CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Implementation of Identity Based Broadcast Encryption Scheme using Weil Pairing

by

Saba Majeed

A thesis submitted in partial fulfillment for the degree of Master of Philosophy

in the

Faculty of Computing

Department of Mathematics

2018

To my mother on her ongoing support and love and my late father.

CAPITAL UNIVERSITY OF SCIENCE & TECHNOLOGY
ISLAMABAD

CERTIFICATE OF APPROVAL

Implementation of Weil pairing on Identity Based Broadcast Encryption Scheme

by

Saba Majeed

MMT153019

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Tayyab Kamran	QAU, Islamabad
(b)	Internal Examiner	Dr. Dur-e-Shawar Sagheer	CUST, Islamabad
(c)	Supervisor	Dr. Rashid Ali	CUST, Islamabad

Dr. Rashid Ali
Thesis Supervisor
April, 2018

Dr. Muhammad Sagheer	Dr. Muhammad Abdul Qadir
Head	Dean
Dept. of Mathematics	Faculty of Computing
April, 2018	April, 2018

Author's Declaration

I, **Saba Majeed** hereby state that my MPhil thesis titled "**Implementation of Identity Based Broadcast Encryption Scheme using Weil Pairing**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MPhil Degree.

(Saba Majeed)

Registration No: MMT-153019

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "*Implementation of Identity Based Broadcast Encryption Scheme using Weil Pairing*" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been dully acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MPhil Degree, the University reserves the right to withdraw/revoke my MPhil degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

(Saba Majeed)

Registration No: MMT-153019

Acknowledgements

First of all, I would like to thank **Almighty Allah** for His countless blessings in my life. He has gifted me a loving family and excellent teachers. He supports me in every path of life.

I would like to express my special thanks to my kind supervisor **Dr. Rashid Ali** for his motivation. He encourages me during my research study. His kind effort and motivation would be never forgotten. I have appreciated the guidance for my supervisor and feeling proud to be a student of such great teacher.

Secondly, I am thankful to all teachers of CUST Islamabad Dr. Muhammad Sagheer, Dr. Abdul Rehman Kashif, Dr. Shafqat Hussain, Dr. M.Afzal and Dr. Rashid Ali for conveying the excellent lectures.

I am grateful to my mother for her prayers, love and motivation. I would like to thank my brothers Anjum Majeed, Amjad Majeed and Asif Majeed for their support in completing my degree program. They supported and encouraged me throughout my life. I would like to thank my all family members for their continuous support and patience during my research work.

I would like to thank my all friends Urwa Aftab, Fatima Ishfaq, Shahana Rizvi, and Nataila Khan for supporting me during degree programs. Especially, I would like to thanks Urwa Aftab for motivating me during research work.

Finally, I am grateful to the management staff of Capital University of Science and Technology, Islamabad for providing a friendly environment for studies.

I am obliged to all people who share their knowledge with me and support me.

“Faith is taking the first step even when you don’t see the whole staircase”

Martin Luther King, Jr.

Abstract

Identity Based Broadcast Encryption Scheme (IBBE) allow the center to transmit data over broadcast channel to the large number of users such that only selected subset of privileged user can decrypt the information. Center encrypt the message by using identity of user so the only privileged users can decrypt it. In this thesis, we review the IBBE scheme introduced by Ming and Wang. They proposed IBBE with group of prime order. Their construction is based on bilinear mapping. Also, they use dual pairing vector space technique in prime order groups. They achieve constant size private key, ciphertext and system parameters. We mainly focused on the implementation of Ming and Wang's Scheme using group of points of elliptic curve. Due to small key size, elliptic curve cryptography (ECC) has gained considerable importance in recent years. Another reason for using ECC is the weil pairing which is considered to be good candidate of bilinear mapping. Using a suitable elliptic curve and weil pairing we constructed two toy examples for the illustration of our IBBE scheme based on ECC. The algorithms for various computations related to points on an elliptic curve, programs implemented in computer algebra system ApCoCoA.

Contents

List of Figures

List of Tables

Abbreviations

IBE	Identity Based Encryption
IBBE	Identity Based Broadcast Encryption
PKC	Public Key Cryptography
PKI	Public Key Infrastructure
PKA	Public Key Authority
CA	Certificate Authority
PKG	Private Key Generator

Symbols

M	Plaintext or Message
C	Ciphertext
E	Encryption Algorithm
D	Decryption Algorithm
PR	Private Key/ Secret Key
ID	Recipient's Identity
$\mathcal{C}'$	Ciphertext Space
$\mathcal{M}'$	Message Space
$\mathcal{A}$	Adversary
$\mathcal{PP}$	Public parameters
$\mathbb{G}$	Group
$\mathbb{Z}$	Set of integers
$\mathbb{R}$	Set of real numbers
$\mathbb{Q}$	Rational numbers
$\mathbb{C}$	Complex numbers
$\mathbb{V}$	Vector space
$\mathbb{G}^*$	Multiplicative Group
$p,\ q$	Prime number
$\mathbb{F}_p$,$\mathbb{Z}_p$	Finite field of order prime p
$\mathbb{F}_{p^k}$	Finite field extension
$\mathcal{K}$	Master Key
$E_{\mathbb{F}}(a, b)$	Elliptic Curve Over field $\mathbb{F}$ and parameters $a,\ b$
$\mathcal{O}$	Point at Infinity

A, B, C Points on elliptic curve

Chapter 1

Introduction

1.1 Cryptography

The security of communication remained a big problem from very beginning. Roman knew some cryptographic methods and used the Shift Cipher or Caesar Cipher [15] while communicating with each other. As the time passed, new methods in cryptography were developed that provided more security. Cryptography [48] is the study of transmitting message in such form so that no third person can read and process it. It is the technique that uses mathematical functions for securing the data or information from adversaries. The original message known as plaintext is converted into codded message (ciphertext) via the encryption algorithm for transmitting it to the public network. The ciphertext is then converted back to plaintext by the receiver or an authorized person via the decryption algorithm. Both sender and receiver use a secret information (known only to sender and receiver) for encryption and decryption algorithms. this secret information known as a key. The entire process is called cryptosystem. The security of a cryptosystem relies only on the security of the key.

Depending on the key, the cryptographic scheme are divided in two main categories, namely Symmetric key cryptography and Asymmetric key cryptography.

In symmetric key encryption [28], only one key is used for both encryption and decryption algorithm that is only known to sender and receiver. Examples of such scheme include DES [7] and AES [38]. The key distribution is main problem in this method. When we have thousands of users to communicate with each other then the distribution of key among all the participants becomes a serious issue.

To solve this problem, Diffie Helman [19] proposed the idea of Asymmetric key cryptography which is also known as Public key cryptography (PKC) [48]. In PKC, recipient has two types of keys for communication one is public key that is made public and the other is private key that is kept secret. For example, RSA [42], ElGamal [47].

There are some issues regarding use of PKC, the issue is how to trust the public key. In order to solve this issue an authority is maintained in which both recipients (sender and receiver) can trust, is called certificate authority (CA) [30]. In certificate system, both recipients submit their public keys to authority and authority verifies their keys and issues some certificates. But management of certificates is complex and cumbersome. The problem associated with PKC is solved by Shamir [45] to introduced the idea of identity based encryption scheme.

1.2 Identity Based Encryption Scheme

Identity based encryption scheme (IBE) is a type of public key cryptography, in which public key of recipient can be an arbitrary string that may be his email address and for private key, the recipient authenticates himself from third party known as Private Key Generator (PKG), whenever recipient first connects to the system. The direct derivation of public key eliminates the need of certificate and reduces the complexity in the system.

In 1984 Shamir [45] introduced the notion of IBE, but he was unable to construct practical scheme of IBE. There have been several proposals for IBE see [18, 35, 49, 50]. But none of these are fully acceptable. Some schemes take a lot of time in generating the private key by private key generator (PKG). The first successful scheme was presented by Boneh and Franklin [10] in 2001. From that time, the IBE is currently active in research area. One prominent application of IBE is its use in broadcast initiated by Fiat and Naor [22].

1.3 Identity Based Broadcast Encryption Scheme

The term broadcast refers to the system in which message is transmitted to multiple users. There is only one sender called a centre or broadcaster and it transmits message to multiple receiver called as subscribers or privileged users. No one from the outside of the set of receiver is able to decrypt the message. FM radio is an example of broadcasting.

Identity based broadcast encryption scheme (IBBE) is generalization of IBE scheme. In IBE there is only one sender and one receiver but in IBBE there is only one sender (known as Center) and set of receivers. Centre encrypts the message by using receiver's identity and send it to corresponding receiver. The trusted third party Private key generator (PKG) generates private key of each receiver in the set. The corresponding receiver can decrypt the message by using their private keys.

The concept of broadcast encryption (BE) was first introduced by Fait and Noar [22], which is based on symmetric key encryption. Later, the first successful scheme on BE [20] was introduced that depends upon the public key encryption scheme. After that, there have been many proposals for BE scheme see [5, 21, 25, 39]. Then, there was a proposal on Identity Based Broadcast Encryption Scheme(IBBE) in 2007 see [44]. For several other proposals on IBBE we refer to [6, 11, 17].

1.4 Current Research

In this research, we focused on IBBE scheme introduced by Ming and Wang [37]. They proposed IBBE with group of prime order. Their construction is based on bilinear mapping. Also, they use dual pairing vector space [16] technique as a tool in their scheme. They scheme achieve constant size of system parameter, private key and ciphertext. Furthermore, they use the dual system encryption [52] for the sake of security.

Due to small key size, elliptic curve cryptography (ECC) has gained considerable importance in recent years[51]. We mainly focused on the implementation of Ming and Wang's Scheme [37] using group of points of elliptic curve [51]. Another reason for using ECC is the weil pairing [34] which is considered to be good candidate of bilinear mapping [46]. Using a suitable elliptic curve and weil pairing we constructed two toy examples for the illustration of our IBBE scheme based on ECC. The algorithms for various computations related to points on an elliptic curve, programs implemented in computer algebra system ApCoCoA [1].

1.5 Thesis layout

Our thesis is organised as follow:

- In **Chapter 2**, we discussed identity based encryption scheme (IBE) in detail. We started with basic definitions for cryptography and issues related to its key management. Then we showed the drawbacks of certificate authority over identity based encryption scheme.

- In **Chapter 3**, we presented the examples of bilinear pairing. For that purpose we choose the elliptic curve and recall its basic definitions and properties. Furthermore, we described the concept of weil pairing and its modified form. We implemented weil pairing and modified form by using ApCoCoA tool [1].

- In **Chapter 4**, we discussed the generalized form of IBE called Identity Based Broadcast Encryption Scheme. We presented the review of IBBE scheme on the group of prime order introduced by Ming and Wang[37].

- In **Chapter 5**, we implemented Ming and Wang scheme [37] using group of points on an elliptic curve and weil pairing [34], which is good example of bilinear mapping on elliptic curve groups. The modified scheme [37] is illustrated by toy examples.

Chapter 2

Preliminaries

In this chapter, we discuss new scheme known as identity based encryption scheme introduced by Shamir [45]. First of all we will define cryptography and issues related to its key management. Furthermore, we will highlight the drawbacks of certificate authority over identity based encryption scheme. We also recall some basic definitions from algebra that will be using throughout in this thesis.

2.1 Cryptography

Cryptography is the branch of cryptology[1], in which communication take place in the secure fashion in such a way that no third party can read or change the information. The sender converts the original message or data (known as plaintext) into coded or scrambled message (called the ciphertext). The process of converting the plaintext into ciphertext is called encryption and process of converting ciphertext back into plaintext is called decryption. On the basis of keys used, cryptography is divided into two categories.

- Symmetric Key Cryptography
- Public Key Cryptography

[1] Cryptology is the science of secret communication

2.1.1 Symmetric Key Cryptography

Symmetric key cryptography [28] is also called the secret key cryptography. It was the only technique that is used for transmitting messages before the development of public key cryptography. In this method, only one key is used for both encryption and decryption. A typical symmetric key cryptography model is shown in Figure 2.1, in which both sender and receiver are using a common key K for encryption and decryption which is not known to adversary (attacker).

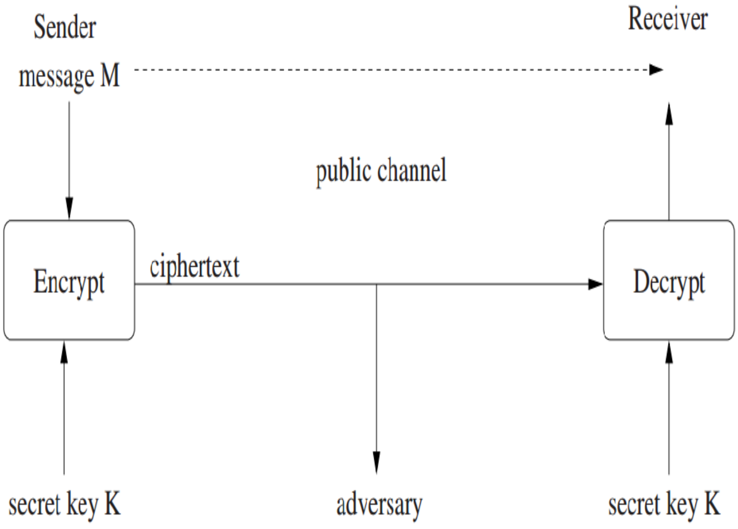

FIGURE 2.1: Symmetric Key Cryptography [14]

The drawbacks of symmetric key cryptography are as follow:

1. **Key sharing:** If there are n number of people communicating with each other, then key distribution is the problem. If one person discloses the key then the whole communication will be compromised.
2. **Authentication:** One of the main problem is authentication, if Alice and Bob communicate with each other then how can Alice prove that the message has arrived from Bob.

2.1.2 Public Key Cryptography

To solve this issue with symmetric key cryptography, Diffie-Welman [19] proposed the idea of public key cryptography in 1976. Their concept is based on one-way trapdoor function for exchanging the key between two parties. Public key cryptography [48] (PKC) allows the communicant to make the encryption key that is make available for all and decryption keys are kept hidden.
The development of PKC is the greatest achievement in the history of cryptography. From earliest to modern time, virtually all cryptosystems worked on permutation and substitution. Moreover, the public key crptosystem based on mathematical function instead of substitution and permutation. Generally, In public key cryptography, encryption and decryption is performed by two different keys one is a public key and the other is a private key. It is likewise known as Asymmetric Encryption. The public key encryption scheme uses six main elements as shown in Figure 2.2. The sender encrypts the plaintext M by using receiver's public key PU and an encryption algorithm E to get the ciphertext C. Then receiver uses his private key PR (that is only known to him) and decrypts ciphertext using corresponding decryption algorithm D. Thus,

$$C = E(PU, M) \tag{2.1}$$

$$M = D(PR, C) \tag{2.2}$$

2.2 Key Management Issues

In PKC, the distribution of public key is the main problem regarding the key management. Many techniques have been proposed for distribution of public key encryption. Some are as follow.

- Public Announcement
- Public Available Directory

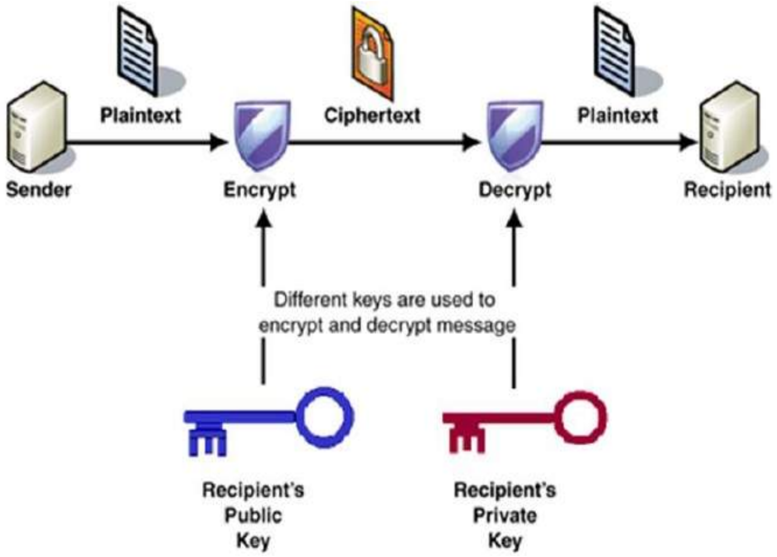

FIGURE 2.2: Public Key Cryptography

- Public key Authority
- Public Key Certificate

2.2.1 Public Announcement

One of the main problem faced by public key encryption scheme is that the public key should be available to everyone. There are algorithms used PGP (Pretty Good Privacy) [26] in which any recipient, sends its public key to other recipients via email or make public announcement [48] as shown in Figure 2.3

One of the major weaknesses regarding public announcement is forgery. Any person could claim to be user A and send his public key to B. In this way forger is able to read all encrypted messages.

2.2.2 Public Available Directory

Greater security can be attained by maintaining public available directory [48]. The trusted authority or system would be responsible for maintenance and distribution of public keys as shown Figure 2.4. Following are the key points of this system.

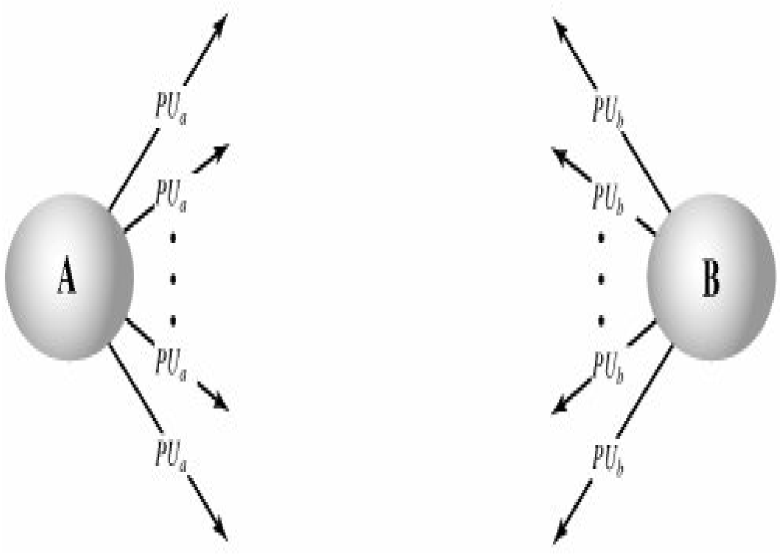

FIGURE 2.3: Public Announcement [48]

1. The authority maintains the record of the name and the public key of each recipient.
2. Any participant can registers his public key with authority. Registration would be in the form of secure communication.
3. If the private key is compromised, then participant can replace his existing key with new one at any time.
4. The participant can access the directory electronically. For this purpose, it is mandatory for participants to communicate with authority securely.

