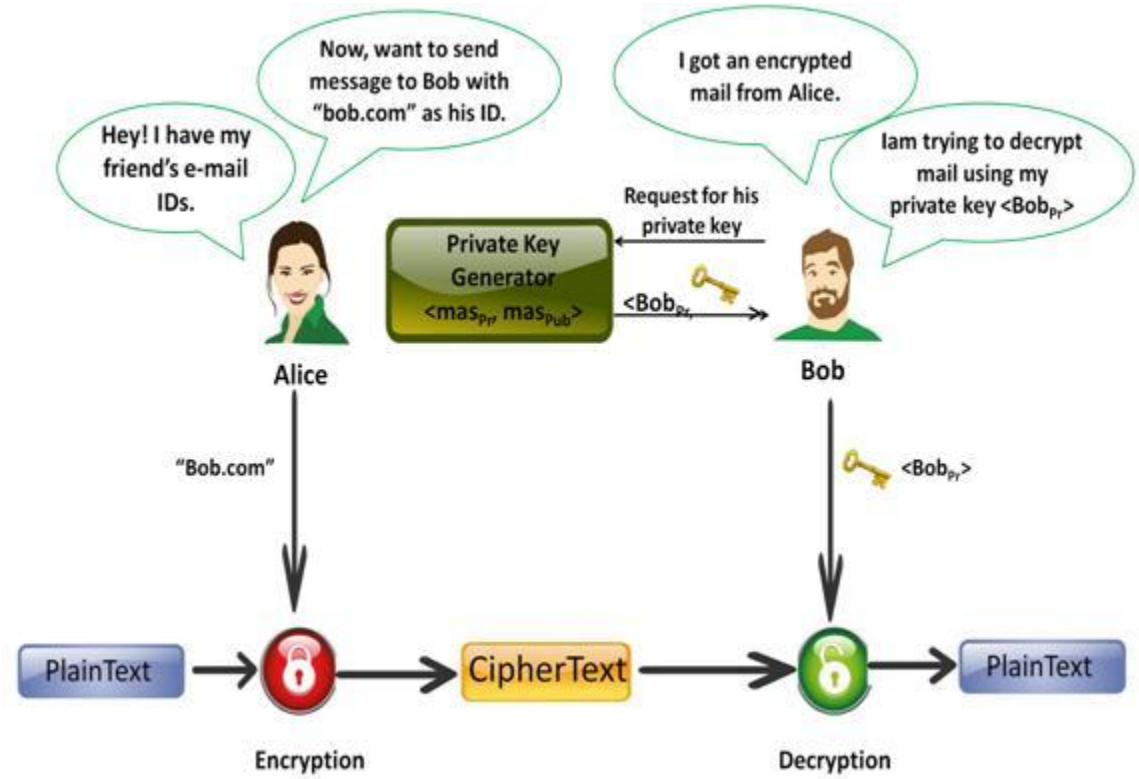
Hey! I have my friend’s e-mail IDs.
Now, want to send message to Bob with “bob.com” as his ID.
I got an encrypted mail from Alice.
Iam trying to decrypt mail using my private key <Bob$_{Pr}$>
Request for his private key
Private Key Generator <mas$_{Pr}$, mas$_{Pub}$>
<Bob$_{Pr}$
Alice
Bob
“Bob.com”
<Bob$_{Pr}$>
PlainText
CipherText
PlainText
Encryption
Decryption

Figure 3. 2 Identity-based encryption

*IBE.Encrypt(params, ID,m)***:** Returns the encryption c of the plaintext message *m* on the input of the public parameters *params*, the id, and the arbitrary length message *m*.

*IBE.Decrypt(*$D_{ID}$*, c)***:** Decrypts the Ciphertext*c* = IBE.Encrypt(*params*, ID,*m*) back to the plaintext message *m* on input of the Private Key$D_{ID}$ corresponding to the receiving identity ID.

### 3.2.2 Comparison with PKI scheme

Now, we are ready to overview some disadvantage and advantage of IBE with traditional PKI system. The difference between the two systems (PKC and IBE) is in the mathematical coordination and verifying between the key pair [15].

**Disadvantage**

When compared to traditional PKI system, the disadvantages of IBE are given as follows:

- *Single point failure*: Every user's Private Key is generated by the PKG in the system consequently suffering the single point of failure. A new user can no more get their Private Key if PKG disconnect the communication due to a plenty of extraction requests.
- *Key escrow*: The PKG extracted user's Private Key and stored it. A malicious PKG can use this information to start tapping on the insecure channel between two users. The inherent property in ID based encryption that Private Keys have to share with PKG is called key escrow in [20]. On the contrary, traditional PKI only authenticates the pair of private and a public key; it does not have key escrow.
- *Public Key revocation*: The generic IBE scheme does not support revocation of Public Keys. Although, if recipient hasty towards the privacy of his Private Key, his Private Key can get compromised. Indeed, several researchers have worked on the same issue [10, 25]. It requires an extra infrastructure that makes the generic IBE system more complex. The main disadvantage of revoking of receiver key is that he can no longer receive an encrypted message. Thus, the practical solution of this problem in [7] is to append the expiration date along with the Public Key so that Public Key will have no use after the expiration date. Similarly, traditional PKI publishes revocation lists for a solution of the same issue, but this list make PKI more complex.

**Advantage**

When compared to traditional PKI system, the benefits of IBE are given as follows:

- System complexity: PKI systems are complex infrastructure due to the support of revocation list and hierarchical organization of CA. On the other hand, IBE scheme has only one PKG serves to understand fully the IBE scheme that lightens complex infrastructure requirement.
- User Amiable: Users with no knowledge of cryptographic primitives no longer have to make aware of the decision on the key length of their key. The Public Key in an IBE scheme formulated in such way that it transparent to users having no pre-knowledge of cryptography. Thus, on an average, for any user it easy to

memorize the username or e-mail address rather than authenticating the Public Key.

- Management of certificate: In traditional PKI, there are large some users who get certificates from the trusted authority. As shown in the previous section, we have seen that it is very difficult for management and distributed the user's certificate. While this could avoide in IBE scheme with the help of PKG,which generates the user's Private Key, using their unique identification entity, on user request.

### 3.2.3 Application of Identity-based encryption

Apart from Data encryption, digital signature, and key management, there are several other practical applications. Boneh and Franklin [7] show some applications:

- *Revocation of the Public Key*: In an IBE scheme, it is easy to make key expiration by encrypting the message using the Public Key "receiver-ID||current-time". Unlike PKC, instead of obtaining new certificates eve time, the receiver will query the PKG to obtain the new Private Key. Thus, IBE scheme is very efficient and powerful way of implementing ephemeral Public Keys. This proposal can also helpful for sending messages in the future since the receiver will unable to decrypt the message until he obtains the Private Key for the date specified by the sender from the PKG.
- *Managing user credential*: encrypt the message using the Public Key "receiver-ID||current-time||Clearance-level", such that the receiver will able to decrypt the message only if he has given Clearance. Consequently, PKG grants and revoke the user credential.
- *Delegation of the Private Key*: Suppose in a company, a manager act as a Private Key Generator and has several assistants each responsible for different jobs. The manager gives his assistants the Private Key corresponding to their responsibilities. So, according to his responsibility each assistant can decrypt the message, but cannot decrypt the message to another assistant. Because the manager has his master key so he can decrypt all messages.

- *Forward secure encryption*: In forward-secure encryption scheme [49], each time the receiver’s Private Key regularly evolves so that the Private Key of a particular period is compromised, every message encrypted in the past will be secure.
- *E-voting* [27]*:* E-voting refers to online voting such as remote Internet voting as well as physical voting. Physical voting uses an electronic machine (direct recording electronic machine) in the Polling-Booth. E-voting provides several many features such as secrecy, fairness, universal verifiability, voter verifiability, so e-voting canensure efficiency and accuracy of voting and provides transparency in the voting system.
- *Authenticity in key agreement:* Diffie-Hellman [4] was the first who established the first feasible approach for constructing a shared secret over an insecure communications network, but no user authentication is there. To provide user authentication, several approaches [26, 28, 52, 48] are introduced. Nan Li proposed in the scheme [13] provide the user authentication with the help of authentication server and the hash algorithm. M. Kumar, et. al [38] introduce the ID- based authenticated key exchange protocol along with the remove the attack subjected to D-H scheme.

### 3.2.4 Security of IBE

Similar to the Public Key system, IBE also follows same security aspects. Therefore, definitions of security are often discussed. In literature, the most favorable security aspects are indistinguishability under chosen plaintext attack (IND-CPA) and indistinguishability under chosen Ciphertext attack (IND-CCA). The anonymity of the encryption scheme is an additional property of the scheme.

*Indistinguishability under Chosen Plaintext Attack*

Indistinguishability under chosen plaintext attack (IND-CPA) is understood by the negligible advantage an adversary has in trying to distinguish which of both given plaintext messages $m_0$ and $m_1$ generated a Ciphertext C. IND-CPA defined with the help of following game described in Game 1 between a challenger and an adversary. The advantage of the adversary in winning the IND-CPA game defined as

$$Adv = |Pr [b = b'] - 1/ 2 |$$

If the adversary has negligible advantage trying to win the IND-CPA game, the IBE system is said to be IND-CPA secure. An adversary is supposed to have an "negligible advantage" if it wins the above game with probability $\frac{1}{2} + \epsilon$. More formally, an IBE system is IBE-IND-CPA secure if for every adversary with advantage Adv in winning the IBE-IND-CPA game illustrated in Game 1 there exists a negligible function $\epsilon$ such that Adv $\leq \epsilon$.

**Game 1:** Generic IBE-IND-CPA [2]

**Aim**: An adversary is challenged to check the IND-CPA security of an IBE scheme by a game.

**Output**: This IBE-IND-CPA Game helps to define the concept of IND-CPA security for IBE schemes.

1. The challenger runs <$msk_{Pr}$, *params*> ← IBE.Setup($1^k$) and returns *params* to the adversary.
2. The adversary can start querying an Oracle$O_{Extract}$($ID_i$) that returns a Private Key $D_{ID}$ ← IBE.Extract(*params*, $msk_{Pr}$ , ID) corresponding to an adversary defined identity $ID_i$.
3. The adversary picks two equal length plaintext messages $m_0$ and $m_1$ and an identity $ID_{encrypt}$. The adversary honestly passes <$m_0$, $m_1$, $ID_{encrypt}$> to the challenger.
4. The challenger picks a Random bit b and executes

   C ← IBE.Encrypt(*params*, $ID_{encrypt}$, $m_b$)

The challenger gives C to the adversary.

5. The adversary continues querying the Oracle$O_{Extract}$ ($ID_i$) adaptively.
6. The adversary outputs a bit b' based on the Ciphertext C. If b = b' the adversary wins the game. If b ≠ b' or if the adversary queried the Oracle $O_{Extract}$($ID_i$) with $ID_i = ID_{encrypt}$ during step 2 or step 5, the adversary loses the game.

*Indistinguishability Under Chosen Ciphertext Attack*

Indistinguishability under chosen Ciphertext (IND-CCA) is a more demanding level of security. Therefore, an algorithm that is IND-CCA secure is considered more secure than an IND-CPA secure algorithm. IND-CCA security implies that an adversary has no advantage in trying to distinguish which of both given plaintext messages m0 and m1 generated a Ciphertext C even if the adversary has access to a list of (plaintext, Ciphertext)-tuples.

IND-CCA defined with the help of a game that challenges an adversary similar to the IND-CPA game. Compared to the IND-CPA game, the IND-CCA game contains two additional steps in which the adversary gets access to another Oracle. The advantage of the adversary in winning the IND-CCA game illustrated in Game 2, is defined as

$$Adv = |Pr[b = b'] – 1/2 |$$

If the adversary has negligible advantage trying to win the IND-CCA game, the IBE system is said to be IND-CCA secure. More formally, an IBE system is IBE-IND-CCA secure if for every adversary with advantage Adv in winning the IBE-IND-CCA game illustrated in Game 2 there exists a negligible function $\epsilon$such that Adv $\leq \epsilon$. In literature, a distinction is often made between a non-adaptive case (IND-CCA1) and adaptive case (IND-CCA2) of IND-CCA. In the non-adaptive case, step 6 and 7 from Game 2 is not allowed. More precisely, an IBE scheme that satisfies Game 2 is said to be IND-CCA2 secure.

**Game 2:** Generic IBE-IND-CCA [2]

**Goal**: An adversary is challenged to check the IND-CCA security of an IBE scheme by a game.

**Result**: This IBE-IND-CCA Game helps to define the concept of IND-CPA security for IBE schemes.

1. The challenger runs <$msk_{Pr}$, *params*> ← IBE.Setup($1^k$) and returns *params* to the adversary.
2. The adversary can start querying an Oracle $O_{Extract}(ID_i)$ that returns a Private Key $D_{ID}$ ← IBE.Extract(*params*, $msk_{Pr}$ , ID) corresponding to an adversary defined identity $ID_i$.

3. The adversary can start querying another Oracle $O_{Decrypt}$ ($D_{IDi}$, $C_j$ ) that returns a plaintext $m_j \leftarrow$ IBE.Decrypt($D_{IDi}$, $C_j$) corresponding to an adversary defined Ciphertext$C_j$ and identity $ID_i$ .
4. The adversary picks two equal length plaintext messages $m_0$ and $m_1$ and an identity $ID_{encrypt}$. The adversary honestly passes <$m_0$, $m_1$, $ID_{encrypt}$> to the challenger.
5. The challenger picks a Random bit b and executes

   C ← IBE.Encrypt(*params*, $ID_{encrypt}$, $m_b$).

   The challenger gives C to the adversary.
6. The adversary continues querying the Oracle $O_{Extract}$ ($ID_i$) adaptively.
7. The adversary continues querying the Oracle $O_{Decrypt}$ ($D_{IDi}$, $C_j$) adaptively.
8. The adversary outputs a bit b' based on the Ciphertext C. If b = b' the adversary wins the game. Otherwise, the adversary loses the game. If the adversary queried the Oracle $O_{Extract}$ ($ID_i$) with $ID_i = ID_{encrypt}$ during step 2 or step 6 or if the adversary queried the Oracle $O_{Decrypt}$ ($D_{IDi}$ , $C_j$ ) with $C_j$ = C during step 3 or step 7, the adversary loses the game as well.