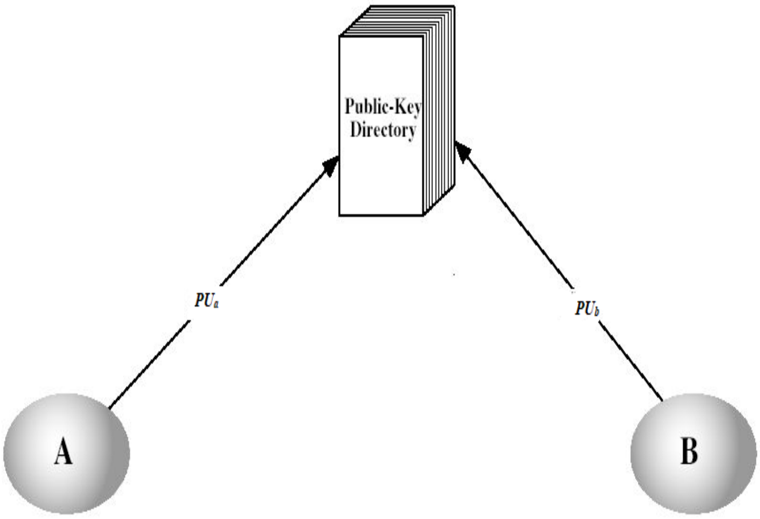

FIGURE 2.4: Public Available Directory [48]

2.2.3 Public Key Authority

Greater security can be achieved by tightening control over the central authority or directory. In this scheme [4], the public key authority, is employed to maintain the directory of public key of all recipients. Therefore, all participants reliably know the public key from central authority, with only authority knowing their corresponding private key. The following are steps as presented in Figure 2.5.

1. A sends time stamped request to central authority for current public key of B
2. Authority encrypt message with his private key (PR_{auth}). The message of authority contains the B's public key PU_b, original request and time stamped as in equation 2.3. So, in this way A can verify that this is not the old message containing B's public key.

$$E(PR_{auth}, [PU_b||Request||Time]) \tag{2.3}$$

3. A store B's public key and uses it to encrypt the message that contain A's identity (ID_A) and nonce (N_1) generated by A as describe in 2.4

$$E(PU_b, [ID_a||N_1]) \tag{2.4}$$

4. The same procedure is repeated by B for obtaining A's public key (PU_a) as described in (1) and (2).
5. When B sends a message to A he encrypts message with A's public key (PU_a), and with random number (N_1) this can be used to verify the original message generated by A and another random number (N_2) generated by B as described in equation 2.5

$$E(PU_a, [N_1||N_2]) \tag{2.5}$$

6. A returns the random number N_2 by using B's public key PU_b to ensure that the original message is sent by B.

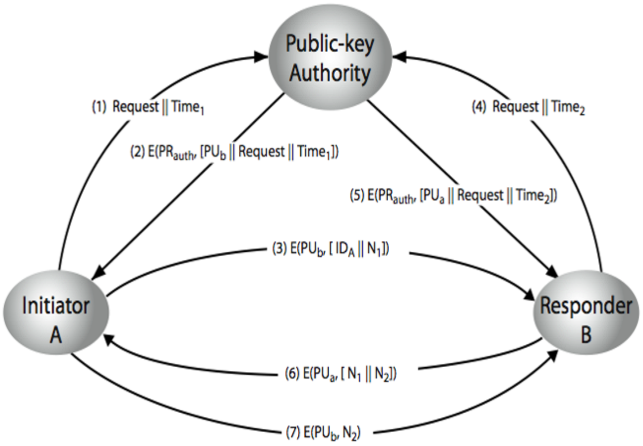
Public-key Authority
(1) Request || Time_1
(2) $E(PR_{auth}, [PU_b \| \text{Request} \| \text{Time}_1])$
(4) Request || Time_2
(5) $E(PR_{auth}, [PU_a \| \text{Request} \| \text{Time}_2])$
(3) $E(PU_b, [ID_A \| N_1])$
Initiator A
Responder B
(6) $E(PU_a, [N_1 \| N_2])$
(7) $E(PU_b, N_2)$

FIGURE 2.5: Public Key Authority [48]

2.2.4 Public Key Certificate

Although, public key authority (PKA) is an efficient scheme, but it has some drawbacks. The public key authority could be a greater threat to a system, because user must contact with authority for the public key of other users that it wishes to contact. If some adversary had broken the public key authority, then the whole system will be compromised. Even without breaking the public key authority, some imprisonment is also possible by tampering the record of directory that is maintained by the public key authority. Furthermore, the use of public key authority frequently needs a large and complex system and it is really difficult to update such a system securely.

Therefore, the concept of public key certificate (PKC) had been introduced by Felder [30] to use certificate for communication without contacting the public key authority. The certificate is the signed message that consists of a public key of owner plus an identity of key owner and this whole blog is signed by a third party. Typically, this third party is certificate authority. Note that Figure 2.6 shows the certificate scheme, in which both recipients A and B supply their public keys PU to certificate authority and requesting for certificate. Certificate authority (CA) issues certificate for both recipients by using their private keys PR. So, A may pass their certificate to B, and B reads and verifies it by using authority's public key PR_{auth} and certificate C_A.

There are some benefits of certification which are stated as under:

1. Any participant can read a certificate and determine the name and pubic key of certificate's owner.
2. Any participant can verify the certificate that is created by Certificate.
3. Only certificate authority can create, modify and manage the certificate.
4. The participant can also verify the time period of every certificate.

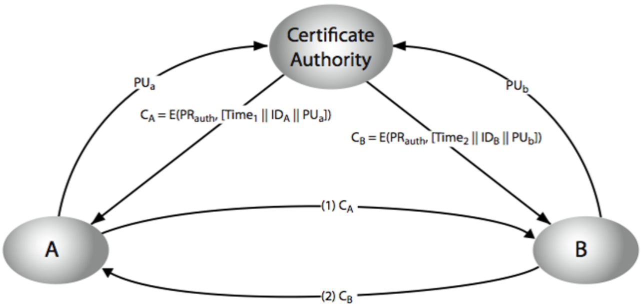
Certificate Authority
PU$_a$
PU$_b$
$C_A = E(PR_{auth}, [Time_1 || ID_A || PU_a])$
$C_B = E(PR_{auth}, [Time_2 || ID_B || PU_b])$
(1) C_A
(2) C_B
A
B

FIGURE 2.6: Public Key Certificate [48]

2.2.4.1 Drawbacks of Certificate Authority

Although, the public key certificate [2] is a very efficient scheme, but it has some drawbacks.

1. When user A wants to communicate with user B, both recipients need a certificate in order to communicate with each other. For offline operations a certificate is required in order to communicate with each other. So for that purpose large scale directory is needed for managing the certificates for offline use.
2. Certificates are large and complex structure so it is hard to update such a system securely.
3. Since the certificate keeps all public and private keys therefore these are large and very expensive schemes.

4. The authority does not give warning when it changes the certificate.
5. A user blindly trusts on certificate authority, if some third party generates the fraudulent certificate and gains access to someone's personal computer. So, in this way certificate authority does not give warning when any site uses the fraudulent certificate.
6. In PKI (Public Key Infrastructure) before the communication takes place the system must register its encryption and signature key to CA, then CA issues the certificate for the proof of its identity. Then this certificate is used by recipient for secure communication (Figure 2.7). Therefore, this method is also time consuming.

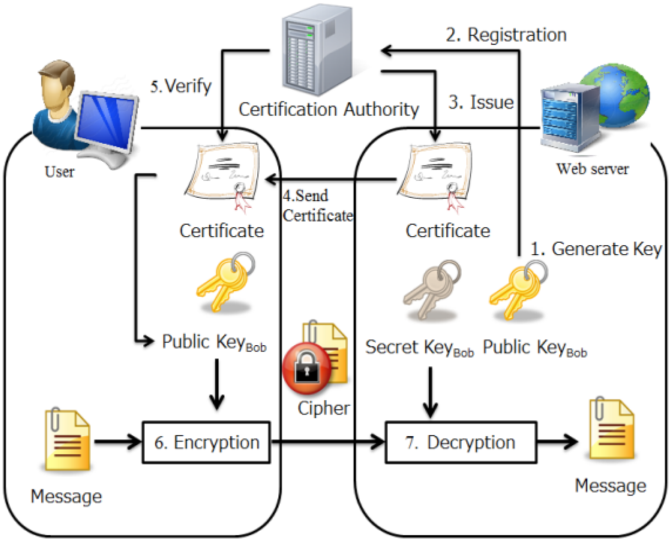

FIGURE 2.7: Public Key Infrastructure

2.3 Introduction to Identity Based Encryption Scheme

To solve the certificate management system, Shamir [45] introduced a new scheme called as Identity Based Encryption Scheme (IBE) in 1984. IBE is a very efficient scheme and currently active in research area of cryptography. This scheme uses an arbitrary string such as a user's identity, email address or IP address and derives the public key from it. The direct derivation of public key eliminates the role of the certificate. Only private keys are generated from trusted third party also called Private Key Generator (PKG). So, in this way large directories are not required for managing public keys of users. In IBE, the private key authority exists only, it does not need to be online, its action replaces with mathematical pairing. Note that in Figure 2.8 when Alice send message to Bob she must contact to certificate authority for Bob's certificate CA look up the Bob's certificate from certificate server and send certificate to Alice. From certificate, Alice uses the public key of Bob, PU_{Bob}, and apply the encryption by using PU_{Bob}. When Bob receives encrypted message he sends his public key to CA and receives the certificate that include his private key. Bob decrypt the message by using his private key. Where, in Figure 2.9 shows that IBE does not need certificate server for keeping the record of recipient's public key. No certificate look up required. IBE need only the private key generator for deriving the private keys by using recipient's identity.

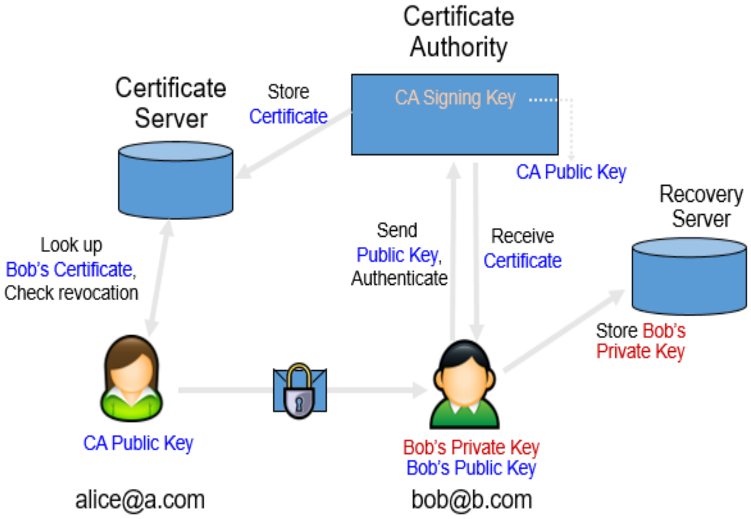

FIGURE 2.8: Public key certificate

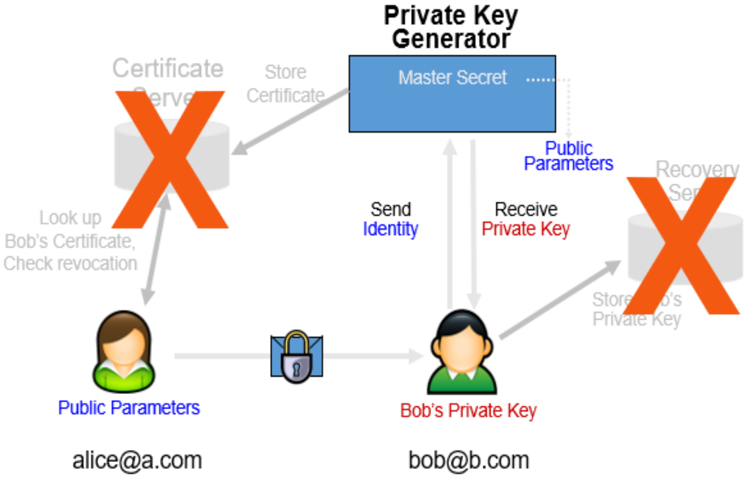

FIGURE 2.9: Identity Based Encryption Scheme

2.3.1 Identity Based Encryption Scheme

As discussed in previous section, identity based encryption scheme (IBE) was first proposed by Shamir [45] in 1984. In this scheme, the pairs of users can communicate and verify each other without sharing their public and private keys, without keeping key directories and without taking the services of third parties. In IBE, the third party is used to generate the private keys in the shape of smart cards when users first connect the network.

IBE scheme is based on public key cryptosytem, but holds some extra key points. Instead of generating the random public keys by using the help of a third party, IBE scheme uses any combination of a user's name, IP address, telephone or office number etc. as a public key. IBE scheme resembles the mail services: if one user knows someone's e-mail address then he will be able to communicate with that user.

Identity based encryption scheme works as follows:

1. User A wants to communicate with B, he signs it with his secret key in his smart card. He encrypt the message by using B's identity (B's name, address etc.) and sends it to B.

2. When B receives the encrypted message, he contacts to third party for obtaining private key (PR_b).
3. B decrypt the message by using his (PR_b) in smart card or verify the message by user's A identity

Here the third party or key generation centre is the trusted party that generates the secret keys of all users. Centre knows some secret information (such as factorization of large numbers). The secret key is issued in the shape of smart cards to all users who join the mesh. The smart card contains a microprocessor, RAM, ROM that contains secret key and the program that contains the message encryption and decryption algorithm. The query is how user can secure his smart card? The user must secure his smart card by using password system or memorizing the part of the key.

The Figure 2.10 show the system of IBE. Shamir's IBE consists of four algorithms.

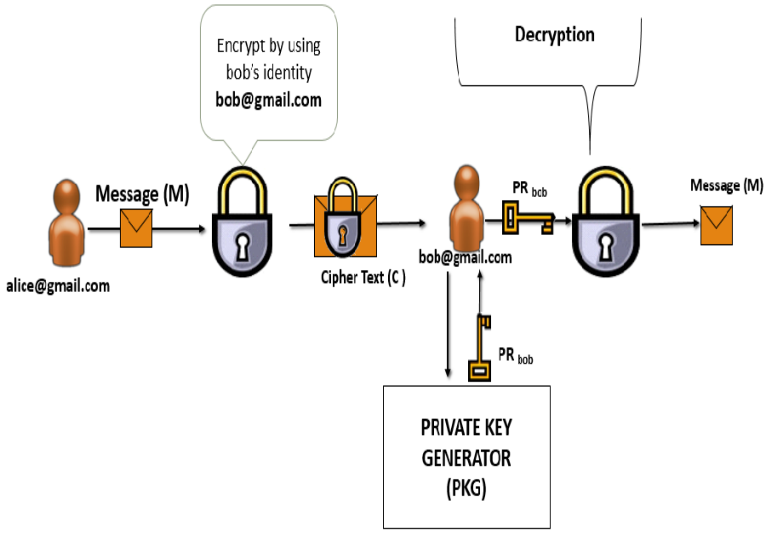

FIGURE 2.10: Identity Based encryption Scheme

1. **Setup:** The setup is the component of a private key generator (PKG). PKG generates the master key $\mathcal{K}$ and public parameter $\mathcal{PP}$. Where master key is kept secret. Public parameter contains the information about message space $\mathcal{M}'$ and ciphertext space $\mathcal{C}'$

2. **Extract**: The PKG runs this extract algorithm, makes session keys or private key for user using his master key and user's identity (ID). This algorithm accepts the identity (ID) of user and master key $\mathcal{K}$ generates private key (S_{ID}) of corresponding identity (ID).
3. **Encrypt**: This algorithm accepts identity (ID) and message as input and produce ciphertext as output.

$$C = E(M, ID)$$

4. **Decrypt**: This algorithm takes ciphertext (C) and private key (S_{ID}) as input and returns messages.

$$M = D(C, S_{ID})$$

There have been several proposals for IBE see [18, 35, 49, 50], but none of these are fully acceptable. Some solutions take a lot of time in generating the private key from private key generation (PKG). The first successful scheme was presented by Boneh and Franklin [10] in 2001. Their scheme is based on bilinear maps defined on prime order groups. They used Weil pairing on elliptic curves as an example such map.

2.3.2 Identity Based Broadcast Encryption Scheme

Identity based broadcast encryption scheme (IBBE) is the generalized form of IBE scheme. In this scheme there is only one sender and having multiple receivers. Sender is called centre or broadcaster and receiver is called subscriber. The scheme was first introduced by Fiat and Naor [22], in which broadcaster encrypt message by using the set of identities and which is decrypted by the only subscriber. The detailed description is given in Chapter 4.

2.4 Cryptanalysis Attacks

The cryptanalysis is branch of cryptology in which we study ciphers, ciphertext and cryptosystem and find weakness in them. It is process of deciphering the ciphertext without the knowledge of secret key. Any attempt to recover the secret key or break the ciphertext is called an Cryptanalysis attacks. Their are many cryptanalysis attacks but we focus on two attacks chosen plaintext attack and chosen ciphertext attack.

- **Chosen Plaintext Attack:** In Chosen Plaintext Attack [12] the adversary can choose the plaintext instead of choosing the big block of text he choose the smaller block and gets its corresponding ciphertext. His goal is to break the secret ciphertext or recover the the secret key.

- **Chosen ciphertext attack:** In Chosen-ciphertext attack [12] the adversary can choose ciphertext and can predict the corresponding plaintext. He has an ability to make the decryption of ciphertext and then regenerate the resulting plaintext from system. In this way he can analyze the secret key.

2.5 Mathematical Background

Before details on IBBE scheme, we first recall some definition from algebra that will be used through the thesis.

Definition 2.5.1 (Groups)

The ***group*** [43] $\mathbb{G}$ denoted by $(\mathbb{G}, *)$ is the set of element under the binary operation * that satisfies the following properties:

1. **Closure:** For all $x, y \in \mathbb{G}$, $x * y \in \mathbb{G}$

2. **Associative:** For all $x, y, z \in \mathbb{G}$ satisfies $(x * y) * z = x * (y * z)$

3. **Identity:** There exist an element $i \in \mathbb{G}$ that satisfies $x * i = i * x = x \quad \forall x \in \mathbb{G}$. i is called the identity of $\mathbb{G}$.

4. **Inverse:** For each element $x \in \mathbb{G}\ \ \exists\, x' \in \mathbb{G}$ the satisfies $x * x' = x' * x = i$. Where i is the identity element of $\mathbb{G}$

Example 2.5.2 Following are the examples of groups.

1. Set of integers $\mathbb{Z}$, real number $\mathbb{R}$, rational number $\mathbb{Q}$, complex number $\mathbb{C}$ are all group under binary operation addition $+$.

2. Set of real numbers $\mathbb{R} \setminus \{0\}$, rational number $\mathbb{Q} \setminus \{0\}$ and complex number $\mathbb{C} \setminus \{0\}$ all group under binary operation multiplication $\times$.

3. Let $\mathbb{Z}_m = \{0, 1, 2, ...m - 1\}$ and $m > 0$ and $m \in \mathbb{Z}$ is group under addition $x * y = x + y$ where $x + y < m$. The binary operation $+$ is called addition modulo m.

4. Set of integers $\mathbb{Z}$ does not form a group under multiplication because multiplicative inverse does not exist (Inverse of 2 is $\frac{1}{2}$ but $\frac{1}{2} \notin \mathbb{Z}$)

Definition 2.5.3 (Abelian Group)

The group $\mathbb{G}$ is said to be abelian group [43] if it satisfies commutative law i.e. for all $x, y \in \mathbb{G}$ we have $x * y = y * x$.

Example 2.5.4 Following are the example of abelian groups.

1. Sets $\mathbb{Z},\ \mathbb{R},\ \mathbb{C},\ \mathbb{Q}$ are abelian group under addition.

2. Sets $\mathbb{R} \setminus \{0\}$, $\mathbb{Q} \setminus \{0\}$, $\mathbb{C} \setminus \{0\}$ form abelian groups w.r.t multiplication.

3. **General Linear Group** is defined as $GL(m) = \{A \in M(m, m) | det(A) \neq 0\}$ where $M(m, m)$ is matrix of order $m \times m$ is a group under multiplication. It is not an abelian group because matrix multiplication is not commutative.

Definition 2.5.5 (Generator)

The generator g is a group element that is capable to generate all group elements.

Definition 2.5.6 (Cyclic Group)

The finite group $\mathbb{G}$ of order n is said to be cyclic if there exists an element $g \in \mathbb{G}$ that generates all elements of $\mathbb{G}$. That is,

$$\mathbb{G} = \{g, g^2, g^3, \ldots, g^n = I\}$$

Where I is the identity element of group $\mathbb{G}$ where g is the generator of $\mathbb{G}$.

Definition 2.5.7 (Field)

The triples $\{\mathbb{F}, +, \times\}$ that is, a set $\mathbb{F}$ together with binary operations $+, \times$ is called field $\mathbb{F}$ if the following properties are satisfied for all $x, y, z \in \mathbb{F}$.

S.No	**Name**	**Addition** $+$	**Multiplication**$\times$
1	Associative	$(x + y) + z = x + (y + z)$	$(x \times y) \times z = x \times (y \times z)$
2	Distributive	$x(y + z) = xy + xz$	$(x + y)z = xz + yz$
3	Commutative	$x + y = y + x$	$x * y = y * x$
4	Identity	$x + 0 = x$	$x \times 1 = x$
5	Inverse	$x + (-x) = 0$	$x(x^{-1}) = 1$

TABLE 2.1: Properties of Field

Example 2.5.8 Following are the examples of fields.

1. Set of real numbers $\mathbb{R}$, set of complex numbers $\mathbb{C}$ and set of rational numbers $\mathbb{Q}$ are field under addition and multiplication.

2. Set of integers $\mathbb{Z}$ is not a field. Because multiplicative inverse of integers not exist.

Definition 2.5.9 (Finite Field)

The **Finite field** or **Galois Field** is the field that has finite number of elements. Particularly, the order of finite field is must be the power of prime number p^n written as $\mathbb{F}_{p^n}$ or $GF(p^n)$ when $n = 1$ finite field has set of integers of modulo p represent in the form of $\{0, 1, 2, ..., p - 1\}$.

Example 2.5.10 Caley table 2.2 and 2.3 show the finite field $\mathbb{F}_{11}$ or $GF(11)$ under addition and multiplication in modulo 11.

Every element has unique additive and multiplicative inverses that is shown in 2.4. eg. additive inverse of 5 is 6 because $5 + 6 = 0 \mod 11$. The multiplicative inverse of 5 is 9 because $5 \times 9 = 1 \mod 11$. Additive identity of the field is 0 and multiplicative identity is 1.

+	0	1	2	3	4	5	6	7	8	9	10
0	0	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10	0
2	2	3	4	5	6	7	8	9	10	0	1
3	3	4	5	6	7	8	9	10	0	1	2
4	4	5	6	7	8	9	10	0	1	2	3
5	5	6	7	8	9	10	0	1	2	3	4
6	6	7	8	9	10	0	1	2	3	4	5
7	7	8	9	10	0	1	2	3	4	5	6
8	8	9	10	0	1	2	3	4	5	6	7
9	9	10	0	1	2	3	4	5	6	7	8
10	10	0	1	2	3	4	5	6	7	8	9

Table 2.2: Finite field under addition in mod 11

·	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	10
2	0	2	4	6	8	10	1	3	5	7	9
3	0	3	6	9	1	4	7	10	2	5	8
4	0	4	8	1	5	9	2	6	10	3	7
5	0	5	10	4	9	3	8	2	7	1	6
6	0	6	1	7	2	8	3	9	4	10	5
7	0	7	3	10	6	2	9	5	1	8	4
8	0	8	5	2	10	7	4	1	9	6	3
9	0	9	7	5	3	1	10	8	6	4	2
10	0	10	9	8	7	6	5	4	3	2	1

Table 2.3: Finite field under multiplication in mod 11

Numbers	**Additive Inverse**	**Multiplicative Inverse**
0	0	–
1	10	1
2	9	6
3	8	4
4	7	3
5	6	9
6	5	2
7	4	8
8	3	7
9	2	5
10	1	1

Table 2.4: List of additive and multiplicative Inverses

Definition 2.5.11 (Extension Field)

Let $\mathbb{F}$ and E be the two fields then $\mathbb{F}$ is said to be extension field of E, if E is the subfield of $\mathbb{F}$. It is denoted by $\mathbb{F}/E$. Furthermore, "let $\mathbb{F}$ be the field and $p(x)$ be any non-constant polynomial. Then there exist the extension E of $\mathbb{F}$ in which $p(x)$ has zero denoted by $E = \mathbb{F}/p(x)$."[8]

Example 2.5.12

1. $\mathbb{R}$, the field of real numbers is the extension field of $\mathbb{Q}$, the field of rational numbers, denoted by $\mathbb{R}/\mathbb{Q}$.

2. $\mathbb{C}$, the field of complex numbers is the extension field of $\mathbb{R}$, field of real numbers, denoted by $\mathbb{C}/\mathbb{R}$.

3. Let $x^5+2x^2+2x+1 \in \mathbb{Z}_3(x)$ then the irreducible factorization of $x^5+2x^2+2x+1 = (x^2+1)(x^3+2x+1)$. There exist extension fields E of $\mathbb{Z}_3(x)$ is $E = \mathbb{Z}_3(x)/(x^3+2x+1)$ and $E = \mathbb{Z}_3/(x^2+1)$.

4. Let $p(x) = x^2+x+2 \in \mathbb{Z}_3(x)$ then there exist the extension field E of $\mathbb{Z}_3$ such that $E = \mathbb{Z}_3(x)/x^2+x+2$. The field $\mathbb{Z}_3(x)/x^2+x+2$ is represented as $\{0, 1, x, 2x, x+1, 2x+1, x+2, 2x+2\}$. Note that $(x^2+x+2)+(x^2+x+2) = 0$ this implies the fact $x^2+x+2 = 0$ so $x^2 = -x-2 = 2x+1$. Therefore, in E there exist the polynomials that are irreducible in $\mod (x^2+x+2)$.

Definition 2.5.13 (Multiplicative Inverse in finite fields)

It is very easy to find the multiplicative inverse of integers in small field by simply construct their caley table and find the inverses of any integer. For example, Table 2.4 shows the inverses but it is not practical when we are dealing with large fields. Multiplicative inverse of any number $b \in \mathbb{F} \mod m$ is possible if $gcd(b, m) = 1$ otherwise it is not possible. The Extended Euclidean algorithm is use to finding the inverse of any integer b in $\mod m$ which is stated below.

Algorithm 2.5.14 (Extended Euclidean Inverse)

1. "Set $(A_1, A_2, A_3) = (1, 0, m)$ and $(B_1, B_2, B_3) = (0, 1, b)$

2. If $B_3 = 0$ then return with answer that $A_3 = gcd(m, b)$ no inverse of element b exist

3. Now check If $B_3 = 1$ then return $B_3 = gcd(m, b)$ $B_2 = b^{-1} \mod m$

4. Now divide A_3 and B_3 set the quotient $Q = A_3 \; div \; B_3$

5. Now let we take $(T_1, T_2, T_3) = (A_1 - Q.B_1, A_2 - Q.B_2, A_3 - Q.B_3)$

6. Set $(A_1, A_2, A_3) = (B_1, B_2, B_3)$

7. Set $(B_1, B_2, B_3) = (T_1, T_2, T_s)$

8. Goto step number 2".[48]

Definition 2.5.15 (Multiplicative Inverse in Extension field)

As previously the extended euclidean Algorithm is also use to find inverse of any polynomial in extension field. This algorithm will find the multiplicative inverse of any polynomial $b(x) \in \mathbb{F}$ modulo an irreducible polynomial when $gcd(b(x), m(x)) = 1$. To find the inverse of $b(x) \mod m(x)$, the following are the steps performed [48]

Algorithm 2.5.16 (Extended Euclidean Inverse)

1. "Set $(A_1(x), A_2(x), A_3(x)) = (1, 0, m(x))$ and $(B_1(x), B_2(x), B_3(x)) = (0, 1, b(x))$

2. If $B_3(x) = 0$ then return with answer that $A_3(x) = gcd(m(x), b(x))$ no inverse of $b(x)$ exist

3. Now check If $B_3(x) = 1$ then return $B_3(x) = gcd(m(x), b(x))$
   $B_2(x) = b^{-1}(x) \mod m(x)$

4. Now divide $A_3(x)$ and $B_3(x)$ set the quotient $Q(x) = A_3(x) \; div \; B_3(x)$

5. Set $(T_1(x), T_2(x), T_3(x)) = (A_1(x) - Q(x).B_1(x), A_2(x) - Q(x).B_2(x), A_3(x) - Q(x).B_3(x))$

6. Set $(A_1(x), A_2(x), A_3(x)) = (B_1(x), B_2(x), B_3(x))$

7. Set $(B_1(x), B_2(x), B_3(x)) = (T_1(x), T_2(x), T_3(x))$

8. Goto step number 2."[48]

Definition 2.5.17 (Vector Space)

Let $\mathbb{V}$ be the non-empty set over the field $\mathbb{F}$ then $\mathbb{V}$ is the vector space [27] along with two binary operations that is vector addition and scalar multiplication.

1. **Vector Addition:** Let $v, w \in \mathbb{V}$ such that $v + w \in \mathbb{V}$
2. **Scalar Multiplication:** Let $a \in \mathbb{F}$ and $v \in \mathbb{V}$ then $a.v \in \mathbb{V}$

and satisfying the following properties:

1. V is the **Abelian group** (Definition 2.5.3) under addition.
2. $a(v + w) = av + aw \quad \forall a \in \mathbb{F}$ and $v, w \in \mathbb{V}$
3. $(a + b).v = a.v + b.v \quad \forall a, b \in \mathbb{F}$ and $v \in \mathbb{V}$
4. $a(b.v) = (a.b)v \quad \forall a, b \in \mathbb{F}$ and $v \in \mathbb{V}$
5. $1.v = v.1 = v$ where $1 \in \mathbb{F}$ and $v \in \mathbb{V}$, 1 is the identity of $\mathbb{F}$

Note: Every Field $\mathbb{F}$ is the vector space over itself

Example 2.5.18

1. Set of polynomials $\mathbb{P}_n$ having degree less than and equal to n is defined as:
   $\mathbb{P}_n = \{a_1.x^1 + a_2.x^2 + \cdots + a_n.x^n | a_i \in \mathbb{F}\ ,\ i \leq n \in \mathbb{N}\}$
   $\mathbb{P}_n = \{\sum_{i=0}^{n} a_i.x^i | a_i \in \mathbb{F}\ ,\ i \leq n \in \mathbb{N}\}$
   Vector addition is defined as:
   $\sum_{i=0}^{n} a_i.x^i + \sum_{i=0}^{n} b_i.x^i = \sum_{i=0}^{n}(a_i + b_i)x^i \quad \because a_i, b_i \in \mathbb{F}$
   and **Scalar multiplication** is defined as:
   $\alpha. \sum_{i=0}^{n} a_i x^i = \sum_{i=0}^{n} \alpha a_i x^i \quad \because \alpha \in \mathbb{F}$
2. The set M_n of all $n \times n$ matrices with all entries from the field $\mathbb{F}$ is the vector space over $\mathbb{F}$

Definition 2.5.19 (Spanning Set/Linear Span)

Let S be the set of vectors S then set of all linear combination of S is called linear span or spanning set [27] of S. This span a vector space is $\langle S \rangle$.

Definition 2.5.20 (Linearly Independent)

Let $\mathbb{V}$ be a vector space over the field $\mathbb{F}$. Then the set of vectors $\{v_1, v_2, v_3, \ldots, v_n\}$ are said to be linearly independent [27] if

$$a_1.v_1 + a_2.v_2 + a_3.v_3, \ldots, a_n.v_n = 0$$

implies each $a_i = 0$

Definition 2.5.21 (Basis)

Let B be the subset of a vector space $\mathbb{V}$ over the field $\mathbb{F}$. Then B is the basis [27] of $\mathbb{V}$, If

1. B is the spanning set of $\mathbb{V}$.
2. B is linearly independent.

Example 2.5.22 The set $\{v_1, v_2\} = \{(2, 2), (-3, 5)\}$ forms a basis of $\mathbb{R}^2$. Infact, let $(x, y) \in \mathbb{R}^2$

$$\begin{aligned}(x, y) &= a_1(2, 2) + a_2(-3, 5)\\ &= (2a_1 - 3a_2, 2a_1 + 5a_2)\end{aligned}$$

It can be written as

$$2a_1 - 3a_2 = x$$

$$2a_1 + 5a_2 = y$$

$$\begin{pmatrix} 2 & -3 \\ 2 & 5 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}$$

Determinant of coefficient matrix is

$$\begin{vmatrix} 2 & -3 \\ 2 & 5 \end{vmatrix} = 10 + 6 = 16 \neq 0$$

Therefore, the system has unique solution. So, every element $(x, y) \in \mathbb{R}^2$ can be written as linear combination of elements in S. Now we check whether $\{v_1, v_2\}$ are linearly independent. Let

$$2a_1 - 3a_2 = 0$$
$$2a_1 + 5a_2 = 0$$

Solving the above equation simultaneously we get $a_1 = a_2 = 0$. Therefore, $\{v_1, v_2\}$ are linearly independent. It follows, that $\{v_1, v_2\}$ forms the basis of $\mathbb{R}^2$.

2.5.1 Bilinear Mapping

As discussed in Section 2.1.1 Diffie Helmen solved the problem of sharing the key between two parties. Choose group $\mathbb{G}$ of order prime p and g is the generator of $\mathbb{G}$. When two parties A and B wants to communicate with each other.

1. A and B chooses some $a \in \mathbb{Z}_p$ and $b \in \mathbb{Z}_p$ respectively.
2. A send g^a to B and B send g^b to A. Where g^a is public key of A and g^b is the public key of B.
3. A compute $SK_A = (g^y)^x$ as secret key of A and B compute $SK_B = (g^x)^y$

Note that if we have three parties A, B and C using the above method, it becomes difficult to calculate public key for C. A bilinear mapping can be used to overcome this difficulty

Definition 2.5.23 (Bilinear Mapping)

It is the mapping the combining two elements of two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ yields element of third group $\mathbb{G}_3$. It is linear in each argument. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ and $\mathbb{G}_3$ be the groups, the bilinear mapping is a function $\phi : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_3$ if it satisfies the following properties.

1. $\phi(X, Y + Z) = \phi(X, Y).\phi(X, Z)$

2. $\phi(X+Y,Z) = \phi(X,Z).\phi(Y,Z)$
3. let $\mathbb{G}_1$ and $\mathbb{G}_2$ be the two groups and bilinear mapping is defined as

   $\phi : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and let X and Y be the generator of $\mathbb{G}_1$ and let $a, b \in \mathbb{Z}$

$$aX = \overbrace{X + X + X + ...X}^{a}$$

$$bY = \overbrace{Y + Y + Y + ...Y}^{b}$$

$$\phi(X{+}X{+}X{+}X...{+}X, Y{+}Y{+}Y...{+}Y) = \overbrace{\phi(X,Y).\phi(X,Y).\phi(X,Y)\ldots\phi(X,Y)}^{ab}$$

   or

   $\phi(aX, bY) = \phi(X,Y)^{ab} \quad \forall a, b \in \mathbb{Z}$

Example 2.5.24 The following are the examples of bilinear mapping.

1. Matrix multiplication is bilinear mapping which is defined as

   $\phi : M_{n\times m} \times M_{m\times n} \rightarrow M_{n\times n}$
2. The dot product between vector space $\mathbb{R}^n$ is also bilinear defined as $\phi(v,w) = v_1.w_1 + v_2.w_2 \ldots v_n.w_n$. It is bilinear mapping in the sense because it is linear transformation in each of its variable.

Using this bilinear mapping, it is easy to exchange key for three parties using these steps.

1. A, B, C chooses $a,\ b,\ c \in \mathbb{Z}$ respectively, that are kept secret.
2. A, B, C publish $ag,\ bg,\ cg$
3. A compute $\phi(bg,\ cg) = \phi(g,\ g)^{bc}$ and calculate secret key

$$SK_A = (\phi(g,\ g)^{bc})^a = \phi(g,\ g)^{abc}$$

4. B compute $\phi(ag,\ cg) = \phi(g,\ g)^{ac}$ and calculate secret key

$$SK_B = (\phi(g,\ g)^{ac})^b = \phi(g,\ g)^{abc}$$

5. Similarly, C compute $\phi(ag,\ bg) = \phi(g,\ g)^{ab}$ and calculate secret key

$$SK_C = (\phi(g,\ g)^{ab})^c = \phi(g,\ g)^{abc}$$

For security purpose, we are interested in the bilinear mapping that satisfies additional two properties stated as:

1. **Non-Degenerate** The mapping ϕ is said to be non-degenerate

$$\phi(X, X) \neq 1$$

   where X is the generator of $\mathbb{G}_1$.