*Anonymous Identity-Based Encryption*

An IBE scheme called anonymous IBE (ANO-IBE) when the Ciphertext does not leak the identity of the recipient. In the overview illustrated in Figure 3.1, this implies that no eavesdropper on the insecure channel between Alice and Bob could derive that Bob is the recipient based on the information in the Ciphertext [9]. ANO-IBE defined with the help of a game that challenges an adversary similar to the IND-CPA game.

An IBE system is said to be anonymous if the adversary has negligible advantage trying to win the ANO-IBE game in Game 3. Again, the advantage of the adversary in winning the IND-CCA game illustrated in Game 2, is defined as

$$\text{Adv} = |\Pr[b = b'] - 1/2 |$$

More formally, an IBE system is ANO-IBE secure if for every adversary with advantage Adv in winning the ANO-IBE game illustrated in Game 3 there exists a negligible function μ (λ) such that Adv ≤ μ. Gentry [37] present the first scheme that combines the notions of IND-CPA and IND-CCA with ANO-IBE. Therefore, a system is then said to be IND-ANO-CPA secure or IND-ANO-CCA secure if it satisfies a modified version of the game in Game 3. For a more detailed discussion on the topic, the reader is referred to the original paper [37].

**Game 3:** Generic ANO-IBE [2]

Aim: An adversary is challenged to check the ANO-IBE security of an IBE scheme by a game.

Output: This ANO-IBE Game helps to define the concept of ANO-IBE security for IBE schemes.

1. The challenger runs <$\text{msk}_{\text{Pr}}$, *params*> ← IBE.Setup($1^{\lambda}$) and returns *params* to the adversary.
2. The adversary can start querying an Oracle $O_{\text{Extract}}(\text{ID}_i)$ that returns a Private Key $D_{IDi}$ ← IBE.Extract(*params*, $msk_{Pr}$ , ID) corresponding to an adversary defined identity $\text{ID}_i$.
3. The adversary picks a plaintext message m and an identity $\text{ID}_{\text{encrypt}}$. The adversary honestly passes<m, $\text{ID}_{\text{encrypt}}$> to the challenger.
4. The challenger picks a Random bit b and executes C ← IBE.Encrypt(*params*, $\text{ID}_{\text{encrypt}}$, m) if b = 0. If b = 1, the challenger computesC ← IBE.Encrypt(*params*, $\text{ID}_{\text{encrypt}}$, r) where r is a Random bit sequence with the same length as the message m. The challenger gives to the adversary.
5. The adversary continues querying the Oracle $O_{\text{Extract}}$ ($\text{ID}_i$) adaptively.
6. The adversary outputs a bit b’ based on the Ciphertext C. If b = b’ the adversary wins the game. If b ≠ b’ or if the adversary queried the Oracle $O_{\text{Extract}}(\text{ID}_i)$ with $\text{ID}_i = \text{ID}_{\text{encrypt}}$ during step 2 or step 5, the adversary loses the game.

### 3.2.5 Overview

Although Shamir [5] introduced an identity-based signature scheme based on RSA in 1984, the practical use of IBE remained an open problem before the concept of bilinear maps introduced. Boneh and Franklin were the first to propose the first practical implementation of IBE based on the Weil pairing in 2001. However, the security proof still depends on the Random Oracle assumption. Independently from Boneh-Franklin scheme [7], Sakai and Kasahara [59] proposed a different IBE scheme at the same period. However, due to lack of language problem and security proof, as this scheme presented in Japanese. This scheme was not much popular. Moreover, then Sakai and Kasahara proposed a revised version of their scheme that is proven IND-CCA secure in the RandomOracle model by Chen et al. Canetti et al. [49] introduced the first secure IBE scheme without relying onthe RandomOracle model.

Although all these references contributed to the evolution of IBE, not all of these schemes are ANO-IBE. The IBE scheme from Boneh and Franklin [7] is IND-ANO-CCA secure since IBE systems in the Random Oracle model are ANO-IBE. In the standard model, it appeared to be harder to construct ANO-IBE schemesat first sight, e.g. it can be proven that the scheme from Boneh and Boyen [61] is not anonymous in its original form. The scheme from Gentry [37] was the first anonymous IBE scheme in the standard model. Boyen and Waters [9] published almost synchronously another IBE scheme in the standard model that is also INDANO-CCA secure.

Algorithm 4: IND-ANO-CCA Boneh and Franklin IBE scheme

**Goal**: Alice wants to send an IBE encrypted message to Bob.

**Result**: Alice sends an IBE encrypted Ciphertext *c* that is successfully decrypted by Bob.

***Setup*** $(1^k)$:

a) The PKG assumes a prime $P \in \mathbb{G}^*$, where, $\mathbb{G}$ is the Group generated by a P.

b) Given $\hat{g}: \mathbb{G} \text{ X } \mathbb{G} \rightarrow \mathbb{F}$ is bilinear mapping, four hash function are

- $H_1: \{0,1\}^* \rightarrow \mathbb{G}^*$
- $H_2 : \mathbb{F} \rightarrow \{0,1\}^l$
- $H_3: \{0,1\}^n \text{ X } \{0,1\}^n \rightarrow \mathbb{Z}_q$
- $H_4: \{0,1\}^n \rightarrow \{0,1\}^n$

where $l$ denotes the length of a message.

c) The PKG Randomly chooses a master key $msk_{Pr} \in \mathbb{Z}_q$ and generate his Public Key $msk_{Pub} = msk_{Pr}.P$.

d) Moreover, then PKG publicly distribute the parameter $<\mathbb{G}, \mathbb{F}, H_1, H_2, H_3, H_4, msk_{Pub}>$.

***Extract (Params, ID, s):***

a) Compute $Q_{ID} = H_1(ID)$.

b) Compute $D_{ID} = msk_{Pr}.Q_{ID}$,

***Encrypt (Params ID, m)***: Now, user Randomly chooses $r \in \mathbb{Z}_q$ and may encrypt her message using user's identity ID by calculates

a) Compute $r=H_3(z,m)$ where $z \in \{0, 1\}^*$,

b) Compute $Q_{ID} = H_1(ID)$

c) Compute $H_2(\hat{g}\,(Q_{ID}, msk_{Pub}.))$.

d) The resulting Ciphertext $C = (U,V, W)$

$= <rP, z \oplus H_2(g^r), m \oplus H_4(z)>$ and sent to user2.

***Decrypt(Params ID, C)***:

a) Compute $g' = e(U, D_{ID})$ and $z' = V \oplus H_2(g')$

b) Compute $m' = W \oplus H_4(z')$

c) Compute $r = H_4(z'.m')$

d) Check $U= r'p$ if yes return the plaintext.

## 3.3 Key escrow

As we have seen that, user's identity (ID) directly used as the Public Key and the corresponding Private Key is generated by the PKG and stores in it. Therefore, an unusual property is inherent in the proposed IBE scheme. This property is called "*key escrow*."Moreover, thesecond is, after extraction of a user's Private Key, the PKG send it over the secure channel, making channel secure is difficult. Key escrow is a situation in which the users' Private Keys stored by an authorized party, such that under the certain criminal condition, the authorized party may access to those keys and can decrypt the private communication. Theses authorized parties might include government or any commercial department that may wish to tap the users' secret communication. However, what happens, when these authorized parties impersonate the users by revealing their

Private Key to criminals. This situation known as the key escrow problem, which may create a serious issue for users in certain situation as shown in figure 3.2.

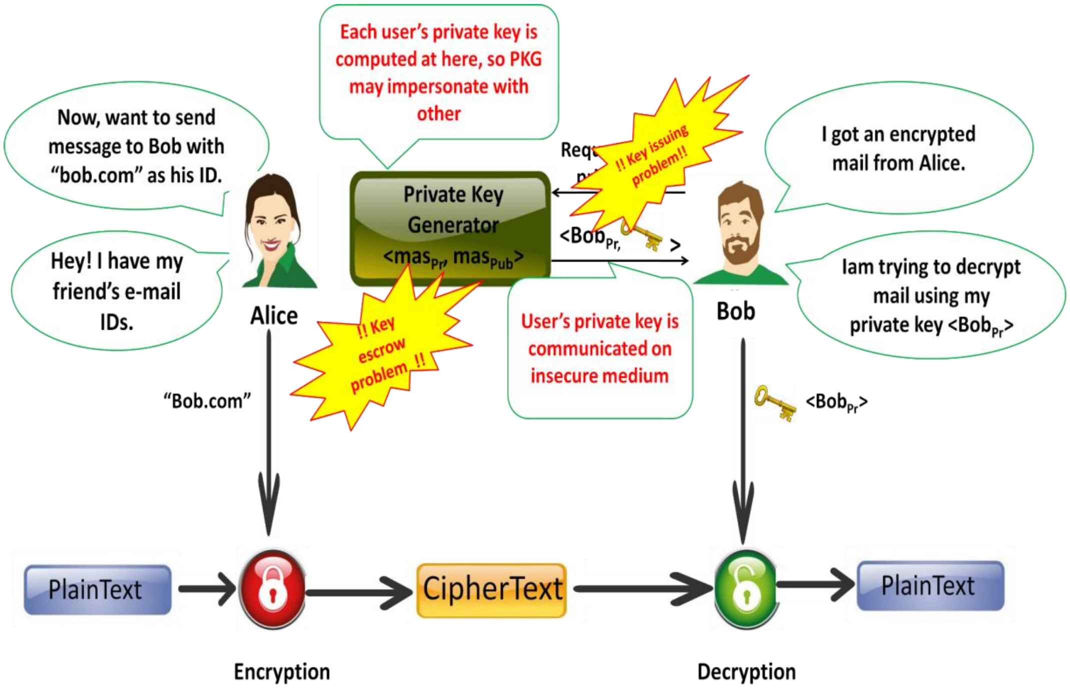
Each user's private key is computed at here, so PKG may impersonate with other
Now, want to send message to Bob with "bob.com" as his ID.
Hey! I have my friend's e-mail IDs.
Alice
Private Key Generator $<mas_{Pr}, mas_{Pub}>$
Req
!! Key issuing problem!!
$<Bob_{Pr},$ >
!! Key escrow problem !!
User's private key is communicated on insecure medium
I got an encrypted mail from Alice.
Iam trying to decrypt mail using my private key $<Bob_{Pr}>$
Bob
"Bob.com"
$<Bob_{Pr}>$
PlainText
CipherText
PlainText
Encryption
Decryption

Figure 3. 3 Key Escrow Problem in Identity-based encryption

The trusted authority (PKG) is the one who never has pre-enrolled and want to share a secret over an authenticated physical medium between all users joined to a same network. That means if two users or confidential channel. They may set up the required channel if and only if both users trust this authority. However, what does "trust" mean here? What parameters are there which defines how much possible a user to trust the authority. According to the Girault proposed in the scheme [22], he defines the levels of trust to PKG into three categories. As shown in Table 2.1, level 1 refers to those PKG, who can easily compute a user's Private Key, so that, he can impersonate any user without being detected, e.g. IBE. Level 2 refers to that PKG, who cannot compute a user's Private Key, but still can impersonate the user by generating fraud certificate e.g. certificates less signature scheme. That is why, it requires the need of level 3, in which PKG cannot computer users' Private Keys, and cannot impersonate any user without being recognized. Clearly, the level 3 is the most advantageous one. The certificate-based scheme is the one which achieved Level 3.

| Level | Can PKG compute user’s secret key? | Can PKG impersonate With another user? | Example |
|---|---|---|---|
| Level 1 | Yes | Yes | IBE |
| Level 2 | No | Yes | CL-PKC |
| Level 3 | No | No | CB-PKC |
| Level 4 | No | No | SCS |

Table 3. 1 Level of trust to PKG

By the logic behind the PKC, the Private Key is secret to the user and the Public Key publicly distributed over the network, need not be protected for confidentiality. It willmake them subjected to several active attacks (such as the substitution of a "false" Public Key to a "true" one in a PKI directory). That is why, with the key pair and user’s identity (ID), it is required to have user’s attribute (G) which guaranteed that $usk_{Pr}$is the original Public Key of the user**.**

## 3.4 Related work

This section includes the literature work of some approach trying to solve the key escrow problem in Public Key encryption.

### 3.4.1 Threshold Key Issuing

Boneh and Franklin introduced the solution of the key escrow problem in their original proposal of Identity-based encryption [7]by distributing the master private $msk_{Pr}$ among different PKG rather than one PKG, instead of protecting the privacy of a user’s Private Key. Master Private Key is successfully distributed among n PKG using the method of threshold cryptography in very robust and robust way. Suppose there is n PKG. the master Private Key$msk_{Pr}$ securely distributed among n PKG easily by t-out-of-n pattern by giving the one share of $msk_{Pr}^{i}$ to each PKG using the technique of the Shamir secret sharing [54]. Each PKG$^{i}$ computes Private Key using their master key share $msk_{Pr}^{i}$as $D_{ID}^{i}$= $msk_{Pr}^{i}$Q$^{ID}$ where, $Q_{ID}$ = H(ID).User can then compute $D_{ID}$ as $D_{ID}$= $\sum \lambda_{i.} D_{ID,}^{i}$ where the $\lambda_i$’s are the appropriate Lagrange coefficients. This scheme can be well made against dishonest PKG without using zero-knowledge proofs [62].

In this scheme, each PKG are supposed to have the equal job. Thus, they have to validate user's identity separately. Verification of the sign depends on the correctness of user identification. It may quite a difficult task. That was the major disadvantage of this approach. Instead of n PKG's check the user's identity and sign the ID, it can happen to verify the user's identity by single PKG and other n-1PKG just sign the ID. So, it is easy to say that this is not a real distributed key generation.