2. **Computable:** There exist an efficient algorithm that is used to compute the mapping $\phi(X, Y)$ for any $X, Y \in G_1$. The efficient algorithm has been discussed in Chapter 3 and Section 3.2.3

Example 2.5.25 Weil pairing is an example of such a map that we will discuss in Chapter 3.

Chapter 3

Weil Pairing on Elliptic Curve

In Chapter 4, we will discuss the identity based broadcast encryption scheme that is based on bilinear pairing. A good example of bilinear pairing is weil pairing [34]. In this chapter, we will explain how a weil pairing can be implemented using elliptic curve. We start with the brief introduction of elliptic curve.

3.1 Elliptic Curve

Generally, the elliptic curve [51] is the equation of two variable. The general form of elliptic curve is the generalized Weiestrass equation [46] as given below.

$$y^2 + axy + by = x^3 + cx^2 + dx + e \quad \text{where} \;\; a, b, c, d, e \;\; \text{are constants} \tag{3.1}$$

In this thesis, we used the simplified form of Weiestrass equation for elliptic curve as mentioned below:

$$y^2 = x^3 + ax + b \tag{3.2}$$

where a and b are constants. Further the variables x, y together with the elements a and b are the elements of some field $\mathbb{F}$ such as field of real numbers $\mathbb{R}$, the field of complex numbers $\mathbb{C}$, or any other finite field $\mathbb{F}_p(= \mathbb{Z}_p)$ or the field extension

$\mathbb{F}_{p^k}$ (p is prime and $k \in \mathbb{Z}$). The set of all points satisfying equation (3.2) will be denoted by $E_{\mathbb{F}}(a, b)$.

Although, it is very difficult to plot elliptic curve in most of the field. So, we give examples of two elliptic curves over the field of real numbers. The first one is $E_{\mathbb{R}}(0, 1)$ and its graph is Figure 3.1. The second is $E_{\mathbb{R}}(-1, 0)$ shown in Figure 3.2.

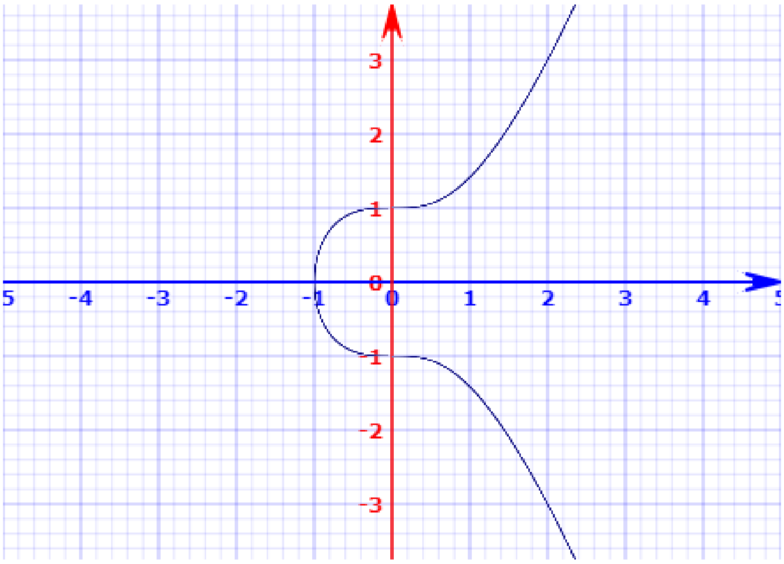

FIGURE 3.1: Elliptic Curve: $y^2 = x^3 + 1$

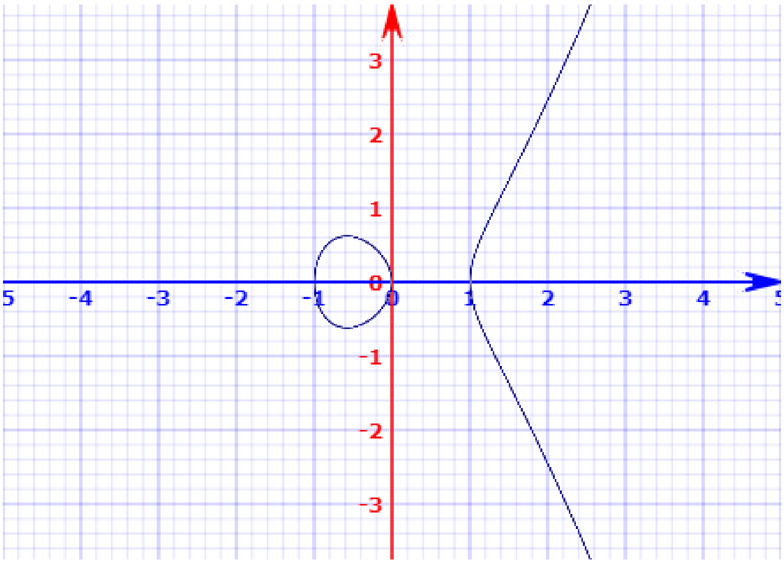

FIGURE 3.2: Elliptic Curve: $y^2 = x^3 - x$

It can be shown that a group can be defined on the set $E_{\mathbb{F}}(a,b)$ for any specific value of a, b in equation (3.2), provided that the following condition is met.

$$4a^3 + 27b^2 \neq 0 \tag{3.3}$$

It is easy to define the group properties over $\mathbb{Q}$ and $\mathbb{R}$ but the problem is that it is slower and inaccurate. So, from cryptographic point of view we are mainly interested in elliptic curve over finite field.

3.1.1 Addition of Points on Elliptic Curve:

Let $A = (x_1, y_1)$ and $B(x_2, y_2)$ be the two points on elliptic curve $E_{\mathbb{F}}(a,b)$ given by equation $y^2 = x^3 + ax + b$. When we add two points $A = (x_1, y_1)$ and $B = (x_2, y_2)$ we draw a line from A to B between them C' be the intersection point. The reflection of $C'(x_3, -y_3)$ is $C(x_3, y_3)$ through the x-axis. So, the addition of points A and B is as follow:

$$A + B = C \tag{3.4}$$

Assume that $A \neq B$. The slope s of line AB is $s = \dfrac{y_2 - y_1}{x_2 - x_1}$. Let $x_1 \neq x_2$ The equation of line AB is $y = sx + y_0$ which implies that

$$y_0 = y - sx = y_1 - sx_1 = y_2 - sx_2$$

Using equation of line and putting value of y in equation (3.2) we get

$$\begin{aligned}(sx + y_0)^2 &= x^3 + ax + b \\ s^2x^2 + y_0^2 + 2sxy_0 &= x^3 + ax + b \end{aligned} \tag{3.5}$$

Now, we already know that x_1 and x_2 be the solution of the equation 3.2. Now we can find the third point x_3 where line meet the curve. Therefore, following cubic equation will be the solution of equation (3.2).

$$(x - x_1)(x - x_2)(x - x_3) = 0$$

Multiplying and rearranging the terms:

$$x^3 + x^2(-x_2 - x_3 - x_1) + x(x_1x_2 - x_2x_3 + x_1x_3) - x_3x_2x_1 = 0 \tag{3.6}$$

Comparing the coefficient of x^2 of 3.5 and 3.6, We get

$$x_2 + x_3 + x_1 = s^2$$

Hence the required point are

$$x_3 = s^2 - x_2 - x_1$$

Now, we can compute y_3 by taking equation of straight line as.

$$\begin{aligned} y_3 &= sx_3 + y_0 \\ -y_3 &= -(sx_3 + y_0) \end{aligned}$$

Now we obtain the points of $C = (x_3, y_3)$ is

$$\begin{aligned} x_3 &= s^2 - x_2 - x_1 \\ y_3 &= -(sx_3 + y_0) \end{aligned}$$

Now consider the case when $A = B = (x_1, y_1)$ so we need to find the equation of tangent line. For finding slope,

$$\begin{aligned} 2y\frac{dy}{dx} &= 3x^2 + a \\ \frac{dy}{dx} &= \frac{3x^2 + a}{2y} \end{aligned}$$

For point A the formula of slope is

$$\frac{dy}{dx} = \frac{3x_1^2 + a}{2y_1}$$

Similarly, one can find the formula of point x_3, y_3 using the above procedure:

$$\begin{aligned} x_3 &= s^2 - 2x_1 \\ y_3 &= -(sx_3 + y_0) \end{aligned} \tag{3.7}$$

Another case we considered in curve $y^2 = x^3 + ax + b$, if two point are the reflection of each other the vertical line will not touch the curve the only solution of the point is line at infinity denoted by $\mathcal{O}$.

3.1.2 Elliptic Curve over finite field

Let $E_{\mathbb{F}}(a, b)$ be the elliptic curve defined over finite field $\mathbb{F}_p$. Therefore, there are finitely many pairs (x, y) with x, $y \in \mathbb{F}_p$, for which the group $E_{\mathbb{F}}(a, b)$ is finite. We take curve of the form.

$$y^2 = x^3 + ax + b \mod p \quad \text{where} \quad 4a^3 + 27b^2 \mod p \neq 0$$

where p is prime number and a and b are the elements of $\mathbb{F}_p = \mathbb{Z}_p$. Now let's discuss an example of elliptic curve over finite field.

Example 3.1.1 Choosing $a = 0$ and $b = 1$ and the field $\mathbb{F}_{11} = \mathbb{Z}_{11}$ in equation (3.8) we get

$$y^2 = x^3 + 1 \mod 11 \tag{3.8}$$

Taking $x = 0 \in \mathbb{Z}_{11}$ in equation (3.8) becomes $y^2 = 1 \mod 11$. Its solution gives $y = 1$ and $y = 10$. So, two points against $x = 0$ are $(0, 1)$ and $(0, 10)$. Again the value $x = 1$, gives

$$y^2 = 2 \mod p$$

which has no solution in $\mathbb{Z}_{11}$. Similarly, for other values of x in $\mathbb{Z}_{11}$ have $x = 2, 3, \ldots 10$ the respective points on the $E_{11}(0, 1)$ are given in Table 3.1.

There are 11 points lying on elliptic curve and when identity $\mathcal{O}$ is added the total becomes twelve points in elliptic curve. Therefore, order is $n = 12$.

x	y^2	$y_{1,2}$	$E_{11}(0,1)$	$E_{11}(0,1)$
0	1	(1, 10)	(0, 1)	(0, 10)
1	2	–	–	–
2	9	(3,8)	(2,3)	(2,8)
3	6	–	–	–
4	10	–	–	–
5	5	(4,7)	(5,4)	(5,7)
6	8	–	–	–
7	3	(5,6)	(7,5)	(7,6)
8	7	–	–	–
9	4	(2,9)	(9,2)	(9,9)
10	0	0	(10, 0)	–

TABLE 3.1: **Elliptic Curve over** $\mathbb{Z}_{11}$

The Figure 3.3 shows the discrete and finite points of elliptic curve which are defined over finite field $\mathbb{Z}_{11}$.

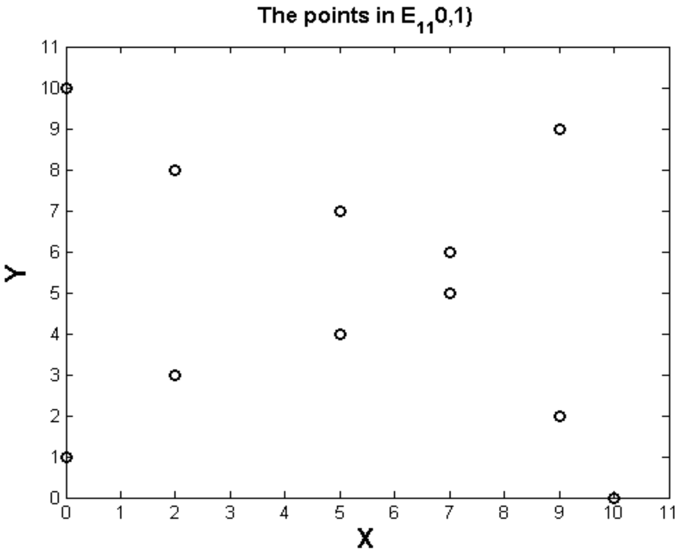

FIGURE 3.3: **Elliptic Curve:** $y^2 = x^3 + 1$ over $\mathbb{Z}_{11}$

Addition of point on elliptic curve on finite field can defined using the same procedure as discussed in Section3.1.1 with all arithmetic operation performed in modulus. This procedure for addition of points on elliptic curve over finite fields $\mathbb{F}$ is summarize by the following algorithm.

Algorithm 3.1.2 (Addition Algorithm)

Input: $A(x_1, y_1)$, $B(x_2, y_2)$, a, b is points on elliptic curve.

Output: $C(x_3, y_3)$

Following are the steps.

1. If $A = \mathcal{O}$ then set point $C = B$
2. If $B = \mathcal{O}$ then set point $C = A$
3. If $A \neq B$ but $x_1 = x_2$. Then $A + B = \mathcal{O}$ where $\mathcal{O}$ is point of infinity
4. If $A = B$ then slope is equal to

   $$s = \frac{3x_1^2 + a}{2y_1} \mod p = (3x_1^2 + a)(2y_1)^{-1} \mod p$$

   Note that taking inverse of denominator term by using extended euclidean inverse its program is mentioned in Appendix.
5. If $A = B$ but $y_1 \neq \mathcal{O}$ then co-ordinates of C are $x_3 = s^2 - 2x_1 \mod p$
   $y_3 = s(x_1 - x_2) - y_2 \mod p$ where slope of line AB is
   $s = (3x_1^2 + a)(2y_1)^{-1} \mod p$
6. If $A = B$ and $y_1 = 0$. It means that point is adding to itself also called point doubling. One can write it as $2 * A = A + A$ then $A + B = \mathcal{O}$ where $\mathcal{O}$ is point of infinity.
7. If $A \neq B$ and $x_1 \neq x_2$ then points of C are $x_3 = s^2 - x_1 - x_2 \mod p$ and $y_3 = x_2 - x_3 - y_2 \mod p$ where slope of line AB is

   $$s = \frac{y_2 - y_1}{x_2 - x_1} \mod p = (y_2 - y_1)(x_2 - x_1)^{-1} \mod p$$

Now, we define the group structure of elliptic curve as:

1. The point at infinity $\mathcal{O}$ is the additive identity of elliptic curve. That is, for any point A on elliptic curve, we have $A + \mathcal{O} = \mathcal{O} + A = A$

2. For any point A on elliptic curve the negative of A is the point with same x-coordinate and negative y-coordinate, that is, if $A = (x, y)$ then $-A = (x, -y)$ satisfies $A + (-A) = \mathcal{O}$. Note that A an $(-A)$ form a vertical line.

3. For any point A and B on elliptic curve having different x-coordinate. To add A and B draw straight line between them and we will find the third intersection point C'. To define group structure we need to define $A+B = C$ where C is the reflection point of C'. Similarly, the line from A to B is same as line from B to A that give the proof elliptic curve satisfies commutative property. Mathematically, it is written as $A + B = B + A$

4. Let $A(x_1, y_1)$, $B(x_2, y_2)$ and $C(x_3, y_3)$ be the points on elliptic curve, the associative law hold states that $(A + B) + C = A + (B + C)$

Now, we will give an example of addition of two points on elliptic curve defined over finite field.

Example 3.1.3 Using Example 3.1.1 we take two points $A(2, 3)$ and $B(5, 4)$ now we calculate $A + B$ on $E_{11}(0, 1)$ as $A \neq B$. Then slope of AB is defined as

$$
\begin{aligned}
s &= \frac{y_2 - y_1}{x_2 - x_1} \mod 11 \\
s &= \frac{4 - 3}{5 - 2} \mod 11 \\
s &= \frac{1}{3} \\
s &= 3^{-1} \mod 11
\end{aligned}
$$

Using the extended euclidean inverse 2.5.14 for modular inverses we get

$$s = 4 \mod 11$$

The formula for finding points is $x_3 = s^2 - x_1 - x_2$ and $y_3 = x_2 - x_3 - y_2$. Putting value of s, A and B, we get

$$x_3 = s^2 - x_1 - x_2 \mod p$$

$$x_3 = 4^2 - 2 - 5 \mod 11$$

$$x_3 = 9 \mod 11$$

$$y_0 = y_1 - sx_1 \mod p$$

$$y_0 = 3 - 4 \times 2 \mod 11$$

$$y_0 = -5 = 6 \mod 11$$

$$y_3 = -(sx_3 + y_0) \mod 11$$

$$y_3 = -(4 \times 9 + 6) = -9 \mod 11$$

$$y_3 = 2$$

Therefore, $A + B = (9, 2) \in E_{11}(0, 1)$ as shown in Table 3.1

Using ApCoCoA [1] program we can add points on elliptic curve is define in Appendix. With the help of program we generate the cayley table of group of points on elliptic curve.
Using Table 3.2 it is easy to verify all abelian group properties of the elliptic curve $E_{11}(0, 1)$.

+	(0,1)	(0,10)	(2,3)	(2,8)	(5,4)	(5,7)	(7,5)	(7,6)	(9,2)	(9,9)	(10,0)
(0,1)	(0,10)	$\mathcal{O}$	(10,0)	(2,3)	(9,9)	(7,6)	(5,4)	(9,2)	(5,7)	(7,5)	(2,8)
(0,10)	$\mathcal{O}$	(0,1)	(2,8)	(10,0)	(7,5)	(9,2)	(9,9)	(5,7)	(7,6)	(5,4)	(2,3)
(2,3)	(10,0)	(2,8)	(0,1)	$\mathcal{O}$	(9,2)	(7,5)	(7,6)	(5,4)	(9,9)	(5,7)	(0,10)
(2,8)	(2,3)	(10,0)	$\mathcal{O}$	(0,10)	(7,6)	(9,9)	(5,7)	(7,5)	(5,4)	(9,2)	(0,1)
(5,4)	(9,9)	(7,5)	(9,2)	(7,6)	(10,0)	$\mathcal{O}$	(2,3)	(0,1)	(0,10)	(2,8)	(5,7)
(5,7)	(7,6)	(9,2)	(7,5)	(9,9)	$\mathcal{O}$	(10,0)	(0,10)	(2,8)	(2,3)	(0,1)	(5,4)
(7,5)	(5,4)	(9,9)	(7,6)	(5,7)	(2,3)	(0,10)	(2,8)	$\mathcal{O}$	(0,1)	(10,0)	(9,2)
(7,6)	(9,2)	(5,7)	(5,4)	(7,5)	(0,1)	(2,8)	$\mathcal{O}$	(2,3)	(10,0)	(0,10)	(9,9)
(9,2)	(5,7)	(7,6)	(9,9)	(5,4)	(0,10)	(2,3)	(0,1)	(10,0)	(2,8)	$\mathcal{O}$	(7,5)
(9,9)	(7,5)	(5,4)	(5,7)	(9,2)	(2,8)	(0,1)	(10,0)	(0,10)	$\mathcal{O}$	(2,3)	(7,6)
(10,0)	(2,8)	(2,3)	(0,10)	(0,1)	(5,7)	(5,4)	(9,2)	(9,9)	(7,5)	(7,6)	$\mathcal{O}$

TABLE 3.2: Addition Cayley Table of Elliptic Curve over $\mathbb{F}_{11}$

3.1.3 Scalar multiplication and order of point

Let $E_{\mathbb{F}}(a,b)$ be the elliptic curve the point multiplication of any point A with some scalar $n \in \mathbb{Z}$ is defined as $nA = \underbrace{A + A + \ldots A}_{n}$. The natural question arises that how many times we add A to itself to get point of infinity $\mathcal{O}$. For that purpose we need to define the order of A.