### 3.4.2 Certificate-Based Encryption

Gentry [45] in 2003 proposed a certificate-based encryption scheme allow key privacy facility using the user-chosen secret key. Subsequently, Joseph K. et.al [44] introduced the certificate-based signature without pairing**.** In certificate-based encryption, trusted the third party called the certification authority (CA) issues a certificate to certify the user's Public Key. Any user who wishes to encrypt the message with the Public Key must first verify the certificate to validate the Public Key. Suppose CA choose $msk_{Pr} \in \mathbb{Z}_q$ as his master key and generate Public Key $msk_{Pub} = msk_{Pr}$.P, where P is generator of given group $\mathbb{G}$.User chooses a secret Random value $usk_{Pr} \in \mathbb{Z}_q$ and generates his Public Key $usk_{Pub} = usk_{Pr}$ P. He sends $usk_{Pub}$ and ID to the CA and requests him to issue a certificate. CA checks the identification of the user. He computes$Q_{ID}$ = $H_1(msk_{Pub,}usk_{Pub}$, ID, T) where T denotes a validity period. Finally, CA computes a certificate $Cert_{ID} = msk_{Pr}Q_{ID}$ and sends to the user. On Receiving $Cert_{ID}$, user computes $Q^{1}_{ID}$ = $H_1$(usk$_{Pub}$, ID) and $D_{ID}$ = $Cert_{ID}$ + $usk_{Pr}$.$Q^{1}_{ID}$ Where, $D_{ID}$ is a user's decryption key. To encrypt a message m for the user, a sender computes $Q'_{ID} = H_1(usk_{Pub}$, ID), $Q_{ID}$ = $H_1(msk_{Pub\,,}usk_{Pub}$ , ID, T) and g = e($msk_{Pub}$, $Q_{ID}$)e($usk_{Pub\,,}$ $Q'_{ID}$). Choosing a Random number r $\in \mathbb{Z}_q$, he computes a Ciphertext C = (U, V) = (rP, m $\oplus$ $H_2(g^r)$)To decrypt the cipher text C = (U, V ), the user recovers the plaintext as V $\oplus$ $H_2$(e(U, $D_{ID}$)) = m. This approach is very useful to avoid key escrow problem and successfully provides the alternate of secure key issuing between CA and user by sending a certificate $Cert_{ID}$ over the public channel or published. With the use of the certificate, it loses the advantage of ID-based scheme. Two main issues with this approach are: one is, it inherits the key the revocation problem which requires the largest amount of storage space and also computing time, verify and revoke certificates. This scheme gives the implicit as well explicit certification to the user. Second, the sender cannot be sure that $usk_{Pub}$ is really the Public Key of the recipient without getting the certificate $Cert_{ID}$ from the CA. Because, instead of the secret key, CA yields the certificate $Cert_{ID}$ to the user.

### 3.4.3 Certificate-Less Public Key Cryptosystem

Al-riyami et al. proposed a model for Public KeyEncryption known as Certificate-less Public Key encryption [46], which remove the key escrow problem without need any certificate to certify the Public Key with entity ID. This approach uses the user-chosen secret key as similar way as the self-certified key. PKG choose a master key $msk_{Pr} \in \mathbb{Z}_q$ and generates $msk_{Pub}$=$msk_{Pr}$.P. Given identity ID, the user requests the PKG to issue a Private Key. The user selects a Random integer $usk_{Pr} \in \mathbb{Z}_q$ and computes the Public Key $usk_{Pub}$=<$N_1$, $N_2$> such that $N_1$=$usk_{Pr}$P, and $N_2$ = $usk_{Pr}.msk_{Pub}$ and sends $usk_{Pub}$ and ID to the PKG and request to issue the Private Key. PKG first check the identification of the user and generates the partial Private Key $D'_{ID}$= $msk_{Pr}$.$Q_{ID}$ where, $Q_{ID}$=$H_1$(ID, $usk_{Pub}$) and sends $D'_{ID}$ to the user. After successfully receiving the partial Private Key, the user extracts his Private Key $D_{ID}$ = $usk_{Pr}$.$D'_{ID.}$ Given message m, sender encrypts the message to give C = (U, V) = (rP, m $\oplus$ $H_2(e(Q_{ID}, usk_{Pub})^r)$) where, $Q_{ID}$=$H_1$(ID, $usk_{Pub}$) and r is Random integer. To decrypts Ciphertext C= (U, V), user decrypt the encrypted message as V $\oplus$ $H_2(e(D_{ID}, U))$.

This scheme successfully eliminates the requirement of a certificate. Thus, there is neither any signature nor certificate which assurances sender whether $usk_{Pub}$ is the real Public Key of the receiver until the communication successfully completed. In spite of the fact that, this approach attains the key privacy, it permits only implicit authentication to the users.

### 3.4.4 HIBE

Horwitz and Lynn in [33] suggested a hierarchy of PKGs in identity-based encryption. First, they introduce the 2 Level hierarchical IBE, where at first level total collusion-resistance is obtained and partial collusion-resistance at second Level. So, it has only limited resistance to user collusion. In scheme [39], Gentry and Silverberg extend the

Boneh and Franklin scheme [7] to construct the fully flexible Hierarchical ID-based encryption scheme. PKGs generate Private Keys only to the users immediately below them in the hierarchy as shown in figure 3.4.

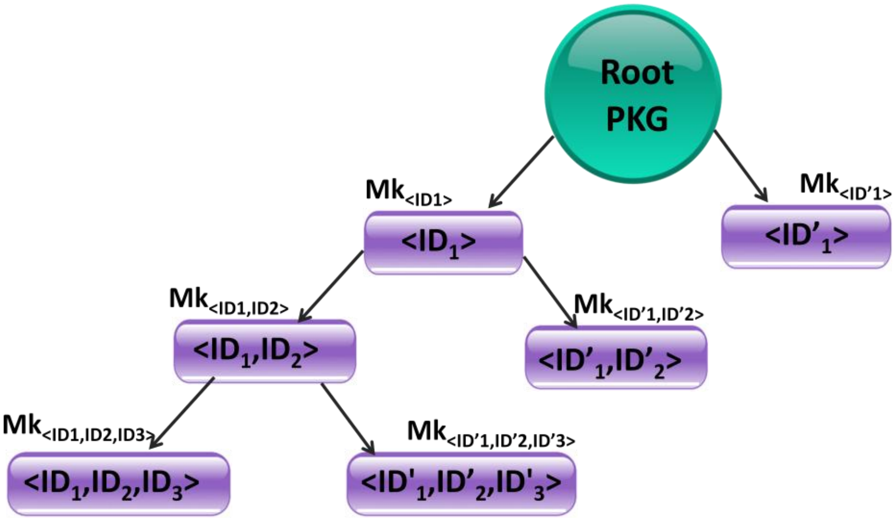

Figure 3. 4 Hierarchy of Private Key Generators

In this scheme, instead of their single identity (ID), Users are identified by a tuple of identities <$ID_1$, $ID_2$, …$ID_i$> that is their ancestor's identity in the hierarchy. Note that, we write [$ID_i$] to denote the <$ID_1$, $ID_2$…$ID_i$>. The Private Key corresponds to [$ID_I$] is<$S_{[IDI]}$, $Q_{\Pi}$, $Q_{[ID1]}$,$Q_{[ID2],}$…. $Q_{[IDI-1]}$>where, $S_{[IDI]}$=$S_{[IDI-1]}$ + $S_{[IDI-1]}$.$P_{[IDI]}$ such that $Q_{[IDI]}$=$S_{[IDI]}$.$P_{\Pi}$ AND $Q_{\Pi}$ =$S_{\Pi}$.$P_{\Pi}$. $Q_{\Pi}$ and $S_{\Pi}$ are Random numbers chosen by the Root PKG. HIBE is used to generate short-lived keys for portable computing devices [33] and access control to confidential pervasive computing information [30].

### 3.4.5 VIBE

To avoid the problem of key escrow, Zhaohui Cheng et al.[31] introduce another key pair of public and Private Key <$Y_{ID}$, $usk_{Pr}$> into the Boneh-Franklin scheme [7], where $usk_{Pr}$∈ $\mathbb{Z}_q$ is the user's Private Key keep secret to the user only and $Y_{ID}$ is derived from $usk_{Pr}$. The rest of the algorithm as worked as follows: In Setup phase, PKG's master Private Key $msk_{Pr}$ keep secret to it and publicly distribute the parameter is *Params* = <$\mathbb{G}$, $\mathbb{F}$, $H_1$, $H_2$, $H_3$, $H_4$, $msk_{Pub}$>. Then, user join the PKG to obtain his Private Key $D_{ID}$ =

$msk_{Pr}$Q$_{ID}$, where Q$_{ID}$ = H$_1$(ID). The user chooses a Random integer $usk_{Pr} \in \mathbb{Z}_q$ and computes Y$_{ID}$= <Y$_1$, Y$_2$> whereY$_1$ = $msk_{Pr}$.P and Y$_2$=$usk_{Pr}.msk_{Pub}$. To encrypt the message, first check e(Y$_1$,$msk_{Pr}$) ?= e(Y$_2$,P) for Y$_1$, Y$_2 \in \mathbb{G}$ then computes Ciphertext C = (U, V, W) = <rP, z $\oplus$H$_2$(g$^r$), m $\oplus$ H$_4$(z)>, where, g= e($msk_{Pub}$ +Y$_1$, Q$_{ID}$), Q$_{ID}$ = H$_1$(ID), r=H$_3$(z, m) and z $\in$\{0, 1\}* and sent to the receiver. For Ciphertext, the receiver can decrypt *C* by computing g’ = e(U, D$_{ID}$), z’ = V $\oplus$H$_2$(g’), m’ = W $\oplus$ H$_4$(z’),r = H$_4$(z’, m’) and check U= r’.p if yes return the plaintext.

In encryption and decryption, it has been seen that g’= g$^r$. Consequently, z’ in decryption and z in encryption are equals. Thus, on applying decryption on the Ciphertext, we get the original message m. Moreover the encryption; this approach is implemented signature and key agreement also. As compared to certificate less encryption, this scheme is slightly slower.

## 3.5 Comparison of existing ID-based encryption scheme

A solution to key escrow proposed by Boneh and Franklin [7] is defined as the master key is derived from the number of different PKG so that each PKG have the knowledge of the partial secret of the Private Key. No single PKG have a complete knowledge of it. So, it requires the largest amount of infrastructure and computational cost. Similarly, hierarchal ID-based encryption [33, 39] needs extra infrastructure and computational cost. In certificate-based encryption [45], key revocation is the biggest negative consequence that also requires a large amount of space and computational time for certificate storage, verify and revoke. To tackle this problem, Al–Riyami et al. [46] successfully eliminates the requirement of a certificate, but his scheme only provides implicit authentication of the Public Key. The sender can never know whether the Public Key is the real Public Key of the receiver. In VIBE [31], to solve the key escrow problem there is another public and Private Key pair. However, this scheme is slightly slower than the certificate-less Public Key encryption because it has an extra point addition operation.

On the basis of computation cost, Table3.2 shows the comparisons of different Public Key encryption schemes that avoid key escrow problem. On the basis of trust level, Table 3.3 shows the comparisons of different Public Key encryption schemes.

| Scheme | Key Generation | Encryption | Decryption |
|---|---|---|---|
| Threshold key issue | 2nM | 1M+1P+1E | 1M+1P |
| CB-PKC | 4M | 2M+1P+1E | 1P |
| CL-PKC | 4M | 1M+1P+1E | 1M+1P |
| HIBE | 2tM | tM+1P+1E | tM +tP |
| VIBE | 4M | 1M+3P+1E | 3M+1P |

Table 3. 2 Comparison of Computation Cost of variant of Public Key encryption which avoids key escrow problem, where, M: Point multiplication, P: No of pairing operation, E: Exponentiation, t: no of user's identities in HIBE from root to leaf and n: number of PKGs in threshold issue scheme

| Scheme | Public information | Trust level |
|---|---|---|
| Threshold key issue | ID | Level 1 |
| CB-PKC | ID, $usk_{Pub}$ | Level 3 |
| CL-PKC | ID, $usk_{Pub}$, C | Level 2 |
| HIBE | ID$_1$,..ID$_i$ | Level 1 |
| VIBE | ID, N$_1$. N$_2$ | Level 3 |

Table 3. 3 Comparison of public information and trust level to PKG of different scheme, where, $N_1 = \boldsymbol{usk_{Pr}}$P: point multiplication of user secret $\boldsymbol{usk_{Pr}}$ with group generator P, $N_2 = \boldsymbol{usk_{Pr}.msk_{Pub}}$: point multiplication of user secret $\boldsymbol{usk_{Pr}}$ with PKG's Public Key and C: commitment of user secret key with the PKG's Public Key.

## 3.6 Summary

After understanding the basic mathematics discussed in chapter 2, this chapter summarized the fundamental building blocks of cryptography. The first section of the chapter introduces the Public Key encryption includes its advantage and disadvantage of managing certificate for the user. In the second section, we introduce the main part of the thesis; identity-based encryption includes its basic definition, compared with PKI, applications, advantages, disadvantages and security definition. In the following section, key escrow problem introduced. In the next section, some literatures related to the key escrow problem are discussed in details. Finally, this chapter comprises with the comparison of existing scheme approach to solving the key escrow problem.

Chapter

# Secure Key Issue 4

This chapter expertly designs an ID-based encryption scheme that serves as a fair balance between the users and government agency. Here, "Fair balance" means that the user has the right to privacy and government has the right to monitor the user's criminal activity over the secure communication.  In the first section (section 4.1), we define some existing solution regarding the key escrow problem that will useful for describing given proposed model. In Section 4.2, the proposed scheme is described by defining cryptographic goals based on the earlier threat model. In the following section, we design decisions on how to achieve these objectives and how this impacts on our model. Section 4.2 concludes with a concrete proposal in the form of an algorithm along with an evaluation section motivating why our cryptographic design goals successfully met. In the next section (section 4.3), we define the security model for our scheme which include IND-CCA Type I and Type II attacks. Mathematical implemented of our model applied in section 4.4.

## 4.1 Models of existing solution

We already discuss the several approaches [22, 31, 39, 44, 45, 46, 47, 48] that avoid the Key escrow problem inherited in generic IBE scheme [5] in section 3.4. The most common work on Identity-based encryption avoid key escrow problem is introduced the Boneh-Franklin [7] using threshold cryptography technique [50]. But it requires large amount infrastructure and more computational cost to implement such a scheme.

Likewise, same drawbacks exist with the hierarchal IBE [9, 12] where the public key user is identified by the tuple of identities from roots to him. Certification based encryption [45] does not preserve the identity-based encryption property along with the public key revocation is additionally a significant disadvantage. However, the public key revocation is not the big issue; it can be removed by managing the key revocation list. That becomes a new problem and may require the high volume of space to store the certificates and computational time to verify those certificates. To solve the problem of managing revocation list, certificate-less encryption [46] was introduced. Indeed, this public key encryption variant solve many problems: key escrow problem, public key revocation, management of certificates, etc. it provides only implicit authentication to the public key. That means the sender will never know whether the given public key is the original public key of the recipient.

## 4.2 Proposed Model: M-IBE

In this section, we describe the cryptographic requirements in designing the proposed model.

### 4.2.1 Cryptographic Goals

The standard general goals stated that the proposed scheme should be user-friendly and efficient to use. Addition to these generals' goals, it is also to define some cryptographic goals. A well-made encryption scheme should be able to achieve following cryptographic goals:

- *Authenticity*: The recipient has reasonable assurance that the claimed sender sends the message.
- *Confidentiality*: The message protected from the disclosure of unauthorized person.
- *Integrity:* The message protected from being modified by the unauthorized users.
- *No key escrow*: User's private keys only disclosed to the owner of claimed identity. No other user should be able to retrieve the private key.
- *Key validation*: Each user in the system should be able to verify the correctness of their private key.

- *Limited key validity*: User private key should not be valid for limited period of the time.

### 4.2.2 Design to Achieve Goals

In this section, our model is modified according to achieving the following cryptographically goals:

*Authenticity:*

Authenticity can be achieved by depending on authenticated encryption scheme. The authentication mechanism still depends on the security guarantees the IBE scheme. Since there is a trusted third party known as private key generator who verifying the user's unique identity corresponding to their public key.Accordingly, such an IBE confirms that a message encrypted with a public identifier can only be seen by the corresponding private key. If the authentication mechanism is insufficient, thus anyone could use to impersonate the user. Our proposed based on the Boneh-Franklin IBE scheme. Therefore, the scheme achieves authenticity.

*Confidentiality:*

Confidentiality can be achieved by applying an encryption scheme before sending a message. Identity-based encryption (IBE) can reach both confidentiality and the general design goals of usability and applicability. During the design of our scheme, we can consider have several IBE schemes: Boneh and Franklin IBE [7], Sakai and Kasahara IBE [59] and Gentry IBE [37]. For the convenient of the desirable issue, we use Boneh-Franklin IBE scheme as the encryption scheme.

*Integrity:*

As similar to authenticity, integrity can also be achieved depending on the security guarantees on the IBE scheme. If the scheme sufficiently authenticated, no one can impersonate the user. As discussed in our proposed, our proposed scheme is derived from the Boneh-Franklin scheme.

*No Key Escrow:*

IBE scheme inherently implies a property known as key escrow, which is undesired in the most practical system. To bypass the key escrow problem, multiple PKG implemented as a distributed key generation system for IBE and several other schemes discussed in Section 3.4. To avoid the key escrow problem, we have proposed a scheme will be discussed in section 4.2.4.

*Key validation:*

Key validation can be achieved by using the secret information to lock the partial private key on trusted third party and unlock is using his secret information. Each can verify the private key with the use of some parameter as shown in Algorithm 4.1

*Limited key validity*

IBE scheme does not provide the revocation of the public key facility in the generic scheme. To attain the revocation of the public key, we can embed an expiration date along with the user identity ID. Thus, as a part of the public key, the expiration date should be publically available to everyone.

### 4.2.3 Current Proposed Model

To bypass the key escrow problem inherited in IBE scheme, an additional entity known as private key privacy organization (PKPO) is added to our model. For the convenience, we call our model as M-IBE. Before explain the model, some definition should be required to understand out model.

**Definition 4.1: (Key escrow)** In Identity-based encryption, user’s private key is generated and stored in PKG. This unusual property inherited in IBE is called the key escrow.

**Definition 4.2: (Key escrow Problem)** An unusual property inherited in generic Identity-based encryption scheme allow:

1. PKG to use user’s private key in a mischievous activity without their permission and pretended as a user.
2. The user may deny that message is not send by him and claim to PKG.

This situation is called the key escrow problem.

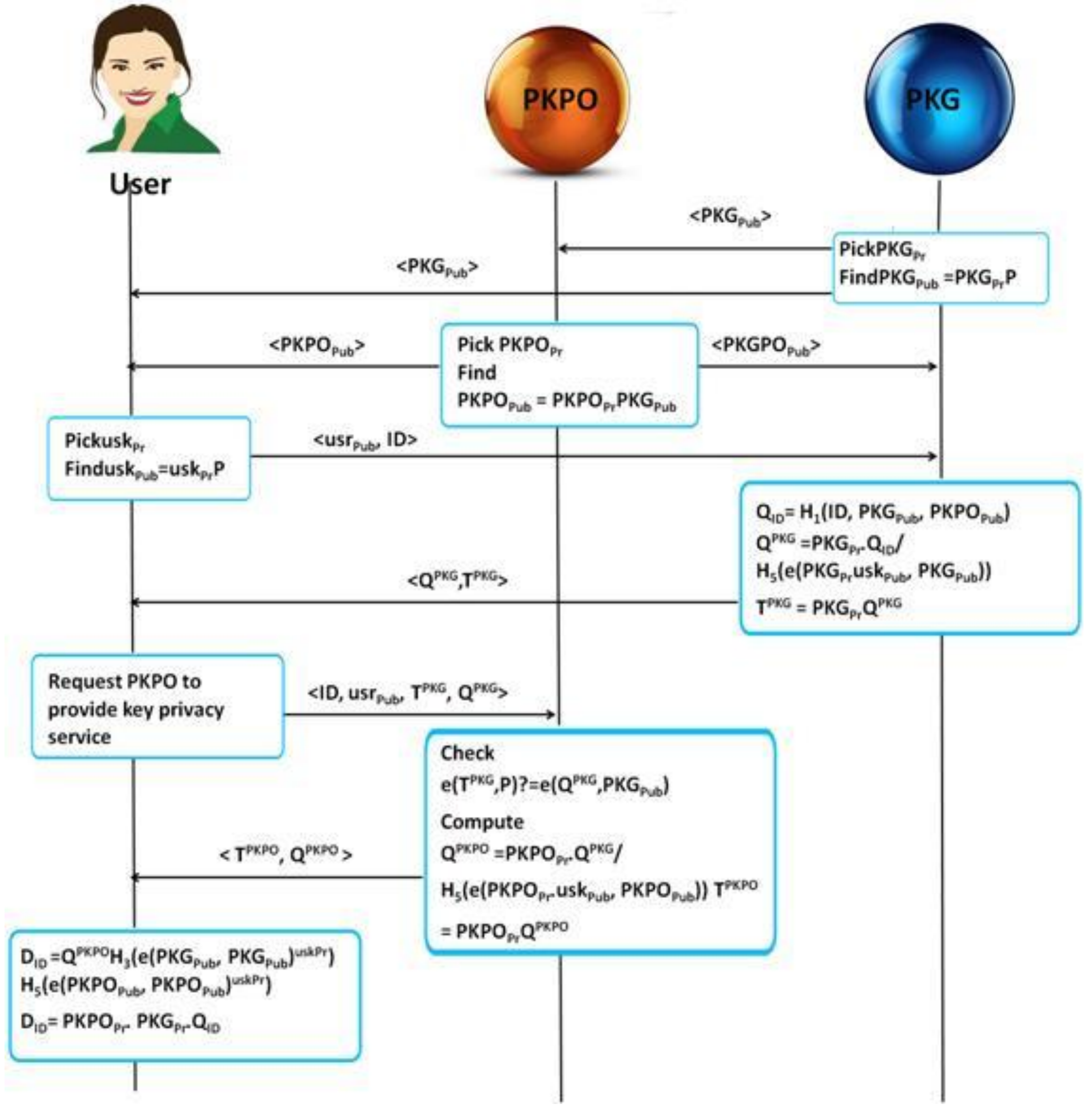
User
PKPO
PKG
$<PKG_{Pub}>$
$PickPKG_{Pr}$
$FindPKG_{Pub} = PKG_{Pr}P$
$<PKG_{Pub}>$
$<PKPO_{Pub}>$
$Pick\ PKPO_{Pr}$
Find
$PKPO_{Pub} = PKPO_{Pr}PKG_{Pub}$
$<PKGPO_{Pub}>$
$Pickusk_{Pr}$
$Findusk_{Pub}=usk_{Pr}P$
$<usr_{Pub}, ID>$
$Q_{ID}= H_1(ID, PKG_{Pub}, PKPO_{Pub})$
$Q^{PKG} =PKG_{Pr}.Q_{ID}/$
$H_5(e(PKG_{Pr}usk_{Pub}, PKG_{Pub}))$
$T^{PKG} = PKG_{Pr}Q^{PKG}$
$<Q^{PKG},T^{PKG}>$
Request PKPO to provide key privacy service
$<ID, usr_{Pub}, T^{PKG}, Q^{PKG}>$
Check
$e(T^{PKG},P)?=e(Q^{PKG},PKG_{Pub})$
Compute
$Q^{PKPO} =PKPO_{Pr}.Q^{PKG}/$
$H_5(e(PKPO_{Pr}.usk_{Pub}, PKPO_{Pub}))\ T^{PKPO}$
$= PKPO_{Pr}Q^{PKPO}$
$< T^{PKPO}, Q^{PKPO} >$
$D_{ID} =Q^{PKPO}H_3(e(PKG_{Pub}, PKG_{Pub})^{uskPr})$
$H_5(e(PKPO_{Pub}, PKPO_{Pub})^{uskPr})$
$D_{ID}= PKPO_{Pr}. PKG_{Pr}.Q_{ID}$

Figure 4. 1 Steps for Private Key Issue

**Definition 4.3: (Private Key Generator)** A single PKG to check user's identification and provide a partial private key to the user.

**Definition 4.4: (Private Key Privacy Organization)** A single key privacy agency is a Non-Government Organization, who has work between the user and PKG (Government agency) for the sake of users. PKPO is introducing to provide the privacy service to the private key by provide their signature in a confused manner.

**Definition 4.5 (Judge)** In case of malicious PKG, PKPO or user, anyone can file a case in court, such that judge recovered the private key in the presence of all three entities. Consequently, monitor the malicious communication.

**Definition 4.6: (User)** An entity who want privacy in their communication with other entity over an insecure medium.

**Assumption:** Here we assume that PKG and PKPO are never colluding each other, So that malicious PKG and/or PKPO can never use their private key.

This Model consists of two algorithms (Setup and Key Extract). Further, KenGen algorithm divided in three processes (Partial key issuing, Key securing process and Key Fetching). First PKG runs the Setup algorithm to creates the $< pkg_{Pr}\,,\ pkg_{Pub} >$, sequentially PKPO generates his key pair $< pkpo_{Pr}\,,\ pkpo_{Pub} >$ and publish public parameter keeping their secret key is to them self. To get the partial key, user request to PKG by sending his identity and hash of his secret info as a confusing factor. PKG checks the user identity and provide the partial private to the user in confused manner if and only if the user is legitimate. Now, the user requests PKPO to provide privacy service, PKPO return the original private key. Only the legitimate user who has a secret key for confusing factor unlocks the message to get the original private key. The user fetches the original private key if he has a secret key corresponding to unlocking the confusing factor. For a given user identity and message, the user encrypts/sign the message. For a particular Ciphertext and his private key, the user can decrypt the message.

### 4.2.4 Acquire secure key Scheme

In this section, we propose a model for Identity-based encryption free from key escrow problem as shown in Algorithm 4.1 this model consists of four algorithms (Setup, Key Extract, Encrypt, and Decrypt).

**Algorithm 4.1 Acquire secure private key**

**Aim**: Alice wants to request the PKG to acquire his private key securely.

**Result**: PKG and PKPO are issuing a private key.

1. **Setup**: The PKG assumes a prime $P \in \mathbb{G}^*$, where, $\mathbb{G}$ is the Group generated by a P. Given e: $\mathbb{G} \text{ X } \mathbb{G} \rightarrow \mathbb{F}$ is bilinear mapping, $H_1$: $\{0,1\}^* \rightarrow \mathbb{G}^*$ and $H_2 : F \rightarrow \{0,1\}^{\ell}$ , $H_3$: $\{0,1\}^n \text{ X } \{0,1\}^n \rightarrow \mathbb{Z}_q$, $H_4$: $\{0,1\}^n \rightarrow \{0,1\}^n$ and $H_5$: $\mathbb{G}_2 \rightarrow \mathbb{Z}_q$ are four hash function, where $\ell$ denotes the length of a message. The PKG randomly chooses a master key $pkg_{Pr} \in \mathbb{Z}_q$ and generate his public key $pkg_{Pub} = pkg_{Pr.}P$. And then KPA randomly chooses a key $pkpo_{Pr} \in \mathbb{Z}_q$ and create his public key $pkpo_{Pub} = pkpo_{Pr.}\, pkg_{Pub}$. Now, publicly distribute the parameter <$\mathbb{G}$, $\mathbb{F}$, $H_1$, $H_2$, $H_3$, $H_4$, $H_5$, $pkg_{Pub}$, $pkpo_{Pub}$>.
2. **Key extract:** As shown in figure 4.1, three entities (user, PKG, and PKPO) are participating to issue a private key. This process includes the following three stages:
    - **Partial key supply**: User chooses $uskg_{Pr} \in \mathbb{Z}_q$ and generate his public key $usk_{Pub} = usk_{Pr.}P$and request to PKG to provide partial private key by giving $uskg_{Pub}$and ID as follows:
        - Check the identification of users
        - Compute the public key of user as

          $$Q_{ID} = H_1(ID, P_0, P_1)$$
        - Compute Partial private key

          $$Q^{pkg} = \frac{pkg_{pr}.Q_{ID}}{H_5\left(e(pkg_{Pr}usk_{Pub}, pkg_{Pub})\right)}$$

          $$T^{pkg} = pkg_{Pr}.Q^{pkg}.$$
        - Moreover,sendsit to the user
    - **Key securing**: User request PKPO to provide key privacy service by sending ID, $usk_{Pub}$, $T^{pkg}$ and $Q^{pkg}$
        - Check e( $T^{pkg}$,P)?=e($Q^{pkg}$,$pkg_{Pub}$)

    - Compute $\mathrm{Q^{pkpo}} = \frac{\mathrm{pkpo_{Pr}.Q^{pkg}}}{\mathrm{H_5\big(e(pkpo_{Pr}usk_{Pub}\ ,pkg_{Pub})\big)}}$

$$T^{pkpo} = \mathrm{pkpo_{Pr}.\,Q^{pkpo}}.$$

    - Send $T^{pkpo}$ and $\mathrm{Q^{pkpo}}$ to the user.