The order of A is the defined as $nA = \mathcal{O}$ where $n \in \mathbb{N}$. If there exist no integer that A is of infinite order. But when are dealing with finite field order is finite. When we say A is of order 2 then it means that $2A = \mathcal{O}$ and easy to calculate. For example the above cayley Table 3.2 shows that $2(5,4) = \mathcal{O}$.

3.1.4 Divisors

Let $E_{\mathbb{F}}(a,b)$ be the elliptic curve defined over finite field $\mathbb{F}_p$, the divisor D on elliptic curve is the linear combination $m_1A_1 + m_2A_2 + \cdots + m_nA_n$ of distinct points A on $E_{\mathbb{F}}(a,b)$ with integral coefficients $m_1, m_2, \ldots, m_n \in \mathbb{Z}$ and $n \in \mathbb{N}$ [3]

$$D = \sum_{A_i \in E_{\mathbb{F}}(a,b)} m_i(A_i) \quad 1 \leq i \leq n$$

Where finitely many m_i are zero.
The degree of divisor D is the sum of coefficient of m_i.

$$deg(D) = \sum_{A_i \in E_{\mathbb{F}}(a,b)} m_i \quad 1 \leq i \leq n$$

The order of divisor D is the integer $ord_p(D) = m_i$
The sum of divisor D simply uses the group law on $E_{\mathbb{F}}(a,b)$ to add the point A to itself m_n times.

$$sum(D) = \sum_{A_i \in E_{\mathbb{F}}(a,b)} m_iA \quad 1 \leq i \leq n$$

To understand the concept of divisor we need to review some definitions that are used in divisors.

Definition 3.1.4 (Rational Function)

The rational function defined over $E_{\mathbb{F}}(a,b)$ as:

$$f = \frac{p(x)}{q(x)}$$

Where $p(x)$ and $q(x)$ are polynomials. Let A be the point on elliptic curve $E_{\mathbb{F}}(a,b)$, if $f(A) = 0$ then f is said to have zero at A and if f is undefined on A then f is said to have pole at A.

The divisor of rational function f is denoted by $div(f)$. Let we have the rational function as follow:

$$f(x) = \frac{a(x-a_0)^{c_0}(x-a_1)^{c_1}(x-a_2)^{c_2}\dots(x-a_n)^{c_n}}{b(x-b_o)^{d_0}(x-b_1)^{d_1}(x-b_2)^{d_2}\dots(x-b_m)^{d_m}} \tag{3.9}$$

Note that multiplicity of a_0 is c_0 and that of b_0 is d_0 and so on. So, the divisor of rational function (3.9) together with their multiplicity are written as

$$div(f) = c_0(a_0) + c_1(a_1) + \dots c_n(a_n) - d_0(b_0) - d_1(b_1) - \dots - d_m(b_m)$$

For the divisors of elliptic curve, we have following result.

Theorem 3.1.5 Let $E_{\mathbb{F}}(a,b)$ be the elliptic curve and D be the divisor on $E_{\mathbb{F}}(a,b)$ with $deg(D) = 0$. Then $\exists$ a function f on $E_{\mathbb{F}}(a,b)$ with $div(f) = D$ if and only if $sum(D) = \mathcal{O}$. For proof see [51].

Example 3.1.6 We find the divisor [3] of elliptic curve as defined above: $y^2 = x^3 + 1$ over the field $\mathbb{F}_{11}$. Let

$$D = (5,7) + (10,0) + (7,5) + (2,8) - 4(\mathcal{O}) \tag{3.10}$$

As $deg(D) = 0$ and using addition Algorithm 3.1.2 it is shown easily that $sum(D) = \mathcal{O}$. Therefore, D is the divisor of function. Now we have to find out the rational functions of divisor. Firstly we have to find the line passing through the point

$(10, 0)$ and $(5, 7)$. General equation of line is give below

$$y - y_1 = s(x - x_1)$$

Where s is the slope between the points

$$s = \frac{y_2 - y_1}{x_2 - x_1}$$

Putting values, we get

$$\begin{aligned} s &= \frac{7-0}{5-10} \mod 11 \\ s &= \frac{7}{-5} = (7) \times (-5)^{-1} = 3 \mod 11 \end{aligned}$$

Putting values in equation of line, we get

$$\begin{aligned} y - 7 &= 3(x - 5) \\ y - 7 &= 3x - 15 \\ y - 3x + 8 &= 0 \end{aligned}$$

$$div(y - 3x + 8) = (10, 0) + 2(5, 7) - 3(\mathcal{O}) \tag{3.11}$$

Where $(5, 7)$ is double of $(10, 0)$. The vertical line through $(5, 7)$ is $y - 5 = 0$. Therefore, the divisor is:

$$div(y - 5) = (5, 7) + (5, -7) - 2(\mathcal{O}) \tag{3.12}$$

Subtracting (3.11) and (3.12), we get

$$div(y - 3x + 8) - div(y - 5) = ((10, 0) + 2(5, 7) - 3(\mathcal{O})) - ((5, 7) + (5, -7) - 2(\mathcal{O}))$$

It can written as

$$\begin{aligned}\frac{div(y-3x+8)}{div(y-5)} &= (10,0)+(5,7)-(5,-7)-(\mathcal{O})\\ \frac{div(y-3x+8)}{div(y-5)} + (5,-7)+(\mathcal{O}) &= (10,0)+(5,7)\end{aligned}$$

Putting value in (3.10)

$$D = div\left(\frac{y-3x+8}{y-5}\right) + (7,5)+(2,8)+(5,-7)-3(\mathcal{O})$$

Similarly

$$(7,5)+(2,8) = (5,7)+div\left(\frac{y+5x+4}{y-5}\right)+(\mathcal{O})$$

$$D = (5,-7)+div\left(\frac{y-3x+8}{y-5}+(5,7)\right)+div\left(\frac{y+5x+4}{y-5}-2(\mathcal{O})\right)$$

We have already calculate the vertical line through (5,7). Therefore

$$\begin{aligned}D &= div(y-5)+div\left(\frac{y-3x+8}{y-5}\right)+div\left(\frac{y+5x+4}{y-5}\right)\\ D &= \left(\frac{(y-3x+8)(y+5x+4)}{(y-5)^2}\right)\end{aligned}$$

Hence, the divisor of $E_{\mathbb{F}}(a,b)$ is:

$$D = \frac{(y-3x+8)(y+5x+4)}{(y-5)}$$

3.2 Weil pairing on elliptic curve

The goal of this section is to introduced the concept of weil pairing [34]. For this we start with the concept of bilinear pairing that has been discussed in Section 2.5.1. Let $A, B \in E_{\mathbb{F}}(a,b)$ and let f_A and f_B be two rational functions about point A and B respectively, having divisors

$$div(f_A) = n(A) - n(\mathcal{O})$$
$$div(f_B) = n(B) - n(\mathcal{O})$$

Where n is order of point A and B. Here, divisor $div(f_A)$ represents that the rational functions f_A become zero when we put point A or any multiple of A and it will gives infinity when we put point of infinity. Similarly, f_B become zero at point B or any multiple of B and pole at point of infinity.

The weil pairing is the $\phi_n : E_{\mathbb{F}}(a,b) \times E_{\mathbb{F}}(a,b) \longrightarrow \mathbb{F}$:

$$\phi_n(A,B) = \frac{\frac{f_A(B+C)}{f_A(C)}}{\frac{f_B(A-C)}{f_B(-C)}} \tag{3.13}$$

Where $C \notin \{\mathcal{O}, A, -B, A-B\}$ and n is the order of point of elliptic curve.

Now, we will discuss that why point C is not equal to $\{\mathcal{O}, A, -B, A-B\}$. Following are the reasons for C.

- If we will put $C = \mathcal{O}$ in equation (3.13) then it will becomes

$$\phi_n(A,B) = \frac{\frac{f_A(B+C)}{f_A(\mathcal{O})}}{\frac{f_B(A-C)}{f_B(\mathcal{O})}}$$

  then rational function f_A, f_B becomes infinity therefore $\phi_n(A,B) = \infty$.

- If $C = A$ then equation (3.13) becomes

$$\phi_n(A,B) = \frac{\frac{f_A(B+A)}{f_A(A)}}{\frac{f_B(A-A)}{f_B(-A)}}$$

$$\phi_n(A,B) = \frac{\frac{f_A(B+A)}{f_A(A)}}{\frac{f_B(\mathcal{O})}{f_B(-A)}}$$

As we calculate the rational function f_A about point A so by putting point A in f_A then it will become zero and first fraction becomes infinity and also $f_B(A-A) = f_B(\mathcal{O})$ $(A+(-A) = \mathcal{O})$ it gives infinity. so weil pairing between A and B becomes infinity i.e. $\phi_n(A,B) = \infty$.

- Similarly, When $C = -B$, weil pairing is equal to

$$\phi_n(A,B) = \frac{\frac{f_A(\mathcal{O})}{f_A(-B)}}{\frac{f_B(A+B)}{f_B(B)}}$$

Therefore, denominator becomes zero and numerator is equal to zero.

- When $C = A - B$ then equation (3.13) becomes

$$\phi_n(A,B) = \frac{\frac{f_A(B+(A-B))}{f_A(A-B)}}{\frac{f_B(A-(A-B))}{f_B(-(A-B))}}$$

$$\phi_n(A,B) = \frac{\frac{f_A(A)}{f_A(A-B)}}{\frac{f_B(B)}{f_B(B-A)}}$$

The rational functions about point A and B are zero i.e. $f_A = 0$, $f_B = 0$ it will give singularity, but $f_A(A-B), f_B(B-A)$ are not equal to zero. Therefore, numerator of weil pairing is equal to zero and denominator equal to infinity. That's why weil pairing equal to infinity i.e. $\phi_n(A,B) = \infty$

3.2.1 Properties of Weil Pairing

The weil pairing has following properties [13]:

1. $\phi_n(A,B)$ is independent of choice of C

2. $\phi_n(A,B)^n$ is nth root of unity.

3. $\phi_n(A,B)$ is bilinear; that is

$$\phi_n(A+B,C)=\phi_n(A,C)\phi_n(B,C)$$
$$\phi_n(A,B+C)=\phi_n(A,B)\phi_n(A,C)$$

4. $\phi_n(A,A)=1\ \forall A\in E_{\mathbb{F}}(a,b) \implies \phi_n(A,B)=\phi_n(B,A)^{-1} \quad \forall A,B\in E_{\mathbb{F}}(a,b)$
   It can be proved that let $1=\phi_n(A+B,A+B)$ by applying third property of weil pairing we have $1=\phi_n(A,A)\phi_n(A,B)\phi_n(B,A)\phi_n(B,B)$
   $1=(1)\phi_n(A,B)\phi_n(B,A)(1)$
   $\phi_n(A,B)=\phi_n(B,A)^{-1}$

5. If $\phi_n(A,B)=1$ then $A=\mathcal{O} \quad \forall B\in E_{\mathbb{F}}(a,b)$

The proof of all these properties are available in [13]

Example 3.2.1 Let elliptic curve be defined as $y^2=x^3+2$ over $\mathbb{F}_p=\mathbb{F}_7$. Let $A=(5,1)$, $B=(6,1)$ be the points of elliptic curve $E_7(0,2)$, both points are of order $n=3$. Now we calculate the weil pairing between the A and B such that

$$\phi_n(A,B)=\phi_3((5,1),(6,1))$$

We will take divisors of A and B, for the natural choice as some authors use $D_{(A)}=(A)-\mathcal{O}$ and $D_{(B)}=(B+C)-(C)$ where C be any point on elliptic curve. Now we will compute the rational function of $D_{(A)}$ and $D_{(B)}$ after that we will calculate weil pairing by dividing the divisor of $D_{(A)}$ by $D_{(B)}$.

Let

$$D_{(5,1)}=((5,1))-\mathcal{O} \text{ and } D_{(6,1)}=((6,1)+(0,4))-((0,4))$$

Adding the points $(6,1)+(0,4)$ by using formula (3.7) we get $(6,1)+(0,4)=(3,1)$. Therefore, the divisor are

$$D_{(5,1)}=((5,1))-\mathcal{O} \text{ and } D_{(6,1)}=((3,1))-((0,4))$$

Now we calculate the rational functions of divisors, calculate the tangent line along $(5,1)$ Using the formula of slope

$$s = \frac{3(x_1)^2 + a}{2(y_1)} \mod 7$$
$$s = \frac{3(5)^2 + 0}{2(1)} \mod 7$$
$$s = 6 \mod 7$$

Therefore, the tangent line along $(5,1)$

$$y - y_1 = s(x - x_1) \mod p$$
$$y - 1 = 6(x - 5) \mod 7x + y + 1 \qquad = 0 \mod 7$$

Similarly, tangent line along $(3,1)$

$$4x + y + 1 = 0$$

Tangent line along $(0,4)$

$$y + 3 = 0$$

After calculation, we get the divisor of rational functions as

$$div(x + y + 1) = 3D_{(5,1)} \text{ and}$$
$$div(4x + y + 1) - div(y + 3) = div(\frac{(4x + y + 1)}{(y + 3)}) = 3D_{(6,1)}$$

Therefore, rational functions are

$$f_{(5,1)} = (x + y + 1), \qquad f_{(6,1)} = \frac{(4x + y + 1)}{(y + 3)}$$

We will compute the rational function $f_{(5,1)}$ by putting points $(3,1)$ and $(0,4)$ in numerator and denominator respectively

$$f_{(5,1)}(D_{(6,1)}) = \frac{(x+y+1)|_{(3,1)}}{(x+y+1)|_{(0,4)}} \mod 7$$
$$f_{(5,1)}(D_{(6,1)}) = \frac{5}{5} \mod 7 = 1 \mod 7$$

Hence, rational function $f_{(6,1)}$ by putting points $(5,1)$ in numerator and denominator

$$f_{(6,1)}(D_{(5,1)}) = \frac{(4x+y+1)|_{(5,1)}}{(y+3)|_{(5,1)}} \mod 7$$
$$f_{(6,1)}(D_{(5,1)}) = \frac{11}{2} \mod 7 = (11) \times (2)^{-1} \mod 7$$
$$f_{(6,1)}(D_{(5,1)}) = 11 \times 4 \mod 7 = 2 \mod 7$$

For point $\mathcal{O}$ the rational function is $f_{(6,1)}(\mathcal{O}) = 1$ because the term y in numerator and denominator represent that when $(x,y) \to \infty$ then ratio between them is 1. Therefore, weil pairing is

$$\phi_3((5,1),(6,1)) = \frac{f_{(5,1)}}{f_{(6,1)}} = \frac{1}{2} \mod 7 = (1) \times (2)^{-1} \mod 7 = 4 \mod 7$$

It is easy to check that 4 is third root of unity $4^3 \mod 7 = 1 \mod 7$

3.2.2 Miller's Algorithm

See in Example 3.2.1 points on an elliptic curve over field of small size is easy but computations are difficult and time taking when we consider large fields, as the process given above is not computer friendly. For that purpose, we use Miller's algorithm [36] to evaluate the rational function and then calculate the weil pairing as well.

Definition 3.2.2 (Miller's function)

Let s be the slope of line AB where $A, B \in E_{\mathbb{F}}(a,b)$. Let say $A = (x_1, y_1)$ and

$B = (x_2, y_2)$ the function $f_{A,B}$ on $E_{\mathbb{F}}(a, b)$ is define as

$$f_{A,B} = \begin{cases} \dfrac{y - y_1 - s(x - x_1)}{x + x_1 + x_2 - s^2} & \text{if } s \neq \infty \\ x - x_1 & \text{if } s = \infty \end{cases} \tag{3.14}$$

The divisor of $f_{A,B}$ is given by $div(f) = (A) + (B) - (A + B) - (\mathcal{O})$

The Miller's algorithm is stated below.