- **Key fetching**: User retrieves his private key

$$D_{ID} = Q^{pkpo}\mathrm{H_5(e(pkg_{Pub,}\ \ pkg_{Pub}\ )^{usk_{Pr}}).\,H_5(e(pkpo_{Pub,}\ \ pkpo_{Pub})^{usk_{Pr}})}$$

$$D_{ID} = \mathrm{pkg_{Pr\ .}pkpo_{Pr}}Q_{ID}$$

The user can check the correctness of his private key by $\mathrm{e(Q_{ID,}\ pkg_{Pub})}$ ?= $\mathrm{e(D_{ID}, P)}$.

## 4.3 Security Model for M-IBE

Now, we are ready to define the adversaries for M-IBE scheme. The general security definition for IBE requires indistinguishability of encryptions against a fully-adaptive chosen cipher text attacker (IND-CCA). By this definition, we have two entities, the adversary $\mathcal{A}$, and the challenger$\mathcal{C}$. After presenting the random public key, the adversary controls in three phase. In phase 1, $\mathcal{A}$ randomly constructs decryption queries on the Ciphertext. In challenge phase, $\mathcal{A}$choose $M_0$, $M_1$ and C* for messages $M_b$ given by the challenger, where $M_0$, $M_1$ are two random message and C* is challenged Ciphertext. In phase 2, $\mathcal{A}$ continues to construct more decryption queries, indeed, cannot have info for the decryption of C*. Finally, $\mathcal{A}$ guess bit b’ corresponding to b. The $\mathcal{A}$'s advantage is defined to be

$$\mathrm{Adv(A)} = 2(\Pr[\mathrm{b'} = \mathrm{b}] - \frac{1}{2})$$

Here, we explore the [BF] model to permit adversaries to extract the partial private keys, or private key, or both, for random Identities, replace the public key with identity with a random value.

List of action that an adversary can taken against an M-IBE scheme are given below:

- *Partial key supply*: To derive the partial key <$Q^0$, $T^0$> for user A, $\mathcal{C}$ provides the output by running the algorithm Partial-key-supply.

- *Key securing*: To acquire the mystified private key $Q^1$ for user A, $\mathcal{C}$ gives output by running the algorithm key-securing.
- *Key obtaining*: $\mathcal{A}$ make a request for user's private key. To compute the real private key, $\mathcal{C}$ runs the algorithm key-extract if the corresponding public key is not changed.
- *Request public key*: Suppose $\mathcal{A}$ has public keys. To calculate the public key $P_A$ for user A, $C$ runs the algorithm set-public-key and respond to$\mathcal{A}$.
- *Replace pubic key*: $\mathcal{A}$ can adaptively replace the public key $P_A$ for user A with any random $P'_A$.
- *Decryption query*: To get private key $S_A$, $\mathcal{C}$ runs algorithm Set-Private-Key and then decryption algorithm and responds to $\mathcal{A}$, if$\mathcal{A}$ has not substituted the user's public key. Otherwise, $\mathcal{C}$ could not decrypt. However, our need is that $\mathcal{C}$ decrypts Ciphertexts for those public keys have been substituted. However, $\mathcal{A}$ is permissible to substitute the public key for $ID_{ch}$ with a new ID and then request a decryption of C*.

We assumed that adversaries who have master-key were not permit to substitute public keys. Here, we will try that our PKG achieve the same level of trust as CA in a traditional PKI. So we will classify adversary into two types, with different potential:

*M-IBE Type I Adversary*

Denoted as $\mathcal{A}_I$, such adversaries do not have master-key. Indeed, $\mathcal{A}_I$ can request public keys and substitute it with new values of its choice, extract partial private and private keys and constructs decryption queries, for each identity of its choice. Additionally, some limitations on adversary $\mathcal{A}_I$ are:

1. Given $ID_{ch,}$ $\mathcal{A}_I$ cannot extract the private key.

2. If the user's public key already been substituted, then $\mathcal{A}_I$ cannot request the private key for any identity.
3. Before challenge phase, $\mathcal{A}_I$ do not allow to substitute the public key for the challenge identity $ID_{ch}$ and extract the partial private key.
4. In Phase 2, $\mathcal{A}_I$do not allow to construct a decryption query on the challenge Ciphertext C* with an identity $ID_{ch}$ and public key$P_{ch}$.

*M-IBE Type II Adversary*

Denoted as $\mathcal{A}_{II}$, such adversaries have master-key. Indeed, $\mathcal{A}_I$ cannot substitute the public key. Using master-key, Adversary $\mathcal{A}_{II}$ can compute partial private keys. Additionally, some limitations on adversary $\mathcal{A}_I$ are:

1. $\mathcal{A}_{II}$does not allow substituting public keys.
2. $\mathcal{A}_{II}$does not allow extracting the private key for $ID_{ch}$.
3. In Phase 2, $\mathcal{A}_I$ do not allow to construct a decryption query on the challenge CiphertextC* with an identity $ID_{ch}$ and the public key$P_{ch}$.

*Chosen Ciphertext security for M-IBE*

M-IBE scheme is semantically secure against an adaptive chosen Ciphertext attack ('IND-CCA secure") if no polynomial bounded adversary $\mathcal{A}$of Type I or Type II has a non-negligible advantage against the challenger in the following game:

- Setup: For security parameter K, Challenger $\mathcal{C}$ runs the Setup algorithm. It responds $\mathcal{A}$ the output of system parameters *params*. Challenger $\mathcal{C}$ keepsthe master key for Type I adversary. Otherwise, it gives to$\mathcal{A}$.
- Phase 1:$\mathcal{A}$provides the number of requests. Each request for partial private key extraction, a Private Key extraction, public key, a substitute public key or decryption query for an individual user. According to the rules defined above, these queries can be run adaptively.

- Challenge Phase: $\mathcal{A}$responds the challenge identity $\mathrm{ID}_{ch}$ and two equal length message $M_0$, $M_1 \in$ M. The Challenger randomly chose a bit b $\in$ {0, 1} and computes C*. If encryption gives $\perp$, then $\mathcal{A}$ lost the game. Otherwise, C* is given to$\mathcal{A}$.
- Phase 2: According the rule defined above, $\mathcal{A}$ provides a second sequence of requests similarly in Phase 1.
- Guess: A responds a guess b'$\in$ {0, 1}. If b'=b, $\mathcal{A}$ wins the game with advantage$\mathrm{Adv(A)} = 2\left(\mathrm{Pr}[\mathrm{b}' = \mathrm{b}] - \frac{1}{2}\right)$

## 4.4 M-IBE schemes from Pairing

In this section, we implement identity-based encryption free from key escrow problem based on our proposed model. The first scheme, BasicM-IBE is identical to the BasicIdent scheme of [7] and contains the basic of our most important scheme FullM-IBE. The master scheme is identical to scheme FullIdent of [7]. Assuming the difficulty of GBDHP, our scheme is IND-CCA secure.

### 4.4.3 Basic M-IBE scheme

Here, we describe the four algorithms required to understand the fundamental ideas underlying our scheme that is identical to BasicIdent of [7].

**Algorithm 4.2 Identity-based encryption free from key escrow problem**

**Aim**: Given Bob identity $\mathrm{ID}_B$ and private $D_{ID}$ obtaining from algorithm 4.1, Alice wants to send an encrypted message to Bob so that no other (including PKG) than Bob can decrypt the message.

**Output**: Alice sends an encrypted Ciphertext *C* that is successfully decrypted by Bob without lose any confidentiality.

1. **Setup**: This phase is identical to setup phase of Algorithm 4.1.
2. **Key extract**: Identical to Key extract phase in algorithm 4.1.
3. **Encrypt:** Alice randomly chooses r $\in\mathbb{Z}_q$ and may encrypt her message

using Bob identity ID by

- Compute $Q_{\mathrm{ID}}$ = $H_1$(ID, $pkg_{Pub}$ , $pkpo_{Pub}$ ) and g=$H_2$(e ($Q_{\mathrm{ID}}$, $pkpo_{Pub}$)).
- The resulting Ciphertext $C = (U,V)$ = <rP, m⊕$H_2(g^r)$ >
- C Sent to Bob.

4. **Decrypt**: Bob can decrypt *C* by computing

g = e(U, $D_{\mathrm{ID}}$) and m = V ⊕ $H_2$(g)

The consistency of the scheme will discuss in next chapter, and we analyze that the value $g^r$ in encryption is similar to the e($D_{\mathrm{ID}}$, U) in decryption. This completes the BasicM-IBE scheme.

### 4.4.4 Full M-IBE scheme

To convert the BasicM-IBE scheme to FullM-IBE scheme, this is a chosen Ciphertext secure IBE system in random oracle model [58].Taking all the cryptographic goals, IBE based on our model is presents in Algorithm 4.3.

**Algorithm 4.3 Identity-based encryption free from key escrow problem**

**Aim**: Given Bob identity $\mathrm{ID_B}$ and private $D_{\mathrm{ID}}$ obtaining from algorithm 4.1, Alice wants to send an encrypted message to Bob so that no other (including PKG) than Bob can decrypt the message.

**Output**: Alice sends an encrypted Ciphertext *C* that is successfully decrypted by Bob without lose any confidentiality.

4. **Setup**: This phase is identical to setup phase of Algorithm 4.1.
5. **Key extract**: Identical to Key extract phase in algorithm 4.1.
6. **Encrypt:** Alice randomly chooses r ∈ $\mathbb{Z}_q$ and may encrypt her message

   using Bob identity ID by

   - Compute r=$H_3$(z,m) where z ∈{0, 1}*,
   - $Q_{\mathrm{ID}}$ = $H_1$(ID, $pkg_{Pub}$, $pkpo_{Pub}$)

     Moreover, g=$H_2$(e ($Q_{\mathrm{ID}}$, $pkpo_{Pub}$)).

- The resulting Ciphertext*C* = (*U*,*V*, *W*) = <rP, z $\oplus H_2(g^r)$, m $\oplus$ $H_4(z)$>
- C Sent to Bob.

3. **Decrypt**: Bob can decrypt *C* by computing

- $g' = e(U, D_{ID})$ and $z' = V \oplus H_2(g')$
- $m' = W \oplus H_4(z')$ and $r = H_4(z', m')$

Return the message m' if U= r'P.

The consistency of the scheme will discuss in next chapter, and we analyze that in decryption z'and encryption z are equals. This completes the FullM-IBE scheme.

## 4.5 Summary

This chapter summarizes the implementation of our scheme problem defined in chapter 1 (section1.1). In this chapter, first we overviewed the model of the related scheme for the solution of the key escrow problem. In the following section, we defined some cryptographic goals need to be reflected in the proposed scheme, design the model for the same. In the next section, we describe an algorithm for our proposed scheme and implement BasicM-IBE and FullM-IBE identical to BasicIden and FullIden of [7]. Full-IBE is secure against IND-CCA adversary.

Chapter

5

# Consistency, Security Proof, and Performance Analysis

In this chapter, first we verify the consistency of Algorithm 4.2 and Algorithm 4.3. Then, we proof that our FullM-IBE scheme is secure against IND-CCA1 and IND-CCA2 adversary attacks.  In the following section, we compared our scheme with existing scheme. Finally, we claim that our scheme fulfills the property as discussed in section 5.4.

## 5.1 Consistency of M-IBE scheme

Here, we proof the correctness of encryption and decryption stage of Algorithm 4.1 and Algorithm 4.2

### 5.1.1 BasicM-IBE scheme

The consistency of the scheme is verified as follows:

We know that $g = e(Q_{ID}, pkpo_{Pub})$

So $\quad g^r \rightarrow e(Q_{ID}, pkpo_{Pub})^r$

$\rightarrow e(Q_{ID}, pkpo_{Pr.}pkg_{Pr}P)^r$

$\rightarrow e(pkpo_{Pr}.\,pkg_{Pr}.Q_{ID}, P)^r$

$\rightarrow e(pkpo_{Pr}.\,pkg_{Pr}.Q_{ID}, rP)$

$\rightarrow e(D_{ID}, U)$

Here, we notice that the value $g^r$ in encryption is similar to the $e(D_{ID}, U)$ in decryption.

### 5.1.2 FullM-IBE scheme

The consistency of the protocol easily verified from

$$g' \leftarrow e(U, D_{ID})$$
$$\leftarrow e(r.P, pkpo_{Pr}.\, pkg_{Pr}.Q_{ID})$$
$$\leftarrow e(pkpo_{Pr}.\, pkg_{Pr}.P, Q_{ID})^{r}$$
$$\leftarrow e(pkpo_{Pub}, Q_{ID})^{r}$$
$$\leftarrow g^{r}$$

Therefore, in decryption z'and encryption z is equals. Consequently, applying decryption on a cipher text recovers the original message m. Moreover, our scheme achieves some influential properties that make it different from existing ID-based cryptosystem.

## 5.2 Secure against IND-CCA.

In this section, we proof that our scheme is secure against the IND-CCA type I and II attacks. Public Key encryption scheme HybridPub [46], will be used as a tool in security proof of FullM-IBE.

**Theorem 1**: Consider there are four Random Oracle hash functions $H_1$, $H_2$, $H_{3,}$ $H_{4,}$ and $H_5$. M-IBE is IND-CCA secure if there is no polynomial bounded algorithm [19] that can solve the GBDHP in groups generated by $\mathcal{G}$ with non-negligible advantage.