Algorithm 3.2.3 (Miller's algorithm)

Take an **input** $A(x_1, y_1) \in E_{\mathbb{F}}(a, b)$, n which is the order of of point A, and K which is the binary representation of n. Now following are the steps that is use to calculate the rational function about point A here it is denoted by $f_A = f_{A,A}$ and $f_B = f_{B,B}$

```
Set $Q = A$; and $f_{A,A} = 1$;        ▷ $A$ is the point at which Weil pairing is calculated
Set $K$ =binary representation of $n$
for $I = lenght(K) - 1$ To 1 do
    $f_{A,A} = f^2_{A,A} \times f_{Q,Q}$        ▷ Using formula (3.14) function between point $Q$ and $Q$
    $Q = 2Q$        ▷ Doubling the point $Q$
    if  $K[I] = 1$ then
        $f_{A,A} = f_{A,A} \times f_{A,Q}$        ▷ Using formula (3.14) function between point $A$ and $Q$
        $Q = Q + A$        ▷ Adding point $Q$ and $A$
    end if
end for
Return $f_{A,A}$
```

Example 3.2.4 Using the 3.2.1 again calculating the weil pairing by using the miller Algorithm. Let $E_7(0, 2)$ be defined as $y^2 = x^3 + 2$. Let $A = (5, 1)$, $B = (6, 1)$ be the points of elliptic curve $E_7(0, 2)$, both points are of order $n = 3$. Now we calculate the weil pairing between the A and B such that

$$\phi_n(A, B) = \phi_3((5, 1), (6, 1))$$

. Let $C = (0, 4)$ of order 3. Follow the steps of Algorithm we have $A(x_1, y_1) = (5, 1)$, $n = 3$ and $K =$ binary representation of $n = (11)$

Set $Q = A = (5, 1)$ and $f_{A,A} = 1$

Stating the loop

Step 1 When $I = Lenght(K) - 1 = 2 - 1$. This means that loop is executed at one time only. Calculate function between Q and Q by using formula 3.14

$$f_{Q,Q} = \frac{(x + y + 1)}{(x + 2)} \text{ and } f^2_{A,A} = 1.1 = 1$$

So,

$$f_{A,A} = f^2_{A,A} \times f_{Q,Q}$$
$$f_{A,A} = \frac{(x + y + 1)}{(x + 2)}$$

Now doubling point Q by using formula (3.7)

$$Q = 2(5, 1) = (5, 6)$$

Since Ith bit of K is second bit of $K = 1$ (condition on line 6 is true). Now we proceed further and calculate function between Q and A by using formula (3.14)

$$f_{A,Q} = x + 2 \text{ and previously calculated } f_{A,A} \qquad = \frac{(x + y + 1)}{(x + 2)}$$

So,

$$f_{A,A} = f_{A,A} * f_{A,Q}$$
$$f_{A,A} = x + y + 1$$

Now we adding point Q and A. Now the point Q becomes

$$Q = Q + A$$
$$Q = \mathcal{O}$$

Now terminate the loop at that point. Similarly, follow the above step we calculate the rational function about point $B = (6, 1)$

$$f_{B,B} = \frac{(2x + y + 1)}{(x + 1)} \times (2x + 2)$$

Therefore, the weil pairing between A and B by using formula 3.13. Now we will evaluate numerator of $f_{A,A}$ about $B + C = (6, 1) + (0, 4) = (3, 1)$ and denominator of $f_{A,A}$ about $C = (0, 4)$ by using ApCoCoA tool. The results are as follows.

$$\frac{f_{A,A}(B + C)}{f_{A,A}(C)} = \frac{5}{5} \mod 7 = 1 \mod 7 \tag{3.15}$$

Similarly, we compute the numerator of $f_{B,B}$ about $A - C = (5, 1) + (0, -4) = (3, 1)$ and denominator of $f_{B,B}$ about $-C = (0, -4)$ using ApCoCoA tool. The result are as follow:

$$\frac{f_{B,B}(A - C)}{f_{B,B}(-C)} = \frac{1}{4} \mod 7 = (1)(4)^{-1} \mod 7$$
$$\frac{f_{B,B}(A - C)}{f_{B,B}(-C)} = (1)(4)^{-1} \mod 7 = 2 \mod 7 \tag{3.16}$$

Dividing (3.15) by (3.16), we get

$$\frac{\frac{f_{A,A}(B+C)}{f_{A,A}(C)}}{\frac{f_{B,B}(A-C)}{f_{B,B}(-C)}} = \frac{1}{2} \mod 7 = (1)(2)^{-1} \mod 7 = 4 \mod 7$$

Hence $\phi_3((5, 1), (6, 1)) = 4$
It is easy to check that 4 is third root of unity $4^3 \mod 7 = 1 \mod 7$

By using the miller Algorithm we will calculate the weil pairing for large fields see next example.

Example 3.2.5 Let $E_{\mathbb{F}}(a,b)$ be the elliptic curve defined as $y^2 = x^3 + 37x$ over field $\mathbb{F}_{1009}$. Using Miller's algorithm 3.2.3 we calculate weil pairing between two points.

Let $A = (8, 703)$,$B = (49, 20)$ be the points on elliptic curve for other points set $A' = 2A = (417, 952)$,$B' = 3B = (561, 153)$ all these points are of order 7. let us choose $C = (0, 0)$ of order 2. Now first we have to calculate the weil pairing between A and B

Follow the steps of algorithm then we have $A(x_1, y_1) = (8, 703)$, $n = 7$ and $K =$ binary representation of $n = (111)$

Set $Q = A = (8, 703)$ and $f_{A,A} = 1$

Starting the loop:

<u>**Step 1**</u>: When $I = Lenght(K) - 1 = 3 - 1 = 2$

Calculate function between Q and Q by using formula 3.14

$$f_{Q,Q} = \frac{(157x + y + 59)}{(x + 592)} \text{ and } f^2_{A,A} = 1.1 = 1$$

So,

$$f_{A,A} = f^2_{A,A} * f_{Q,Q}$$

It gives

$$f_{A,A} = \frac{(157x + y + 59)}{(x + 592)}$$

Now, the double the point Q we get

$$Q = 2Q = 2(49, 20)$$

$$Q = (417, 952)$$

Since Ith bit of K is second bit of $K = 1$ (condition on line 6 is true). Now we proceed further and calculate function between Q and A by using formula (3.14)

$$f_{A,Q} = \frac{(66x + y + 787)}{(x + 105)} \text{ and previously calculated } f_{A,A} \quad = \frac{(157x + y + 59)}{(x + 592)}$$

So,

$$f_{A,A} = f_{A,A} * f_{A,Q}$$

Multiplying the functions we get.

$$f_{A,A} = \frac{(272x^2 + 223xy + y^2 + 319x + 846y + 19)}{(x^2 + 697x + 611)}$$

Now we adding point Q and A. Now the point Q becomes

$$Q = Q + A$$
$$Q = (417, 952) + (8, 703)$$
$$Q = (904, 920)$$

Step 2:

In second iteration the value of I is decremented as $I = 1$

Calculate function between Q and Q by using formula (3.14).

$$f_{Q,Q} = \frac{(118x + y + 371)}{(x + 1001)} \text{ and value of } f_{A,A} \text{ from first iteration } f_A \quad = \frac{(157x + y + 59)}{(x + 592)}$$

$$f_A = f_A^2 * f_{Q,Q}$$

Squaring the function $f_{A,A}$ and multiplying it with $f_{Q,Q}$

$$\begin{aligned}\text{Numerator of } f_{A,A} &= 244x^5 + 460x^4y + 535x^3y^2 + 992x^2y^3 + 564xy^4 + y^5 + \\ & 839x^4 + 919x^3y + 815x^2y^2 + 452xy^3 + 45y^4 + 49x^3 + \\ & 238x^2y + 157xy^2 + 507y^3 + 677x^2 + 825xy + 10y^2 + \\ & 369x + 889y + 743 \\ \text{Denominator of } f_{A,A} &= x^5 + 377x^4 + 640x^3 + 648x^2 + 905x + 72 \\ f_{A,A} &= \frac{\text{Numerator of } f_{A,A}}{\text{Denominator of } f_{A,A}} \qquad (3.17)\end{aligned}$$

Doubling the point Q we get

$$Q = 2 * Q$$

$$Q = (8, 306)$$

Since Ith bit of K is third bit of $K = 1$ (condition on line 6 is true). Now we proceed further and calculate function between Q and A by using formula (3.14)

$$f_{A,Q} = \frac{(118x + y + 371)}{(x + 1001)}$$

Now, squaring the function $f_{A,A}$ (3.17) and multiplying it by $f_{A,Q}$ i.e $f_{A,A} = (f_{A,A})^2 f_{A,Q}$ we get

$$\begin{aligned}\text{Numerator of } f_{A,A} &= 244x^6 + 460x^5y + 535x^4y^2 + 992x^3y^3 + 564x^2y^4 + xy^5 + \\ & 905x^5 + 266x^4y + 571x^3y^2 + 588x^2y^3 + 578xy^4 + 1001y^5 + \\ & 400x^4 + 958x^3y + 700x^2y^2 + 927xy^3 + 649y^4 + 285x^3 + \\ & 939x^2y + 772xy^2 + 989y^3 + 1007x^2 + 343xy + 929y^2 + \\ & 818x + 960y + 110\end{aligned}$$

$$\text{Denominator of } f_{A,A} = x^5 + 377x^4 + 640x^3 + 648x^2 + 905x + 72$$
$$f_{A,A} = \frac{\text{Numerator of } f_{A,A}}{\text{Denominator of } f_{A,A}}$$

Now we adding point Q and A. Now the point Q becomes

$$Q = Q + A$$
$$Q = \mathcal{O}$$

Now the loop is decremented and value of $I = 0$ which is not true. Here we stop iteration and the when point becomes $Q = \mathcal{O}$.

Similarly, by using the above steps one can calculate miller function using miller Algorithm 3.2.3 about point B denoted by $f_B = f_{B,B}$. Using the ApCoCoA code see Appendix A the value of $f_{B,B}$ is calculated as

$$\begin{aligned}
\text{Numerator of } f_{B,B} &= 316x^6 + 279x^5y + 343x^4y^2 + 911x^3y^3 + 70x^2y^4 + xy^5 + \\
&398x^5 + 663x^4y + 201x^3y^2 + 734x^2y^3 + 103xy^4 + 960y^5 + \\
&110x^4 + 627x^3y + 35x^2y^2 + 274xy^3 + 431y^4 + 405x^3 + \\
&762x^2y + 776xy^2 + 848y^3 + 225x^2 + 814xy + 981y^2 + \\
&406x + 117y + 1003 \\
\text{Denominator of } f_{B,B} &= x^5 + 97x^4 + 190x^3 + 138x^2 + 391x + 142 \\
f_{B,B} &= \frac{\text{Numerator of } f_{B,B}}{\text{Denominator of } f_{B,B}}
\end{aligned}$$

That's all about miller Algorithm, now we calculate the weil pairing about A and B by using formula 3.13. For that purpose, we will evaluate numerator of $f_{A,A} = f_A$ about $B + C$ and denominator of $f_{A,A} = f_A$ about C by using ApCoCoA tool.

The results are as follows.

$$\begin{aligned}\frac{f_A(B+C)}{f_A(C)} &= \frac{-24102900}{2109450} \quad \bmod 1009 \\ \frac{f_A(B+C)}{f_A(C)} &= 739 \quad \bmod 1009 \end{aligned} \tag{3.18}$$

By using the formula of Weil pairing 3.13, we will evaluate numerator of $f_{B,B} = f_B$ about $A - C$ and denominator of $f_{B,B} = f_B$ about $-C$.

Using ApCoCoA tool the result is

$$\begin{aligned}\frac{f_B(A-C)}{f_A(-C)} &= \frac{-475448}{217069054} \quad \bmod 1009 \\ \frac{f_B(A-C)}{f_A(-C)} &= 574 \quad \bmod 1009 \end{aligned} \tag{3.19}$$

Now, dividing (3.18) by (3.19) by calculating the inverse of denominator by using extended euclidean Algorithm 2.5.14 and multiplying the inverse by numerator, then we get the required pairing that is.

$$\phi_n(A,B) = \frac{739}{574} \quad \bmod 1009 = (739)(574)^{-1} \quad \bmod 1009 = 105 \quad \bmod 1009$$

All calculation has been done using ApCoCoA program see Appendix.

It is easy to verify that 105 is the 7th root of unity such that
$105^7 \quad \bmod 1009 = 1 \bmod 1009$
Now by using the above procedure one can calculate the weil pairing between $A' = (417, 952)$ and $B' = (561, 153)$ is

$$\phi_n(A', B') = 394 \quad \bmod 1009$$

Verification of bilinear property on weil pairing

$$\begin{aligned}\phi_n(A', B') = \phi_n(2A, 3B) &= \phi_n(A,B)^{2\times 3} \\ &= \phi_n(A,B)^6 = (105)^6 \quad \bmod 1009 \\ &= 394 \quad \bmod 1009\end{aligned}$$

3.3 Modified Weil Pairing

As weil pairing is an efficient method to calculate pairing between two points on the elliptic curve, but there are some limitations of weil pairing. When we calculate the weil pairing between the same points or or between the multiple of same points $\phi_n(sA, tA)$ where $s, t \in \mathbb{Z}^+$ always equal to 1. For example, let's say Alice, Bob and Charles want to communicate with each other. They selected elliptic curve $y^2 = x^3 + 37x$ over $\mathbb{F}_{1009}$ and point of order 7 (as in Example 3.2.5). If Alice chooses $A = (8, 703)$, Bob chooses $B = (417, 952)$ and Charles chooses the $C = (904, 920)$, then Where $B = 2A$ and $C = 3A$. Pairing between Alice and Bob is by using the ApCoCoA program given in Appendix A

$$\phi_n(A, B) = ((8, 703), (417, 952)) = 1$$

Pairing between Alice and Charles is by using the ApCoCoA program given in Appendix A

$$\phi_n(A, C) = ((8, 703), (904, 920)) = 1$$

Similarly, pairing between Bob and Charles is:

$$\phi_n(B, C) = ((417, 952), (904, 920)) = 1$$

From all above cases, note that $\phi_n(A, A) = 1$ and by using the definition we have,

$$\phi_n = (iA, jA)(A, A)^{ij} = 1^{ij} = 1 \qquad \because i, j \in \mathbb{N}$$

That is, weil pairing between such points always results in a trivial answer. As we discussed earlier, that we need non-degenerate mapping that satisfies the property

$$\phi_n = (A, A) \neq 1$$

So, to solve this issue we use modified form of weil pairing. Modified weil pairing is defined as:

$$\hat{\phi_n}(A, B) = \phi_n(A, \omega(B))$$

Where ω is distortion map [29] on elliptic curve.

Definition 3.3.1 (Distortion Map)

Let $p \geq 3$ be a prime number,$E_{\mathbb{F}}(a, b)$ be the elliptic curve, and A be the point in $E_{\mathbb{F}}(a, b)$. The distortion map is defined as: $\omega : E_{\mathbb{F}}(a, b) \longrightarrow E_{\mathbb{F}}(a, b)$ and satisfies the condition.

- $\omega(mA) = m\omega(A) \qquad \therefore m \in \mathbb{N}$ and $A \in E_{\mathbb{F}}(a, b)$
- $\phi_n(A, \omega(B))$ is the primitive nth root of unity, that is $\phi_n(A, \omega(B))^n = 1$

The distortion mapping satisfies the following axioms:

1. Let $p \geq 3$ be a prime number, $E_{\mathbb{F}}(a, b)$ is the elliptic curve, and A be the point in $E_{\mathbb{F}}(a, b)$. Let A be the point on $E_{\mathbb{F}}(a, b)$ and distortion mapping on $E_{\mathbb{F}}(a, b)$

   $$\omega : E_{\mathbb{F}}(a, b) \longrightarrow E_{\mathbb{F}}(a, b)$$

   If B, B' are the multiples of A

   $$\hat{\phi_n}(B, \omega(B')) = 1 \quad \Leftrightarrow \quad B = \mathcal{O} \text{ or } B' = \mathcal{O} \quad (\mathcal{O} \text{ is the point at infinity})$$

2. Let elliptic curve defined as $y^2 = x^3 + 1$ over $\mathbb{F}_p$. Let β be the primitive third cube root of unity. Where $\beta \neq 1$ and $\beta^3 = 1$. The distortion mapping is defined as $\omega(A) = \omega(x_1, y_1) = (\beta x_1, y_1)$ and $\omega(\mathcal{O}) = \mathcal{O}$.

3. If $A \in E_{\mathbb{F}}(a, b)$ then $\omega(A) \in E_{\mathbb{F}}(a, b)$

4. ω satisfies the addition law that is:

$$\omega(A)+\omega(B)=\omega(A+B) \quad A,\ B\in E_{\mathbb{F}}(a,b) \tag{3.20}$$

For proof of all axioms are available in [3].

Following are the results that define the field in which β lies.

Proposition 3.3.2 Let p be the prime and $p\equiv 2 \bmod 3$, $\mathbb{F}_p$ does not contain primitive third root of unity but $\mathbb{F}_p^2$. [3]

For proof of proposition is available in [3].
As $\beta\neq 1$ but $\beta^3=1$ this implies that $\beta^3-1=0$ which is equal to $(\beta-1)(\beta^2+\beta+1)=0$ but we have the condition that $\beta\neq 1$ then it implies that $\beta^2+\beta+1=0$ that implies $\beta^2=-\beta-1$. From this fact we conclude that we use the extension field $\mathbb{F}_p^2$ having the irreducible polynomials in $\bmod \beta^2+\beta+1$.
Now we give the example of modified weil pairing

Example 3.3.3 Choosing $a=0$ and $b=1$ and $\mathbb{F}=\mathbb{F}_{29}$ in equation (3.8). Let $A=(8,7)\in E_{29}(0,1)$ be the point of order 5. Then by using distortion mapping $\omega(A)=(8\beta,7)$. Using the above procedure that is use to calculate weil pairing (see Example 3.2.5) one can calculate the modified weil pairing by following the steps of miller Algorithm. By using ApCoCoA code defined in Appendix the Modified Weil pairing is between A and $\omega(A)$.

$$\hat{\phi_n}(A,A)=\phi_n(A,\omega(A))$$
$$\phi_n=((8,7),(8\beta,7))=15\beta+10$$

Here we choose $C=(0,28)$ and work on field F_{29}^2
It is easy to verify that $15\beta+10$ is the 5th root of unity such that $15\beta+10^5\equiv 1 \bmod 29$ and the polynomial is reduced in $\bmod \beta^2+\beta+1$

Chapter 4

Identity Based Broadcast Encryption Scheme

In this chapter, we will briefly described the identity based broadcast encryption scheme (IBBE). Then, we will review IBBE scheme introduced by Ming and Wang [37].

4.1 Broadcast Encryption Scheme

The term Broadcast [9] refers to a form of system in which message is transmitted in the form of video or audio contents. For example, FM radio is the first analogue broadcasting system in which audio message is sent through airwaves.

In the view of cryptography, the idea of broadcast encryption is to perform single encryption of message that can be decrypted by multiple recipients. Communication takes place between the center and the set of receivers. Set of receivers are called privileged. The idea of broadcast encryption was first introduced by Fait and Naor [22]. Their proposal is based on symmetric key cryptography. Later, public key broadcast encryption was introduced [20]. The difference is that the public key encryption scheme uses single recipient for communication whereas broadcast scheme uses multiple recipients. More generally, in public key cryptography center provide public key to the recipients who join the system. When center encrypts the message only privileged users can decrypt it. For security issues, no one from outside of subset can decrypt the message. In literature there are many broadcast encryption schemes have been proposed such as [5, 21, 25, 39]. One prominent class of broadcast encryption scheme is the class of identity based broadcast encryption scheme [44].

4.2 Identity Based Broadcast Encryption Scheme

Identity-based encryption schemes are public key cryptosystems that can use any string as a public key of each receiver. If the public key broadcast encryption is identity-based, senders are able to send ciphertexts to any set of receivers who had never engaged any setup procedure with the system. This implies that its public key size does not depend on the number of receivers.

Identity based broadcast encryption scheme (IBBE) is the generalized form of IBE see Section 1.2. In IBE scheme the communication is between two or three parties, but in IBBE scheme there is only one sender and multiple receivers. Similarly, IBBE scheme consists of four algorithms that is used in IBE scheme. Let Ω be the set of receiver and t be the maximum possible size of Ω.