**Proof**: This theorem is similar to the Theorem 1 in [46]. Theorem1 can be proved by proving the number of lemmas. It can be made into a concrete security reduction relating the advantage $\epsilon$of Type I or Type II attacker against M-IBE to that of an algorithm to solve GBDHP or BDHP. Theorem 1 for Type I adversaries follows by combining Lemmas 2, 3 and 4. Similarly, Theorem 1 for Type II adversaries follows by combining Lemmas 4, 5 and6.

**Lemma 2**:Consider there are five Random Oracle hash functions $H_1$, $H_2$, $H_3$, $H_4$ and $H_5$ and there have being an adversary $\mathcal{A}_I$ of IND-CCA Type I against FullM-IBE with advantage$\epsilon$, running time t.Suppose $\mathcal{A}$constructsat most $q_i$> 0 queries for $H_i$, where, $1 \le i \le 5$ and at most $q_D$ > 0queries to decryption. Then there is an adversary$\mathcal{B}$, behave either as a Type I or a Type II IND-CPA adversary, has advantage at least $\frac{\epsilon}{4q_1q_5}\lambda^{q_D}$against HybridPub. Its running time is$t + O(q_3 + q_4)q_d t'$. Where,

$$\begin{aligned}1-\lambda \le (q_3 + q_4).\,\epsilon_{OWE}(t + O(q_3 + q_4)q_d t', q_2\,)\\ + \epsilon_{GBDHP}(t + O(q_3 + q_4)q_d t' + 3q^{-1} + 2^{-n+1})\end{aligned}$$

The advantage of any type I or Type II is at least $\epsilon_{OWE}$(T, q'), runs in time T and constructs q' queries to $H_2$, and the advantage of any algorithm to solve GBDHP is at least $\epsilon_{GBDHP}$(T). Here, t' is the running time of the BasicM-IBE encryption algorithm.

**Lemma 3**:Consider $H_3$ and $H_4$ are Random Oracles and there is a Type I and Type II IND-CPA adversary $\mathcal{A}_I$ and $\mathcal{A}_{II}$ respectively against HybridPub with advantage$\epsilon$ and construct at most $q_3$> 0queries to $H_3$and at most $q_4$> 0 queries to$H_4$. Then there is a Type I and Type II OWE adversary $\mathcal{A}'_I$ and $\mathcal{A}'_{II}$ both have an advantage at least $\frac{\epsilon}{2(q_3 + q_4)}$against BasicPub. Its running time is O (time ($\mathcal{A}_I$)) and O (time ($\mathcal{A}_{II}$)) respectively.

**Lemma 4**: Consider $H_2$ is a Random Oracle, and there is a $\mathcal{A}_I$ and $\mathcal{A}_{II}$ Type I and Type II OWE adversary has advantage$\epsilon$against BasicPub, constructs at most $q_2$ > 0 queries to $H_2$. Then there is an adversary$\mathcal{B}$ to solve the GBDHP has advantage at least $(\epsilon - \frac{1}{2^n})/q_2$. Its running time is O(time($\mathcal{A}_I$)) and O(time($\mathcal{A}_{II}$)) respectively.

**Lemma 5**: Consider $H_1$ is a Random Oracle and that there is an IND-CCA Type II adversary $\mathcal{A}_{II}$ on FullM-IBE with advantage $\epsilon$ which makes at most $q_1$ > 0 queries to $H_1$. Then there is an IND-CCA Type II adversary on HybridPub with advantage at least $\frac{\epsilon}{q_1}$which runs in time O(time($\mathcal{A}_{II}$)).

**Lemma 6**: Consider $H_3$ and $H_4$are RandomOracles and there is a Type II IND-CCA adversary$\mathcal{A}_{II}$ against HybridPub with advantage $\epsilon$ and construct at most $q_D$> 0 queries to decryption, at most $q_3 > 0$ queries to $H_3$ and at most $q_4 > 0$ queries to $H_4$. Then there is a Type II OWE adversary $\mathcal{A}'_{II}$ against BasicPub with

$$\mathrm{time}(A'_{II}) = \mathrm{time}(A_{II}) + O(n(q_3 + q_4))$$
$$\mathrm{adv}(A'_{II}) \geq \frac{1}{2(q_3 + q_4)}.((\epsilon + 1)(1 - q^{-1} - 2^{-n})^{q_D} - 1)$$

**Lemma 7**: In Lemma 2, Algorithm $\mathcal{KE}$ responds with correct to each decryption queries with advantage at least λ where

$$1 - \lambda \leq (q_3 + q_4).\epsilon_{OWE}(t + O(q_3 + q_4)q_d t', q_2) + \epsilon_{GBDHP}(t + O(q_3 + q_4)q_d t' + 3q^{-1} + 2^{-n+1})$$

Here, the advantage of any type I or Type II is at least $\epsilon_{OWE}$(T, q'), operates in time T and constructs q' queries to $H_2$, and the advantage of any algorithm to solve GBDHP is at least $\epsilon_{GBDHP}$(T). Here, t' is the running time of the BasicM-IBE encryption algorithm and t is the running time of adversary $\mathcal{A}_I$.

**Proof 2**: Let$\mathcal{A}_I$ be a Type I IND-CCA adversary againstFullM-IBEl,hasadvantage ϵ, runs in time t, construct at most makes $q_i > 0$ queries to Random Oracle $H_i(1 \leq i \leq 4)$ and a decryption query is at most $q_D$> 0.There is another adversary $\mathcal{B}$ derive from $\mathcal{A}_I$ that pretend as a Type I IND-CCA adversary or Type II IND-CCA adversary. We assume two challengers' $\mathcal{C}_I$ and $\mathcal{C}_{II}$ are available to $\mathcal{B}$ for two different challenges.

First, Adversary $\mathcal{B}$ chooses a Random bit c and an index I such that $1 \leq I \leq q_1$. $\mathcal{B}$ wishesto play with $\mathcal{C}_I$ and aborts $\mathcal{C}_{II}$ where $\mathcal{C}_I$ passes $\mathcal{B}$ with a Public Key $K_{pub}$ = <$\mathbb{G}_1$, $\mathbb{G}_2$, e, n, P, $P_0$, $P_1$, Q, $H_2$, $H_3$, $H_4$>, if c = 0. Otherwise, $\mathcal{B}$ wishes to play with $\mathcal{C}_{II}$ and aborts $\mathcal{C}_I$ where $\mathcal{C}_{II}$ passes $\mathcal{B}$ with a Public Key $K_{pub}$ along with the value $s_0$ ans $s_1$such that $P_0 = s_0.P$ and $P_1= s_0.s_1.P$. Let the event that $\mathcal{A}_I$ picks $ID_I$ such that $ID_I = ID_{ch}$ is denoted by

$\mathcal{H}$, the event that $\mathcal{A}_I$extracts the partial Private Key for user $ID_I$ be $\mathcal{F}_0$and the event that $\mathcal{A}_I$substitute the Public Key of user $ID_I$denoted as $\mathcal{F}_1$.

Here, $H_i$ is a Random Oracle that will be ruled by $\mathcal{B}$ and managed as follows:

- $H_1$ queries: $\mathcal{B}$ manages a list for $H_1$of tuples $<ID_i, Q, b_i, x_i, y_i, P_{0i}, P_{1i}, P_{2i}>$, empty in initially. When $\mathcal{A}_I$ make queries $H_1$ on input ID$\in \{0, 1\}^*$, $\mathcal{B}$ outputs as follows:

  a) $\mathcal{B}$ outputs$H_1(ID) = Q_i \in G_1^*$ if ID found in the $H_1$ list,

  b) $\mathcal{B}$ Randomly chooses $b_I$from $Z_q$, gives$H(ID) = b_{I.}Q$ and adds the tuple $<ID, b_{I.}Q, b_I, \perp, \perp, P_0, P_1, P_2>$ to the $H_1$ list, if ID does not find in the $H_1$ list such that ID is I-th distinct $H_1$ query made by $A_{I.}$

  c) Otherwise, when ID does not exist in the list and ID is the i-th distinct $H_1$ query made by $\mathcal{A}_I$where $i \neq I$, $\mathcal{B}$ randomly chooses $b_i, x_i$ and $y_i$ from $Z_q$, gives$H(ID) = b_{i.}P$ and adds $<ID, b_{i.}P, b_i, x_i, y_i, x_{i.}P, x_{i.}P_0, y_{i.}P>$ to the $H_1$ list if ID does not find in list where ID is the i-th $H_1$ query such that $I \neq i$.

  After the definition of $H_1$, the FullM-IBE partial Private Key for $ID_i = b_{i.}P_0$ such that $i \neq I$. Thus,the Public Key is $<x_{i.}P, x_{i.}P_0>$ and the Private Key is $<x_{i.}b_{i.}P_0>$ for $ID_i$ when c = 0. Otherwise, $\mathcal{B}$ can compute $s_{0.}b_{I.}Q$, the partial Private Key of $ID_I$.

- $H_2$ and $H_4$ queries: $\mathcal{A}_I$makes $H_2$ queries and passed to challenger $\mathcal{C}$. Adversary$\mathcal{A}_I$ makes $H_4$ query and $\mathcal{B}$ passes this query to $\mathcal{C}$, indeed, keeps list $<\sigma'_i, H_{4,i}>$ and $\mathcal{C}$'s answer to them.

- $H_3$ queries:

  1) Adversary $\mathcal{A}_I$ makes $H_4$ query and $\mathcal{B}$ passes this query to $\mathcal{C}$, indeed, keeps list $<\sigma'_{i.}M_j, H_{3,j}>$ and $\mathcal{C}$'s answer to them.

  2) $\mathcal{B}$ manages a list of tuples $< Q, b_i, x_i, y_i, N_{1i}, N_{2i}>$ which is initially empty.

- $H_5$ queries:

  Let $Z_i= e(x_{i.}y_{i.}P, -x_{i.}b_{i.}P)$ When $\mathcal{A}_I$ queries$H_5$, $\mathcal{B}$responds as follows:

a) If Z find in the $H_5$ list in tuple $<Z_i, Q_i, b_i, x_i, y_i, N_{1i}, N_{2i}>$, then $\mathcal{B}$responds with $H_5(Z) = Q_i \in G_1^*$.

b) If Z does not find in the list and Z is the I-th distinct $H_1$ query made by $\mathcal{A}_I$, then $\mathcal{B}$ picks $b_I$at Random from $Z_q$, outputs $H(Z) = b_{I.}Q$ and adds the entry $<ID, b_{I.}Q, b_I, \perp, \perp, N_1, N_2>$to the $H_1$ list.

c) Otherwise$\mathcal{B}$ picks $b_I$, $x_I$ and $y_I$ at random from $Z_q$, output $N_1= x_{I.}b_{I.}Q$ and $N_2= x_{I.}y_{I.}P$ and adds the entry $<ID, b_{I.}Q, b_I, \perp, \perp, N_1, N_2>$ to the $H_5$ list.

Phase 1: $\mathcal{A}_I$ receives *params* from $\mathcal{B}$ and makes a number of requests for a user, including a partial Private Key extraction, a Private Key extraction, a request for a Public Key, substitution a Public Key or a decryption query.$\mathcal{B}$ replies to these requests as follows:

- Partial Private Key Extraction: Let $\mathcal{A}_I$ make the request on $ID_i$. One of the three case will occur:
    - $\mathcal{B}$ outputs with $b_iP_0$, if $i \neq I$.
    - $\mathcal{B}$ aborts, if i = I and c = 0.
    - $\mathcal{B}$ outputs with $s.b_{I.}Q$, if i = I and c = 1.
- Private Key Extraction: Let $\mathcal{A}_I$ make the request on $ID_i$. Let the Public Key for $ID_i$ has not been the substitute. One of the two case will occur:
    - $\mathcal{B}$respondsx$_{i.}b_{i.}P_0$, if $i \neq I$.
    - Otherwise $\mathcal{B}$ aborts.
- Request for Public Key: Let $\mathcal{A}_I$ make the request on $ID_i$.$\mathcal{B}$ returns $<x_{i.}P, x_{i.}P_0>$ by obtaining the $H_1$ list.
- Replace Public Key: Let $\mathcal{A}_I$ make the request on $ID_i$ to substitute the Public Key with value $<P'_{0i}, P'_{1i}>$. One of the two case will occur:
    - $\mathcal{B}$ aborts, if i = I and c = 1.

- Otherwise, the existing entries $P_{0i}$ and $P_{1i}$in the $H_1$ listis substituted with the new entries <$P'_{0i}$, $P'_{1i}$> and $\mathcal{B}$ makes a request $\mathcal{C}$ to substitute the Public Key <$P'_0$, $P'_1$> in$K_{pub}$ with new values <$P'_{0i}$, $P'_{1i}$>, if i = I.

- Decryption Queries: Let $\mathcal{A}_I$ make request to decrypt the Ciphertext < U, V, W > for $ID_l$, As $\mathcal{B}$ is pretending as an IND-CPA adversary, so he will not use the challenger $\mathcal{C}$ to reply the query ( I = l ).Alternatively, for existing Public Key < $P_{0l}$, $P_{1l}$ > of $ID_i$ and a cipher text C = <U, V, W>, $\mathcal{B}$ responds to each decryption query with advantage at least λ, where λ is proved in Lemma 7. $\mathcal{B}$ responds to each decryption queries as follows:

  i. $\mathcal{B}$ search tuple < $\sigma_j$, $M_j$ $H_{3,j}$> on the $H_3$ list. Accumulate these tuples in a list $S_1$. If $S_1$ is empty, output ⊥and halt.

  ii. $\mathcal{B}$ search every pairs < $\sigma'_i$, $H_{4,i}$> in the $H_4$ list, for every tuple < $\sigma_j$, $M_j$ $H_{3,j}$>in S1. If $\sigma_j = \sigma'_I$, add tuple < $\sigma_j$, $M_j$, $H_{3,j}$, $H_{4,i}$> in$S_2$ list. If $S_2$ is empty, then output ⊥ and halt.

  iii. For W = $M_j \oplus H_{4,i.}$, $\mathcal{B}$ find in $S_2$ for such an entry. Responds $M_j$ as the result of <U, V, W>, if exists. Otherwise, responds ⊥.