1. **Setup:** Given t be the maximum possible size of set of identities $\Omega = \{ID_1, ID_2, ID_3, \ldots, ID_m\}$. The private key generator (PKG) generate public parameter $\mathcal{PP}$ and master key $\mathcal{K}$ as output. The master key $\mathcal{K}$ are kept secret and $\mathcal{PP}$ include the information of plaintext space $\mathcal{M}'$ and ciphertext space $\mathcal{C}'$ that are made public.

2. **Extract:** Take $\mathcal{PP}$, $\mathcal{K}$ and $ID_i \in \Omega$ as input. PKG generates private key $\mathcal{K}_{ID_i}$ and send it to the corresponding user. Where $i \in \{1, 2, 3 \ldots m\}$

3. **Encryption** (E)**: Input:** $\mathcal{PP}$, message M and set of identities Ω

   Output: The corresponding ciphertext C.

$$E(\Omega, \mathcal{PP}, M) \to C$$

4. **Decryption** (D)**: Input:** $\mathcal{PP}$, $ID_i \in \Omega$ and it's corresponding private key $\mathcal{K}_{ID_i}$

   Output: The corresponding message M.

$$D(\mathcal{PP}, \Omega, ID_i, \mathcal{K}_{ID_i}) \to M$$

4.3 IBBE with group of prime order

In this section, we make the review of IBBE with group of prime order[37] this scheme is introduced by Ming and Wang. The authors proposed bilinear paring in prime order group. For the construction of IBBE, they use the pairing in dual vector space [16] in prime order group. For security purpose, they use dual system encryption scheme proposed by Waters [52]. Now, we start with some basic definitions that are used in this technique.

Definition 4.3.1 (Bilinear Pairing)

Let $\mathbb{G}_1$, $\mathbb{G}_2$ be the two cyclic groups of prime order p and let g be the generator of G_1 the bilinear mapping ϕ is a map $\phi : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ satisfies the following properties:

1. **Bilinear:** Let $u, v \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_p$ then $\phi(u^a, v^b) = \phi(u, v)^{ab}$

2. **Non-Degenerate:** $\phi(g, g) \neq 1$

3. **Computable:** There exist an efficient algorithm to compute the pairing $\phi(u, v)$ for all $u,\ v \in G_1$

For the construction, the authors use the tool called dual-pairing on vector space [16] which is bilinear pairing on vector space. For a vector

$v = (v_1, v_2, \ldots, v_m) \in \mathbb{Z}_p^m$ and $g \in \mathbb{G}_1$ we write g^v for m-tuples of elements of $\mathbb{G}_1$ and $\mathbb{Z}_p^m$ denote the prime field of m-tuples[37].

$$g^v = (g^{v_1}, g^{v_2}, \ldots, g^{v_m}) \tag{4.1}$$

The vector addition and scalar multiplication can also defined as follows in the exponents.

$$\begin{gathered} g^{v+w} = (g^{v_1+w_1}, g^{v_2+w_2}, \ldots, g^{v_m+w_m}) \quad \forall v,\ w \in \mathbb{Z}_p^m \\ g^{cv} = (g^{cv_1}, g^{cv_2}, \ldots, g^{cv_m}) \quad \forall c \in \mathbb{Z}_p^*,\ v \in \mathbb{Z}_p^m \end{gathered}$$

Define the bilinear mapping ϕ_m on m-tuples of $\mathbb{G}_1$ by pairing component wise and multiplying the result in $\mathbb{G}_2$. That is for $v = (v_1, v_2, \ldots, v_m)$ and
$w = (w_1, w_2, \ldots, w_m)$,

$$\begin{aligned} \phi_m(g^v, g^w) &= \phi(g^{v_1}, g^{w_1}) \cdot \phi(g^{v_2}, g^{w_2}) \ldots \phi(g^{v_m}, g^{w_m}) \\ \phi(g,g)^{v_1w_1+v_2w_2+\cdots+v_mw_m} &= \phi(g,g)^{v_1w_1} \cdot \phi(g,g)^{v_2w_2} \cdots \phi(g,g)^{v_mw_m} \\ &= \phi(g,g)^{v \cdot w} \end{aligned}$$

where dot product $\cdot$ is calculated in modulo p. In this paper, they use dual orthonormal basis that is defined as.

Definition 4.3.2 (Dual Orthonormal Bases)
Let $B = \{b_1, b_2, b_3, \ldots b_m\}$ and $B^* = \{b_1^*, b_2^*, b_3^*, \ldots, b_m^*\}$ be the basis of $\mathbb{Z}_p^m$. For any finite value of m, B and B^* are said to be dual orthonormal bases [40]. If it satisfies the following condition.

$$b_i \cdot b_j^* = \begin{cases} 0 \mod p & \text{if } i \neq j \\ r \mod p & \text{if } i = j \end{cases} \tag{4.2}$$

Where r is a random element of $\mathbb{Z}_p$.
The set of pairs of dual orthonormal bases of dimension m is denoted by $Dual(\mathbb{Z}_p^m, r)$

where r is the dot product of the vectors b_i and b_i^* in $\mathbb{Z}_p^*$. That is, $b_i \cdot b_i^* = r$. Choosing a random pair dual orthonormal bases from $Dual(\mathbb{Z}_p^m, r)$ will be denoted by

$$Dual(\mathbb{Z}_p^m, r) \longrightarrow (B, B*).$$

with the dual pairing vector space are equipped with orthonormal subspaces under the pairs ϕ_m. In this way, the notion of subgroup is replaced by that of subspace in the exponents. We get a workable analogue to prime-order subgroups of composite-order groups by using dual pairing vector spaces. A result in [37] roughly states that "If one starts by sampling a random pair of dual orthonormal based and then applies a linear change of basis to a subset of basis vectors (maintaining the orthonormal property), the resulting bases are also, distributed as random pair, independent of change of basis that was applied" .

For the construction, we define that how this scheme change the bases. Thus starts with a pair (B, B^*) of dual orthonormal bases over $\mathbb{Z}_p^m$, let $A \in \mathbb{Z}_p^{m\times m}$ $(m \leq t)$ be an invertible matrix. Let $S_m \subseteq [t]$ be the subset of size m. Let B_m denotes an $t \times m$ matrix over $\mathbb{Z}_p$ whose columns are vectors $\mathbf{b}_i$ such that $i \in S_m$. Thus Order(B_m)=$t \times m$ and Order(A)=$m \times m$ implies that Order($B_m A$)=$t \times m$. Now, B_A is formed by retaining all vectors $\mathbf{b}_i \in B$ for which $i \notin S_m$ and exchanging all other $\mathbf{b}_i$'s $(i \in S_m)$ with the columns of $B_m A$. Similarly, B_A^* is formed by the retaining all the vectors $\mathbf{b}_i^* \in B$ for which $i \notin S_m$ and exchange all other $\mathbf{b}_i$'s$(i \in S_m)$ with the columns of $B_m^*(A^{-1})^T$, where, as above, B_m^* is $t \times m$ matrix whose columns are the vectors $\mathbf{b}_i^* \in B^*$ such that $i \in S_m$ and columns are the vectors $\mathbf{b}_i^* \in B^*$ such that $i \in S_m$ and $(A^{-1})^T$ is the transpose of the matrix A^{-1}. Now, we give the example to explain the procedure.

Example 4.3.3 Let we take $p = 29$, $t = 3$ and $m = 2$ so $\mathbb{Z}_p^t = \mathbb{Z}_{29}^3$. Let $B = \{(1, 2, 3), (2, 0, 1), (3, 1, 0)\}$ be the basis of $\mathbb{Z}_{29}^3$. First of all, we will calculate the orthonormal basis which satisfies the condition (4.2) where we choose $r = 2 \in \mathbb{Z}_{29}^*$ as random number. Let

$$B^* = \{(b_{11}^*, b_{12}^*, b_{13}^*), (b_{21}^*, b_{22}^*, b_{23}^*), (b_{31}^*, b_{32}^*, b_{33}^*)\}$$

be the orthonormal basis. To find the orthonormal basis we multiply the components of B with B^* and make the following system of linear equation.

$$\begin{cases} 1b_{11}^* + 2b_{12}^* + 3b_{13}^* = 2 \\ 2b_{11}^* + 0b_{12}^* + 1b_{13}^* = 0 \\ 3b_{11}^* + 1b_{12}^* + 0b_{13}^* = 0 \end{cases} \tag{4.3}$$

Transforming into the augmented form and reduce by reduce echelon form

$$\begin{bmatrix} b_{11}^* \\ b_{12}^* \\ b_{13}^* \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & | & 2 \\ 2 & 0 & 1 & | & 0 \\ 3 & 1 & 0 & | & 0 \end{bmatrix}$$

Using the following row operation: $(2R_1 - R_2),\ (3R_1 - R_3)$

$$\begin{bmatrix} b_{11}^* \\ b_{12}^* \\ b_{13}^* \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & | & 2 \\ 0 & 4 & 5 & | & 4 \\ 0 & 5 & 9 & | & 6 \end{bmatrix}$$

Now multiplying the second row by inverse of 4 using extended euclidean inverse see Section 2.5.14 i.e. $(4)^{-1} = 22 \mod 29$, we get

$$\begin{bmatrix} b_{11}^* \\ b_{12}^* \\ b_{13}^* \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & | & 2 \\ 0 & 1 & 23 & | & 1 \\ 0 & 5 & 9 & | & 6 \end{bmatrix}$$

Using the following row operations: $5 * R_2 - R_3$

$$\begin{bmatrix} b_{11}^* \\ b_{12}^* \\ b_{13}^* \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & | & 2 \\ 0 & 1 & 23 & | & 1 \\ 0 & 0 & 19 & | & 28 \end{bmatrix} \mod 29$$

Now, multiplying the third row by inverse of 19 i.e $(19)^{-1} = 26$.

$$\begin{bmatrix} b^*_{11} \\ b^*_{12} \\ b^*_{13} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & | & 2 \\ 0 & 1 & 23 & | & 1 \\ 0 & 0 & 1 & | & 3 \end{bmatrix} \mod 29$$

Using the following row operations: $23R_3 - R_2,\ 3R_3 - R_1$

$$\begin{bmatrix} b^*_{11} \\ b^*_{12} \\ b^*_{13} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & | & 2 \\ 0 & 1 & 23 & | & 1 \\ 0 & 0 & 19 & | & 28 \end{bmatrix} \mod 29$$

Multiplying third row by inverse of 19 i.e $(19)^{-1} = 26$ (see Section2.5.14)

$$\begin{bmatrix} b^*_{11} \\ b^*_{12} \\ b^*_{13} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & | & 2 \\ 0 & 1 & 23 & | & 1 \\ 0 & 0 & 1 & | & 3 \end{bmatrix} \mod 29$$

Use the following row operations: $23 \times R_3 - R_2,\ 3 \times R_3 - R_1$.

$$\begin{bmatrix} b^*_{11} \\ b^*_{12} \\ b^*_{13} \end{bmatrix} = \begin{bmatrix} 28 & 27 & 0 & | & 7 \\ 0 & 28 & 0 & | & 10 \\ 0 & 0 & 1 & | & 3 \end{bmatrix} \mod 29$$

Multiplying second row by inverse of 28 i.e $(28)^{-1} = 28$ (see Section2.5.14)

$$\begin{bmatrix} b^*_{11} \\ b^*_{12} \\ b^*_{13} \end{bmatrix} = \begin{bmatrix} 28 & 27 & 0 & | & 7 \\ 0 & 1 & 0 & | & 19 \\ 0 & 0 & 1 & | & 3 \end{bmatrix} \mod 29$$

Use the following row operation: $27 \times R_2 - R_1$ that will change the first row as follow:

$$\begin{bmatrix} b^*_{11} \\ b^*_{12} \\ b^*_{13} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & | & 13 \\ 0 & 1 & 0 & | & 19 \\ 0 & 0 & 1 & | & 3 \end{bmatrix} \mod 29$$

Now the first component of B^* is

$$b^*_{11} = 13 \mod 29 \qquad b^*_{12} = 19 \mod 29 \qquad b^*_{13} = 3 \mod 29$$

Now, by multiplying the second component of orthonormal basis with all component of basis we will get the following system of linear equation.

$$\begin{cases} 1b^*_{21} + 2b^*_{22} + 3b^*_{23} = 0 \\ 2b^*_{21} + 0b^*_{22} + 1b^*_{23} = 2 \\ 3b^*_{21} + 1b^*_{22} + 0b^*_{23} = 0 \end{cases} \tag{4.4}$$

Similarly, by using the above procedure the second component of B^* is

$$b^*_{21} = 19 \mod 29 \qquad b^*_{22} = 1 \mod 29 \qquad b^*_{23} = 22 \mod 29$$

Similarly, the system of equation for third component of B^* is

$$\begin{cases} 1b^*_{31} + 2b^*_{32} + 3b^*_{33} = 0 \\ 2b^*_{31} + 0b^*_{32} + 1b^*_{33} = 0 \\ 3b^*_{31} + 1b^*_{32} + 0b^*_{33} = 2 \end{cases} \tag{4.5}$$

Therefore, we get

$$b^*_{31} = 3 \mod 29 \qquad b^*_{32} = 22 \mod 29 \qquad b^*_{33} = 23 \mod 29$$

Calculated orthonormal bases are as follow:

$$B = \{(1,2,3),(2,0,1),(3,1,0)\} \quad B^* = \{(13,19,3),(19,1,22),(3,22,23)\}$$

For verification we will calculate the dot product between the bases as follow:

$$\begin{array}{lll} (1,2,3)\cdot(13,19,3)=2 & (1,2,3)\cdot(19,1,22)=0 & (1,2,3)\cdot(3,22,23)=0 \\ (2,0,1)\cdot(13,19,3)=0 & (2,0,1)\cdot(19,1,22)=2 & (2,0,1)\cdot(3,22,23)=0 \\ (3,1,0)\cdot(13,19,3)=0 & (3,1,0)\cdot(19,1,22)=0 & (3,1,0)\cdot(3,22,23)=2 \end{array}$$

Now, applying the above procedure of change of bases as follow: Let bases are written in form of vectors as follow:

$$B = \left(\underbrace{\begin{bmatrix}1\\2\\3\end{bmatrix}}_{\mathbf{b}_1}, \underbrace{\begin{bmatrix}2\\0\\1\end{bmatrix}}_{\mathbf{b}_2}, \underbrace{\begin{bmatrix}3\\1\\0\end{bmatrix}}_{\mathbf{b}_3} \right) \qquad B^* = \left(\underbrace{\begin{bmatrix}13\\19\\3\end{bmatrix}}_{\mathbf{b}_1^*}, \underbrace{\begin{bmatrix}19\\1\\22\end{bmatrix}}_{\mathbf{b}_2^*}, \underbrace{\begin{bmatrix}3\\22\\23\end{bmatrix}}_{\mathbf{b}_3^*} \right)$$

Let $t = 3$, $m = 2$ Let A be the invertible matrix of order$(A) = m \times m = 2 \times 2$ and we take $S_m \subset [t] = S_2 \subseteq [3]$ of size $t = 2$. As we take the subset S of order 2 then it means that the possible entries we choose $\{(b_1, b_2), (b_2, b_3), (b_1, b_3)\}$ in the same way for orthonormal basis $\{(b_1^*, b_2^*), (b_2^*, b_3^*), (b_1^*, b_3^*)\}$. Now, we choose vectors (b_1, b_2) and (b_1^*, b_2^*) and apply changes on only these two vectors this make a rectangular matrix $B_m = B_2$ of order $m \times t = 3 \times 2$

$$A = \begin{bmatrix}1 & 0\\2 & 1\end{bmatrix} \in \mathbb{Z}_{29}^{2\times 2}$$

$$B_2 = \begin{bmatrix}1 & 2\\2 & 0\\3 & 1\end{bmatrix}$$

$$B_2A = \begin{bmatrix} 5 & 2 \\ 2 & 0 \\ 5 & 1 \end{bmatrix} \mod 29$$

Now the first two vectors of B is change and equal to $b_1 = (5, 2, 5)$ and $b_2 = (2, 0, 1)$. Therefore, matrix A is applied to change B to B_A. As we take subset of $Order = 2$ only change the first two vectors of B so B_A is formed by retaining vector $\mathbf{b}_3 \in B$ for which $3 \notin S_2$ and we are exchanging all other $\mathbf{b}_i$'s $(i \in S_2)$ with the columns of B_2A where $i = 1, 2$

$$B_A = B_2A = \begin{bmatrix} 5 & 2 & 3 \\ 2 & 0 & 1 \\ 5 & 1 & 0 \end{bmatrix} \mod 29$$

We get new basis as $B_A = \{(5, 2, 5), (2, 0, 1), (3, 1, 0)\}$ which are linearly independent by using method define in 2.5.21

Now we will change the orthonormal basis by using the above procedure B_A^* is formed by multiplying the B_2^* by $(A^{-1})^T$ is the transpose of the matrix A^{-1}. First of all, we will calculate $(A^{-1})^T$.

$$A^{-1} = \begin{bmatrix} 1 & 0 \\ 27 & 0 \end{bmatrix} \mod 29$$

$$(A^{-1})^T = \begin{bmatrix} 1 & 27 \\ 0 & 1 \end{bmatrix} \mod 29$$

$$B_2^* = \begin{bmatrix} 13 & 19 \\ 19 & 1 \\ 3 & 22 \end{bmatrix}$$

$$B_2^*(A^{-1})^T = \begin{bmatrix} 13 & 22 \\ 19 & 21 \\ 3 & 16 \end{bmatrix} \mod 29$$

Therefore, B_A^* is

$$B_A^* = \begin{bmatrix} 13 & 22 & 3 \\ 19 & 21 & 22 \\ 3 & 16 & 23 \end{bmatrix} \mod 29$$

The new orthonormal basis are $B_A^* = \{(13,19,3),(22,21,16),(3,22,23)\}$. Checking the condition of orthonormal by taking dot product.

$$\begin{aligned} &B_{A_1}^* \cdot B_{A_1} = 2, \quad && B_{A_1}^* \cdot B_{A_2} = 0, \quad && B_{A_1}^* \cdot B_{A_3} = 0 \\ &B_{A_2}^* \cdot B_{A_1} = 0, && B_{A_2}^* \cdot B_{A_2} = 2, && B_{A_2}^* \cdot B_{A_3} = 0 \\ &B_{A_3}^* \cdot B_{A_1} = 0, && B_{A_3}^* \cdot B_{A_2} = 0, && B_{A_3}^* \cdot B_{A_3} = 2 \end{aligned}$$

4.4 Construction of IBBE Scheme

IBBE scheme consists of four algorithms. Before discussing the algorithm the authors take some assumptions.