Challenge Phase: $\mathcal{A}$ responds the challenge identity $ID_{ch}$ and two equal length message$m_0$, $m_1 \in$ M. Let $\mathcal{A}_I$ do not allow extracting the Private Key for identity $ID_{ch}$. Algorithm $\mathcal{B}$responds as follows:

- $\mathcal{B}$ aborts, if $ID_{ch} \neq ID_I$.

- Otherwise $ID_{ch}$ = $ID_I$ and $\mathcal{B}$gives $\mathcal{C}$the pair $m_0$, $m_1$ as the messages on which it wishes to be challenged.

$\mathcal{C}$ outputs C'= <U', V', W'>, such that C' is the HybridPub encryption of $m_b$. Then $\mathcal{B}$sets C*= <$b_I^{-1}$.U', V', W'> and passes C*to $\mathcal{A}_I$ such that C*is the FullM-IBE encryption of

$m_b$ with Public Key <$P_{0I}$, $P_{I1}$>. Let <$P_{0ch}$, $P_{1ch}$> be the Public Key for identity $ID_{ch}$ during the challenge phase.

Phase 2: Similar to phase 1, $\mathcal{B}$ repeatedly reply to $\mathcal{A}_I$'s requests.

Guess: In the end, $\mathcal{A}_I$ may form a guess b’ for b. $\mathcal{B}$ respondsb’as a guess for b. If $\mathcal{A}_I$ take more time than time t, or take the large number of attempts to make $q_i$ queries or $q_D$ decryption queries, then $\mathcal{B}$ should abort $\mathcal{A}_I$ and output a Random guess for bit b.

Analysis: During the whole execution process, if $\mathcal{B}$ does not abort and the decryption queries respond by $\mathcal{B}$ is uses correctly, then algorithm $\mathcal{A}_I$ is considered to be a real attack. Furthermore, the encryption of $m_b$ is the challenge Ciphertext C* under the Public Key of $ID_{ch}$, such that $b \in \{0,1\}$is Random. So according to the definition of an adversary $\mathcal{A}_I$ we have that $2(Pr[b = b'] = 1/2\ ) \geq \epsilon$.

Now we have to inspect that during the execution process the probability that $\mathcal{B}$ does not halt. Inspecting the execution process, we realize that $\mathcal{B}$ can abort for one of following reasons:

- When c = 0 and the event $\mathcal{F}_0$ happens, denoted as $\mathcal{H}_0$
- When c = 1 and the event $\mathcal{F}_1$ happens, denoted as $\mathcal{H}_1$
- Because $\mathcal{A}_I$ made a Private Key extraction on $ID_I$ at some point, denoted as $\mathcal{F}_2$
- When $\mathcal{A}_I$ chose $ID_{ch} \neq ID_I$, denoted as $\mathcal{F}_3$
- Or $\mathcal{A}_I$ chose $Z_{ch} \neq Z_I$, denoted as $\mathcal{F}_4$

Theprobability that $ID_{ch} = ID_I$ is equal to $1/q_1$ because $\mathcal{A}_I$ construct $q_1$ queries of $H_1$and $\mathcal{B}$ havea choice to choose I from the set of $q_1$ query. Hence Pr[$\mathcal{H}$] =Pr[$\mathcal{F}_3$] = $1/q_1$. Similarly, $\mathcal{A}_I$construct$q_5$ queries of $H_5$ and B have a choice to choose I from the set of the$q_5$ query. Thus, the probability that $Z_{ch} = Z_I$ is equal to $1/q_5$. Hence Pr[$\mathcal{F}_4$] = $1/q_5$. As we know if $\mathcal{A}_I$choose $ID_{ch} = ID_I$, then no Private Key extraction on $ID_I$ will bepermitted.

From all this information:

$$\Pr[\mathcal{B}\text{does not abort}] = \Pr[\neg\mathcal{H}_0 \wedge \neg\mathcal{H}_1 \wedge \neg\mathcal{F}_2 \wedge \neg\mathcal{F}_3 \wedge \neg\mathcal{F}_4] = \frac{1}{q_1}\frac{1}{q_5}\Pr[\mathcal{H}_0 \wedge \mathcal{H}_1 \mid \mathcal{H}]$$

Because, two events $H_0$ and $H_1$ are mutually exclusive. So, we can write

$$\Pr[\mathcal{H}_0 \wedge \mathcal{H}_1|\mathcal{H}] = 1- \Pr[\mathcal{H}_0|\mathcal{H}] - \Pr[\mathcal{H}_i|\mathcal{H}]$$

And because the event $\mathcal{F}_i|\mathcal{H}$is independent of the event c = I

$$\Pr[\mathcal{H}_i|\mathcal{H}]= \Pr[(c = i) \wedge \mathcal{F}_i|\mathcal{H}] =\frac{1}{2}\Pr[\mathcal{F}_i|\mathcal{H}]$$

Now, we have

$$\Pr[\mathcal{B}\text{does not abort}] =\frac{1}{q_1 q_5}(1- \frac{1}{2}\Pr[\mathcal{F}_0|\mathcal{H}] - \frac{1}{2}\Pr[\mathcal{F}_i|\mathcal{H}])$$

According to the rules subjected to adversary described in security model, an adversary cannot allow to extract the partial Private Key as well as the substitute the Public Key. So, we have $\Pr[\mathcal{F}_0 \wedge \mathcal{F}_1|\mathcal{H}] = 0$.This implies that $\Pr[\mathcal{F}_0|\mathcal{H}]+\Pr[\mathcal{F}_1|\mathcal{H}] \leq 1$. Hence, we realize that Pr[$\mathcal{B}$does not abort] $\geq \frac{1}{2q_1 q_5}$

Finally, now we examine the probability that $\mathcal{B}$ in decryption query phase correctly controls all $q_D$ decryption queries of $\mathcal{A}_I$'s. Thus, $\mathcal{B}$'s advantage is at least $\frac{\epsilon}{2q_1 q_5}\lambda^{q_D}$. It follows that either $\mathcal{B}$'s advantage as a Type I adversary or $\mathcal{B}$'s advantage as a Type II adversary $\frac{\epsilon}{4q_1 q_5}\lambda^{q_D}$. This completes the proof.

**Proof of Lemma 3**: This Proof is similar to the Proof of Lemma 10 of [53], with modify with an addition of query $q_5$.

**Proof of Lemma 4**: This proof is identical to the proof of Theorem 4.1 in [7], with modification with an addition of query $q_5$for type I adversary and Type II adversary.

**Proof of Lemma 6**: This lemma can be proven through theorem 14 of [53], assuming that $msk_{Pr}$ can be made available to Type II adversaries.

## 5.3 Performance

In chapter 4, we proposed M-IBE scheme (a variant of CL-PKC). Indeed, our approach is the modification in the key generation process of Boneh-Franklin scheme [7].Table 5.1 shows the comparisons of our scheme with existing scheme on the basis of computation cost. According to the level to trust to PKG in Girault [22],Table 5.2 shows the comparisons of our scheme with different Public Key encryption schemes. Here, we conclude that our scheme achieve Level 3.

| Scheme | Key Generation | Encryption | Decryption |
|---|---|---|---|
| Threshold key issue | 2nM | 1M+1P+1E | 1M+1P |
| CB-PKC | 4M | 2M+1P+1E | 1P |
| CL-PKC | 4M | 1M+1P+1E | 1M+1P |
| HIBE | 2tM | tM+1P+1E | tM +tP |
| VIBE | 4M | 1M+3P+1E | 2M+1P |
| M-IBE | 6M+1P+1E | 1M+1P+1E | 1M+1P |

Table 5. 1 Comparison of Computation Cost of proposed scheme with existing scheme, where, M: Point multiplication, P: No of pairing operation, E: Exponentiation, t: no of user's identities in HIBE from root to leaf and n: number of PKGs in threshold issue scheme

| Scheme | Public information | Trust level |
|---|---|---|
| Threshold key issue | ID | Level 1 |
| CB-PKC | ID, $usk_{Pub}$ | Level 3 |
| CL-PKC | ID, $usk_{Pub}$, C | Level 2 |
| HIBE | $ID_1$,..$ID_i$ | Level 1 |
| VIBE | ID, $N_1$. $N_2$ | Level 3 |
| M-IBE | ID | Level 3 |

Table 5. 2 Comparison of public information and trust level to PKG of proposed scheme with existing scheme, where, $N_1 = \boldsymbol{usk_{Pr}}$P: point multiplication of user secret $\boldsymbol{usk_{Pr}}$ with group

generator P, $N_2 = \boldsymbol{usk_{Pr}}.\boldsymbol{msk_{Pub}}$: point multiplication of user secret $\boldsymbol{usk_{Pr}}$ with PKG's Public Key and C: commitment of user secret key with the PKG's Public Key.

Note that, other than our scheme each scheme needs at most 4 hash functions, but our scheme needs 1 more extra hash functions. However, the computational time of hash function is not the big issue as it is very fast as compared to the pairing computation or scalar operation. So, we ignore the computational time of the hash operation and point addition because the number of hash operations are almost equal in all schemes and point addition is lightweight as compared to other heavy operation. On the basis of table 5.1 and 5.2, we compared our scheme with existing scheme as follows:

### 5.3.1 Comparison with Boneh-Franklin

By including a Non-government organization (PKPO) between the user and PKG, we modify the Boneh-Franklin's IBE scheme to remove the Key escrow Problem. PKPO provides key privacy to the user. The computation cost of encryption and decryption algorithm in our scheme is equal to the Boneh-Franklin scheme [7]. While in terms of Private Key issuing, our scheme needs 2 extra point multiplications, 1 pairing operation, and 1 exponentiation. Thus, for key issuing our scheme is slightly slower. As compared to the solution of Key escrow problem solved by the Boneh-Franklin, our scheme is very fast to issuing the Private Key as shown in Table 5.1. On comparing with B-F approach, our scheme achieves level 3 of trust level as shown in Table 5.2.

### 5.3.2 Comparison with CL-PKC

As compared to CL-PKC, our scheme is slightly slower in the key issuing process but takes the equal number of computation time in encryption and decryption process. Additionally, in our scheme user chosen secret information is used for only secure extracting the Private Key. Therefore, our scheme provides implicit as well as explicit authentication as compared with CL-PKC, which provides only implicit authentication. Moreover, our scheme achieve level 3, on the other hand, CL-PKC achieve level 2 of trust level to PKG.from Table 5.2, we conclude that our scheme have minimum number of 1 public information share on the communication network.While CL-PKC takethree public information.

### 5.3.3 Compared with other scheme

On comparing with CB-PKC, our scheme preserves the advantage of ID-based cryptography and has one number of public information on the network. While CB-PKC have two numbers public information as shown in Table 5.2. Both schemes achieve the third level of trust on PKG. On the computational point of view, our scheme is faster on encryption and decryption process. While our scheme is slightly slower as comparison to another scheme in the key issuing process.

On comparing with VIBE, as shown in Table 5.1 and 5.2, our scheme is performed well as faster in encryption algorithm and has a minimum number of public information shared on the network.

Here, we analyze that our scheme is most advantageous variant of generic IBE scheme that remove the inherited key escrow problem with following advantage:

- Minimum of public information shared on the network
- Achieve trust level 3 on PKG
- Preserve the advantage of the ID-based cryptosystem.

Moreover, with the minimum computational cost of encryption and decryption, our scheme will more efficient for low power consumption devices, where key issuing overload is dominating to the server side.

## 5.4 Claim

CLAIM 1: **No Key Escrow Problem.** *An adversary cannot decrypt the encrypted message without the knowledge of PKPO's Private Key*$\mathrm{pkpo}_{\mathrm{pr}}$*; even he/she know the master key*$\mathrm{pkg}_{\mathrm{pr}}$.

This claim follows fromTheorem1 in Section 5.2.

CLAIM 2: **Partial key escrow**. *An adversary cannot decrypt the encrypted message without the knowledge of PKPO's Private Key*$\mathrm{pkpo}_{\mathrm{pr}}$*; even he/she know the master key*$\mathrm{pkg}_{\mathrm{pr}}$.

This claim also follows from Theorem 1 in Section 5.2.

CLAIM3: **Preserve ID-based cryptosystem**. *As compared to certificate-based encryption where a certificate is required to obtain the Public Key, user's identity ID is used to generate the Public Key.*

CLAIM 4: **Key recovery.** *In case of compromising the Private Key by the PKG, user along with the PKPO may claim that the Private Key $D_{ID}$ has been compromised by PKG and file a legal case against the PKG in court. Consequently, PKG will be present in the Court for their punishment. In case of the malicious user, PKPO may claim on victim request that the user may be malicious and acting as a mischievous activity. Consequently, all three entities (PKG, PKPO, and user) will present in the court on- a legal court order and release their master key to derive the user Private Key $D_{ID.}$ Thus, the judge may recover the Private Key and decrypt the malicious message.*

## 5.5 Summary

In this chapter, we verify the consistency of encryption and decryption of algorithm 4.2 and algorithm 4.3. Then, we proved that our scheme is secure against IND-CCA1 of type I and Type II attacks in Theorem1 which proved with the help of Lemma 2 to Lemma 7. In following section, we compared our scheme with HIBE, VIBE, threshold key issuing, CL-PKC, and CB-PKC. Tothe comparison, we conclude that according to table 5.1 and table 5.2, our scheme is more advantageous than the other scheme. Finally, we claim that our scheme how generic IBE transformed to the fair ID-based cryptosystem that preserve the objective in chapter 1 (section 1.4)

Chapter 6

# Conclusion and Future Work

This final chapter comes to an end by presenting a general review of the discussed topics along with the final result. Additionally, the shortcoming of our solution, possible future scope and other domains in which our scheme is fruitful are summarized. The Implementation of the proposed IBE approach fulfills the fair balance between the user's rights to privacy and the government's rights to monitor the unlawful message. With the definition of a security model describing all considered entities in the model. We show that proposed model is secure against an IND-CCA of Type I and Type II adversary attacks with their proofs. The defined model, then served as a framework to state cryptographic design goals that were achieved by relying on earlier specified cryptographic building blocks.

The proposed model is slightly slower than the previous solution scheme as it needs extra pairing computation and exponential operation during the key issuing algorithm. We know that the key issuing process is an algorithm, executed once on the server side while user joins the network. Therefore, it is not the big issue to solve. One more limitation is that it assumed that PKG and PKPO never collude each other. If they to do so, user's Private Key may be compromised. However, on the doubt of unlawful message, the guanine entity from the three entities (PKG, PKPO, and User) may file a case against the malicious entity. Finally, the judge may catch a suspected entity out of the three.

The resulting infrastructure modeled for low computation on the client side and overload shifted to the server side (PKG) on the cloud, which is environment-friendly, practically applicable and instantly prepared to use.

*Instantly ready to use*: In contrast to previous solutions, present infrastructure is instantly ready to use. Once the user joins the network and issuing their Private Key, they are no longer required to encrypt and decrypt a message with the minimum computation cost with the lowest number of public information sharing on the network.

*Environment-friendly*: As compared to the previous approach, our scheme is the one who claim to environment-friendly.Moreover, Since it fulfills the fair balance between the user's rights to privacy and the government's rights to monitor the unlawful message.

*Practically applicable*: With the increasing influence of application on wireless devices in our daily life, some applications are even expected to increase day by day. In the proposed scheme, the maximum computation operation is shifted to the server side with the minimum number of computations on the client side. Therefore, the present solution is applicable for encryption and decryption in low computation devices like mobile, laptop and tablets on the customer side.

We conclude this work by emphasizing the applicability of our current scheme to other domains than encryption and decryption on wireless devices. The proposed scheme can also apply to the signature and authenticated key agreement protocol. Consequently, the proposed scheme is also valuable for e-mail applications also.

# References

[1] Ben Lynn, “On the implementation of pairing-based cryptography”, Master’s thesis, Stanford,2007.

[2] B. A. Forouzan, “Cryptography & Network Security”, McGraw-Hill, Inc., 2007.

[3] Z. Cao and F. Zhang, editors. “Pairing-Based Cryptography” - Pairing 2013 *-6th International Conference, Beijing, China*, November 22-24, 2013, RevisedSelected Papers, volume 8365 of Lecture Notes in Computer Science. Springer,2014.

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

[5] A. Shamir, “Identity-based cryptosystem and signature scheme”, *proc. Crypto 84*, pp 47-53, 1984.

[6] A. Enge, "Bilinear pairings on elliptic curves," arXiv preprintarXiv:1301.5520, 2013.

[7] D.BonehandM.K.Franklin,“IdentitybasedencryptionfromtheWeilpairing”,*IACRCryptology* ePrintArchive,2001:90,2001.

[8] Elliptic curve of equation $y^2 = x^3 - x + 1$, http://arstechnica.com/security/2013/10/a-relatively-easy-to-understand-primer-on-elliptic-curve-cryptography/

[9] X.Boyen and B.Waters, “Anonymous hierarchical identity-based encryption (without random oracles)”, *In C.D work, editor, CRYPTO*, volume 4117 of Lecture Notes in Computer Science, pp 290–307, Springer, 2006.

[10] A. Boldyreva, V. Goyal, and V. Kumar, “Identity-based encryption with efficient revocation”, *IACR Cryptology* ePrint Archive, 2012:52, 2012.

[11] D. Boneh, “The decision Diffie-Hellman problem”, *In J. Buhler, editor, ANTS*, volume 1423 of Lecture Notes in Computer Science, pages 48–63. Springer, 1998.

[12] D. Boneh, X. Boyen, and E.-J. Goh, “Hierarchical identity based encryption with constant size ciphertext”, *In R. Cramer, editor, EUROCRYPT*, volume 3494 of Lecture Notes in Computer Science, pages 440-456. Springer, 2005.

[13] Nan Li (2010), “Research on Diffie – Hellman Key Exchange Protocol”, *IEEE 2nd International Conference on Computer Engineering and Technology*, Vol. No 4, pp 634 – 637, 2003.

[14] J. Menezes, P. C. van Oorshot and S. A Vanstone, “Handbook of Applied Cryptography”, *CRC Press*, New York, USA, 1997

[15] Kenneth G. Paterson and Geraint Price, “A comparison between traditional Public Key Infrastructures and Identity-Based Cryptography”, *Information Security Technical Report*, Vol. 8, No. 3, pp 57-72, 2003.

[16] R.L. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”. *Communications of the A.C.M.*, Vol. No 21, issue No 2, pp 120-126, 1978.

[17] Rational points on line over elliptic curve, http://www.cs.mcgill.ca/~rwest/link-suggestion/wpcd_2008-09_augmented/wp/e/Elliptic_curve.htm

[18] U. Maurer and Y. Yacobi, “Non-interective public-key cryptography”, *Proc. Of Eurocrypto ’91, Lecture Nores in Computer Sciences*, Springer-Verlag, Vol. No 547, pp 498-507, 1992.

[19] P. Shor, “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”, *SICOMP*, Vol. No 26, Issue 5, pp 1484–1509, 1997.

[20] H. Abelson, R. Anderson, S. M. Bellovin, J. Benalob, M. Blaze, W. Diffie, J. Gilmore, P. G. Neumann, R. L. Rivest, J. I. Schiller, and B. Schneider, “The risks of key recovery, key escrow, and trusted third-party encryption”, World Wide Web J., vol. 2, No. 3, pp 241–257, 1997.

[21] Y. Desmedt and J. Quisquater, “Public-key Systems based on the Difficulty of Tampering“, *Proc. of Crypto ’86*, Springer-Verlag, Lecture Notes in Computer Sciences, Vol. No 263, pp 111-117, 1987.

[22] M. Girault. “Self-certified public keys”, *In EUROCRYPT ’91*, Vol. 547, LNCS, pp. 490 – 497, Springer, 1991.

[23] H. Tanaka, “A realization scheme for the identity-based cryptosystem”, *Proc. of Crypto ’87*, Springer-Verlag, Lecture Notes in Computer Sciences, Vol. No 293, pp 341-349.

[24] R. Sakai, K. Ohgishi, and M. Kasahara, “Cryptosystems based on pairing”, *Proc. of SCIS ’00*, Okinawa, Japan, Jan. pp 26-28, 2001.

[25] D. Boneh, X. Ding, G. Tsudik, and C.-M. Wong,“A method for fast revocation of public key certificates and security capabilities”, In D. S. Wallach, editor, USENIX Security Symposium. USENIX, 2001.

[26] Harn, L., and Lin, H.-Y., “An authenticated key agreement without using one-way hash functions”. *Proc. 8th Nat. Conf. on Information Security,* Kaohsiung, Taiwan, pp 155–160, 1998.

[27] David Gray, Caroline Sheedy, “E-Voting: A New Approach Using Double-Blind identity-Based Encryption”, *Public Key Infrastructures, Services and Applications*, DOI: 10.1007/978-3-642-22633-5_7, pp 93-108, 2011.

[28] J. Cha and J. Cheon, ”An identity-based signature from gap Diffie- Hellman groups”, *In: Proc. PKC’2003,* Lecture Notes in Computer Science, vol. 2567, pp 18-30, 2003.

[29] L. Guillou and J. Quisquater, “A paradoxical indentity-based signature scheme resulting from zero knowledge”, *In: Proc. CRYPTO’88*, Lecture Notes in Computer Science, vol. 403, pp 216-231, 1990.

[30] FEIGE, U., FIAT, A. and SHAMIR, “A.: Zero knowledge proofs of identity”, *Proceedings of STOC* 1987, pp. 210–217, 1987.

[31] Cheng Z H, Comley R, Vasiu L. “Remove Key Escrow from the Identity-Based Encryption System”, *Foundations of Information Technology in the Era of Network and Mobile Computing*, Paris, France, August, 2004.

[32] Yang Bo, Ma Wenping, Wang Yumin, “A new secret sharing threshold scheme and key escrow system”, *Acta Electronic sinica*, vol. 26, No. 10, 1998.

[33] J. Horwitz and B. Lynn, “Toward Hierarchical Identity-Based Encryption”, *Proceedings of EUROCRYPT 2002*, LNCS 2332, pages 466-481, Springer-Verlag 2002.

[34] C. Cocks, “An Identity-based Encryption Scheme Based on Quadratic Residues”, *Cryptography and Coding - Institute of Mathematics and Its Applications International Conference on Cryptography and Coding – Proceedings of IMA 2001*, LNCS 2260, pages 360–363, Springer-Verlag 2001.

[35] Taher ElGamal, “*A public key cryptosystem and a signature scheme based on discrete logarithms”, IEEE Trans. Inform. Theory,* Vol. **31**, No. 4, pp 469–472, 1985.

[36] J. Baek, J. Newmarch, R. Safavi-naini, and W. Susilo. “A survey of identity-based cryptography”, *In Proc. of Australian Unix Users Group Annual Conference*, pp 95–102, 2004.

[37] C. Gentry, “Practical identity-based encryption without random oracles”, *In S. Vaudenay, editor, EUROCRYPT*, volume 4004 of Lecture Notes in Computer Science, pp 445–464.Springer,2006.

[38] M. Kumar, C. P. Katti and P. C. Saxena, “An ID-based Authenticated Key Exchange Protocol”, *International Journal of Advance study in computer science and engg.*, Vol. 4, No. 5 pp 11-25, 2015

[39] C. Gentry and A. Silverberg, “Hierarchal ID-based Cryptography”, *proceeding of Asiacrypt 20023,* LNCS 2501 pp. 548-566, 2002.

[40] A. Shamir, “A Partial Key escrow: a new approach to key escrow conference”, *Private communication made at Crypto 95,* August 1995.

[41] K. Peterson, “ID-Based signature from pairings on elliptic curve”, *Electronic Letter*. Vol 38, No. 18, pp 1025-1026, 2002.

[42] B. Lynn, “Authenticated Identity-based encryption”, *cryptology ePrint archive,* Report 2002/072, 2002.

[43] J. Malone-Lee, “Identity-based Signcryption”, *cryptology ePrint Archive,* Report 2002/072, 2002.

[44] Joseph K. et.al, “Certificate-Based Signature Schemes without Pairings or Random OraCL-PKCs”, *11th Information Security Conference (ISC ‟08)* Springer Verlag, 2008.

[45] C. Gentry. “Certificate-based encryption and certificate revocation problem”, *In EUROCRYPT 2003,* Vol. 2656, LNCS, pp. 272 – 293, Springer, 2003.

[46] S. S. Al-Riyami and K.G. Paterson, “Certificate-less public key cryptography”. *In ASIACRYPT 2003,* Vol. 2894, LNCS, pp. 452-473, Springer 2003.

[47] Sherman S.M. Chow, “Removing Escrow from Identity-based Encryption”, *In Public Key Cryptography*, Vol 5443 of LNCS, pp 256-276.Springer, 2009.

[48] Yuh-Min Tseng, et al., (2007), “A mutual authentication and key exchange scheme from bilinear pairings for low power computing devices”, *IEEE, Computer Software and Applications Conference, COMPSAC 2007,* Vol. No. 2, pp 700-710, 2007.

[49] Ran Canetti, Shai Halevi , Jonathan Katz, “A forward-secure public-key encryption scheme”, *Proceedings of the 22nd international conference on Theory and applications of cryptographic techniques*, Warsaw, Poland. May 04-08, 2003.

[50] P. Gammell, “An introduction to threshold cryptography*”, in cryptoBytes, a technical newsletter of RSA Laboratories*, Vol. 2, No.7, 1997.

[51] A proposed federal information processing Standard for an Escrowed Encryption standard (EES), Federal registration, July 30, 1993.

[52] N. Smart, “An identity-based Authenticated Key Agreement Protocol based on weil pairing”, *Electronic Letter*. Vol 38, pp 630-632, 2002.

[53] E. Fujisaki and T. Okamoto, “Secure integration of asymmetric and symmetric encryption scheme”, *In M. J. Wiener, editor, Proc. CRYPTO 1999,* LNCS vol. 1666, pp537-554 Springer 1999.

[54] A. Shamir, “How to Share a Secret”, *Communications of the ACM*, Vol. 22, pp 612-613, 1979.

[55] N. Koblitz, “A course in number theory and cryptography”, vol. 114. Springer, 1994.

[56] V. Miller, “Short programs for functions on curves," *Unpublished manuscript*, vol. 97, pp. 101-102, 1986.

[57] B. Preneel, “Cryptographic hash functions," *European Transactions on Telecommunications*, vol. 5, no. 4, pp. 431-448, 1994.

[58] M. Bellare and P. Rogaway, “Random Oracles are Practical: A Paradigm for Designing Efficient Protocols”, *1st Conference on Computer and Communications Security*, ACM, 1993, pp. 62–73

[59] R. Sakai and M. Kasahara. “ID-based cryptosystems with pairing on elliptic curve”. *IACR Cryptology ePrint Archive*, 2003:54, 2003.

[60] M. Cooper, Y. Dzambasow, P. Hesse, S. Joseph, and R. Nicholas. Internet X.509 “Public Key Infrastructure: Certification Path Building”, *RFC 4158* (Informational), Sept. 2005.

[61] D. Boneh and X. Boyen. “Efficient selective-id secure identity-based encryption without random oracles”,*In C. Cachin and J. Camenisch*, editors, EUROCRYPT, volume 3027 of Lecture Notes in Computer Science, pages 223–238. Springer, 2004.

[62] Silvio Micali, Fair Public-Key Cryptosystems, Proceedings of the 12th Annual International Cryptology Conference on Advances in Cryptology, p.113-138, August 16-20, 1992

[63] S. Micali and A. Shamir. Partial key escrow. Manuscript, February 1996.

[64] Cao Z F. A threshold key escrow scheme based on public key cryptosystem. Sci China Ser E-Tech Sci, 2001, 44(4): 441–448.

[65] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz, "Comparing Elliptic Curve Cryptography and RSA on 8-bit CPUs." Boston, Massachusetts: 6th International Workshop on Cryptographic Hardware and Embedded Systems, August 2004.