1. **Assumption:** For construction of IBBE scheme, let $m = 6$ and t denotes the maximum number of set of possible users.

2. **Setup:** This algorithm is run by private key generator (PKG) which create the whole IBBE environment. PKG creates master key $\mathcal{K}$ and public parameter $\mathcal{PP}$ for IBBE.

   Input: Security parameter λ and bilinear mapping $\phi : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ where $\mathbb{G}_1$ and $\mathbb{G}_2$ are the cyclic group of order prime p and orthonormal bases $D = (d_1, d_2 \dots, d_6)$, $D^* = (d_1^*, d_2^* \dots, d_6^*) \in \mathbb{Z}_p^m$

   The system chooses randomly $\alpha, \theta, \sigma \in \mathbb{Z}_p^*$.

   Output: Master key $\mathcal{K}$

$$\mathcal{K} = \{g^{\alpha\theta d_1^*}, g^{\theta d_1^*}, g^{\theta d_2^*}, g^{\sigma d_3^*}, g^{\sigma d_4^*}\} \tag{4.6}$$

Where $\mathcal{K}$ is kept secret and public parameters are as follow.

$\mathcal{PP} = \{\mathbb{G}_1, \mathbb{G}_2, g, p, \phi(g,g)^{\alpha\theta d_1 d_1^*}, g^{d_1}, g^{d_2}, g^{d_3}, g^{d_4}\}$

3. **Extract: Input:** Set of identities of size m i.e. $\Omega = \{ID_1, ID_2, \dots, ID_6\}$, $\mathcal{PP}$ and $\mathcal{K}$

   Private Key Generator (PKG) randomly chooses $r_1^1, r_1^2, \dots r_1^6, r_2^1, r_2^2, \dots r_2^6 \in \mathbb{Z}_p^*$

   Output: Two private keys $\mathcal{K}_1$ and $\mathcal{K}_2$ corresponding to any identity $ID_i \in \Omega$ where $1 \leq i \leq m$.

$$\begin{aligned} \mathcal{K}_1 &= g^{\alpha\theta d_1^* + r_1^i ID_i \theta d_1^* - r_1^i \theta d_2^* + r_2^i ID_i \sigma d_3^* - r_2^i \sigma d_4^*} && (4.7) \\ \mathcal{K}_2 &= \begin{cases} g^{(r_1^1 + r_1^2 + \cdots + r_1^{i-1} + r_1^{i+1} + \cdots + r_1^6)(ID_1 + ID_2 + \dots, ID_6)\theta d_1^*}. \\ g^{r_1^i (ID_1 + ID_2 + \cdots + ID_{i-1} + ID_{i+1} + \dots ID_6)\theta d_1^*}. \\ g^{-(r_1^1 + r_1^2 + \cdots + r_1^{i-1} + r_1^{i+1} + \cdots + r_1^6)\theta d_2^*}. \\ g^{(r_2^1 + r_2^2 + \cdots + r_2^{i-1} + r_2^{i+1} + \cdots + r_2^6)(ID_1 + ID_2 + \dots, ID_6)\sigma d_3^*}. \\ g^{r_2^i (ID_1 + ID_2 + \cdots + ID_{i-1} + ID_{i+1} + \cdots + ID_6)\sigma d_3^*}. \\ g^{-(r_2^1 + r_2^2 + \cdots + r_2^{i-1} + r_2^{i+1} + \cdots + r_2^6)\sigma d_4^*} \end{cases} && (4.8) \end{aligned}$$

4. **Encrypt: Input:** Message M, $\mathcal{PP}$ and $ID_i \in \Omega$ where $1 \leq i \leq m$ for which the private keys $\mathcal{K}_1$ and $\mathcal{K}_2$ has calculated.

   Then center chooses randomly $s_1, s_2 \in \mathbb{Z}_p^*$.

   Output: (C_1, C_2)

$$\begin{aligned} C_1 &= M.\phi(g,g)^{\alpha\theta s_1 d_1 \cdot d_1^*} && (4.9) \\ C_2 &= g^{s_1 d_1 + s_1 (ID_1, ID_2, \dots, ID_6) d_2 + s_2 d_3 + s_2 (ID_1, ID_2, \dots, ID_6) d_4} && (4.10) \end{aligned}$$

5. **Decrypt: Input:** $C = (C_1, C_2)$, $\mathcal{PP}$ and private keys $\mathcal{K}_1$, $\mathcal{K}_2$

   Output: Message M.

$$M = C_1 / \phi_6(\mathcal{K}_1 \mathcal{K}_2, C_2) \tag{4.11}$$

ϕ_6 calculates the bilinear pairing on 6-tuples between product of private keys $\mathcal{K}_1.\mathcal{K}_2$ and C_2.

6. **Correctness:** To check C_1, C_2 are valid one can do the following calculations.

$$
\begin{gathered}
\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2) = \phi\big((g^{\alpha\theta d_1^*+r_1^iID_i\theta d_1^*-r_1^i\theta d_2^*+r_2^iID_i\sigma d_3^*-r_2^i\sigma d_4^*}). \\
(g^{(r_1^1+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)(ID_1+...,ID_6)\theta d_1^*}g^{r_1^i(ID_1+\cdots+ID_{i-1}+ID_{i+1}+...ID_6)\theta d_1^*} \\
g^{-(r_1^1+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)\theta d_2^*}g^{(r_2^1+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)(ID_1+...,ID_6)\sigma d_3^*} \\
g^{r_2^i(ID_1+\cdots+ID_{i-1}+ID_{i+1}+\cdots+ID_6)\sigma d_3^*}g^{-(r_2^1+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)\sigma d_4^*}), \\
(g^{s_1d_1+s_1(ID_1,...,ID_6)d_2+s_2d_3+s_2(ID_1,...,ID_6)d_4})\big)
\end{gathered}
$$

Writing the above equation component wise, then we get

$$
\begin{gathered}
\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2) = \phi((g^{\alpha\theta d_1^*+r_1^iID_i\theta d_1^*}g^{-r_1^i\theta d_2^*}g^{+r_2^iID_i\sigma d_3^*}g^{-r_2^i\sigma d_4^*}). \\
(g^{(r_1^1+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)(ID_1+...,ID_6)\theta d_1^*+(r_1^i)(ID_1+\cdots+ID_{i-1}+ID_{i+1}+...ID_6)\theta d_1^*} \\
g^{-(r_1^1+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)\theta d_2^*} \\
g^{(r_2^1+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)(ID_1+...,ID_6)\sigma d_3^*+r_2^i(ID_1+\cdots+ID_{i-1}+ID_{i+1}+\cdots+ID_6)\sigma d_3^*} \\
g^{-(r_2^1+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)\sigma d_4^*}), (g^{s_1d_1+s_1(ID_1,...,ID_6)d_2+s_2d_3+s_2(ID_1,...,ID_6)d_4}))
\end{gathered}
$$

Now, writing them component wise such that the exponent having d_1^* is written with d_1^*

$$
\begin{gathered}
\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2) = \phi((g^{\alpha\theta d_1^*+r_1^iID_i\theta d_1^*} \\
g^{\theta d_1^*((r_1^1+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)(ID_1+...,ID_6)+(r_1^i)(ID_1+\cdots+ID_{i-1}+ID_{i+1}+...ID_6))} \\
g^{-r_1^i\theta d_2^*}g^{-(r_1^1+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)\theta d_2^*}g^{r_2^iID_i\sigma d_3^*} \\
g^{(r_2^1+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)(ID_1+...,ID_6)\sigma d_3^*+r_2^i(ID_1+\cdots+ID_{i-1}+ID_{i+1}+\cdots+ID_6)\sigma d_3^*} \\
g^{-r_2^i\sigma d_4^*}g^{-(r_2^1+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)\sigma d_4^*}), \\
(g^{s_1d_1+s_1(ID_1,...,ID_6)d_2+s_2d_3+s_2(ID_1,...,ID_6)d_4}))
\end{gathered}
$$

Further simplify the terms and multiplying component wise (4.2) by using the law of exponent which is stated as $x^i.x^j = x^{i+j}$.

$$
\begin{aligned}
&\phi_m(\mathcal{K}_1\mathcal{K}_2, C_2) = \phi\{\\
&\quad g^{\theta d_1^*[\alpha+r_1^i(ID_i+(ID_1+\cdots+ID_{i-1}+ID_{i+1}+\ldots ID_6))+(r_1^1 ID_1+\cdots+r_1^{i-1} ID_{i-1}+ID_i+r_1^{i+1} ID_{i+1}+\cdots+r_1^6 ID_6)]}\\
&\quad g^{\theta d_2^*[-r_1^i-(r_1^1+r_1^2+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)]}\\
&\quad g^{\sigma d_3^*[r_2^i(ID_i+(r_2^1 ID_1+\cdots+r_2^{i-1} ID_{i-1}+ID_i+r_2^{i+1} ID_{i+1}+\cdots+r_2^6 ID_6)+(ID_1+\cdots+ID_{i-1}+ID_{i+1}+\cdots+ID_6)]}\\
&\quad g^{\sigma d_4^*[-r_2^i-(r_2^1+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)]},\\
&\quad (g^{s_1 d_1+s_1(ID_1,\ldots,ID_6)d_2+s_2 d_3+s_2(ID_1,\ldots,ID_6)d_4})\}
\end{aligned}
$$

$$
\begin{aligned}
\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2) = \phi((&g^{\theta d_1^*(\alpha+(ID_1+ID_2+\cdots+ID_6)(r_i+((r_1^1+r_1^2+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6))}\\
&g^{\theta d_2^*(-(r_1^1+r_1^2+\cdots+r_1^6))} g^{\sigma d_3^*((ID_1+ID_2+\cdots+ID_6))(r_2^i+(r_2^1+r_2^2+\cdots+r_2^6))}\\
&g^{\sigma d_4^*(-(r_2^1+r_2^2+\cdots+r_2^6))}, (g^{s_1 d_1+s_1(ID_1,ID_2,\ldots,ID_6)d_2+s_2 d_3+s_2(ID_1,ID_2,\ldots,ID_6)d_4}))
\end{aligned}
$$

$$
\begin{aligned}
\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2) = \phi((&g^{\theta d_1^*(\alpha+(ID_1+ID_2+\cdots+ID_6)((r_1^1+r_1^2+\cdots+r_1^6))}\\
&g^{\theta d_2^*(-(r_1^1+r_1^2+\cdots+r_1^6))} g^{\sigma d_3^*((ID_1+ID_2+\cdots+ID_6))((r_2^1+r_2^2+\cdots+r_2^6))}\\
&g^{\sigma d_4^*(-(r_2^1+r_2^2+\cdots+r_2^6))}, (g^{s_1 d_1+s_1(ID_1,ID_2,\ldots,ID_6)d_2+s_2 d_3+s_2(ID_1,ID_2,\ldots,ID_6)d_4}))
\end{aligned}
$$

$$
\begin{aligned}
\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2) = \phi(&g^{\theta d_1^*\alpha+(ID_1+ID_2+\cdots+ID_6)(r_1^1+r_1^2+\cdots+r_1^6)\theta d_1^*}\\
&g^{-(r_1^1+r_1^2+\cdots+r_1^6)\theta d_2^*} g^{(ID_1+ID_2+\cdots+ID_6)(r_2^1+r_2^2+\cdots+r_2^6)\sigma d_3^*}\\
&g^{-(r_2^1+r_2^2+\cdots+r_2^6)\sigma d_4^*}, (g^{s_1 d_1+s_1(ID_1,ID_2,\ldots,ID_6)d_2+s_2 d_3+s_2(ID_1,ID_2,\ldots,ID_6)d_4}))
\end{aligned}
$$

Now, we will calculate the pairing between K_1K_2 and C_2 by using (4.2), we calculate the dot product of exponents dot product of exponents are written as

$$\begin{aligned}
&\alpha\theta s_1 d_1^* d_1 + (r_1^1 + r_1^2 \cdots + r_1^6)(ID_1 + ID_2 + \cdots + ID_6)\theta s_1 d_1^* \cdot d_1 \\
&- (r_1^1 + r_1^2 \cdots + r_1^6)(ID_1 + ID_2 + \cdots + ID_6)\theta s_1 d_2^* \cdot d_2 \\
&+ (r_2^1 + r_2^2 \cdots + r_2^6)(ID_1 + ID_2 + \cdots + ID_6)\sigma s_2 d_3^* \cdot d_3 \\
&- (r_2^1 + r_2^2 \cdots + r_2^6)(ID_1 + ID_2 + \cdots + ID_6)\sigma s_2 d_4^* \cdot d_4
\end{aligned}$$

By using the condition of orthonormal (4.2) the dot product of same bases is equal to r

$$\begin{aligned}
&\alpha\theta s_1 d_1^* d_1 + (r_1^1 + r_1^2 \cdots + r_1^6)(ID_1 + ID_2 + \cdots + ID_6)\theta s_1 r \\
&- (r_1^1 + r_1^2 \cdots + r_1^6)(ID_1 + ID_2 + \cdots + ID_6)\theta s_1 r \\
&+ (r_2^1 + r_2^2 \cdots + r_2^6)(ID_1 + ID_2 + \cdots + ID_6)\sigma s_2 r \\
&- (r_2^1 + r_2^2 \cdots + r_2^6)(ID_1 + ID_2 + \cdots + ID_6)\sigma s_2 r
\end{aligned}$$

After putting random r so some terms are cancel out so exponent are simplified as

$$\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2) = \phi(g, g)^{\alpha\theta s_1 d_1^* d_1}$$

which is the pairing of encryption.

4.5 Security Analysis

- **Subspace assumption:** given $\mathbb{G}_1, \mathbb{G}_2, \phi, p$ and orthonormal bases $B = (b_1, b_2, \ldots, b_m)$, $B^* = (b_1^*, b_2^*, \ldots, b_m^*)$ and pick randomly
  $g \in \mathbb{G}_1, \eta, \beta, \tau_1, \tau_2, \tau_3, \mu_1, \mu_2, \mu_3 \in \mathbb{Z}_p^*$ and $U_1 = g^{\mu_1 b_1 + \mu_2 b_{k+1} + \mu_3 b_{2k+1}}$,$U_2 = g^{\mu_1 b_2 + \mu_2 b_{k+2} + \mu_3 b_{2k+2}} \ldots$
  $U_k = g^{\mu_1 b_k + \mu_2 b_{2k} + \mu_3 b_{3k}}$
  Similarly, for $1 \leq i \leq k$
  $V_i = g^{\tau_1 \eta b_i^* + \tau_2 \beta b_{k+i}^*}$, $W_i = g^{\tau_1 \eta b_i^* + \tau_2 \beta b_{k+i}^* + \tau_3 b_{2k+i}^*}$

Let $\mathcal{D} = (g^{b_1}, g^{b_2}, \dots, g^{2k}, g^{3k+1}, \dots, g^{b_n}$
$, g^{\eta b_1^*}, \dots, g^{\eta b_k^*}, g^{\beta b_{k+1}^*}, \dots, g^{\eta b_{2k}^*}, g^{b_{2k+1}^*}, \dots, g^{b_n^*}, U_1, U_2, \dots, U_k, \mu_3)$
It is very hard to distinguish between $V_1, V_2 \dots, V_k$ and $W_1, W_2 \dots, W_k$

- The security of the scheme relies on decisional linear assumption (DLIN), Let $\mathbb{G}_1$, $\mathbb{G}_2$, ϕ, p and pick randomly $g, f, h \in \mathbb{G}_1$ and take $c_1, c_2, w \in \mathbb{Z}_p$ and compute $T_1 = g^{c_1+c_2}$, $T_2 = g^{c_1+c_2+w}$. It is hard to distinguish between T_1 and T_2.

  The construction of proposed scheme is based on DLIN that utilize subspace assumption. As it is mentioned in [31] that “if the DLIN assumption holds, subspace assumption also holds”.

4.5.1 Chosen Ciphertext Attack

By lemma [37] it is proof that IBBE scheme is secure against the chosen ciphertext attack. IBBE scheme uses dual system encryption scheme that was first introduced by Waters [32, 52] that has been emerged as the useful tool for achieving the full security. In dual system encryption scheme the keys and ciphertext are of two forms. Normal keys and normal ciphertext are used in the real IBBE system. But semi-functional keys and semi-functional ciphertext are not used in real system. They are only use in security proof. Namely, semi-functional and normal, these keys and ciphertext have some properties.

1. Normal keys can decrypt both form of ciphertexts (semi-functional or normal).

2. Semi-functional keys can also decrypt both form of ciphertexts (semi-functional or normal).

3. Normal ciphertext can be decrypted from semi-functional keys.

4. Semi-functional ciphertext can be decrypted from normal keys.

5. When semi-functional key is use to decrypt the semi-functional ciphertext then decryption will fail.

By using the definition, when adversary will chose the ciphertext due to random parameter it is very difficult for him to differentiate between semi-functional ciphertext and normal ciphertext. When he decrypt the semi-functional ciphertext with semi-functional key then decryption will fail. By using this technique the IBBE is secure against the chosen ciphertext attack.

For this IBBE scheme we provide the definition of semi-functional key and ciphertext which are as follow.

1. **Semi-functional Keys** Using the construction of IBBE scheme, PKG construct $(\mathcal{K}_1, \mathcal{K}_2)$ that are the normal keys. Then take the random number $t_5,\ t_6,\ t_5^{'},\ t_6^{'}$ so the semi-functional keys are given below:

$$\mathcal{K}_1^{'} = \mathcal{K}_1.g^{t_5 d_5^* + t_6 d_6^*} \tag{4.12}$$

$$\mathcal{K}_2^{'} = \mathcal{K}_2.g^{t_5^{'} d_5^* + t_6^{'} d_6^*} \tag{4.13}$$

2. **Semi-Functional Ciphertexts:** Firstly, calculate the normal ciphertext (C_1, C_2) from the encrypt algorithm as defined above then calculation of semi-functional cipher text are as follow:

$$C_1^{'} = C_1 = M.\phi(g, g)^{\alpha\theta s_1 d_1 d_1^*} \tag{4.14}$$

$$C_2^{'} = C_2.g^{z_5 d_5 + z_6 d_6} \tag{4.15}$$

Where z_5, z_6 selected randomly from $\mathbb{Z}_p^*$

4.5.2 Chosen Plaintext Attack

For proving that an IBBE scheme is secure against the chosen plaintext attack we proceed through the four lemmas that are defined in [37]. All lemmas using the game between the Algorithm $\mathcal{A}$ which is run by adversary and Algorithm $\mathcal{B}$ which is run by PKG. Note that the all games are probabilistic and depends upon the chance that the adversary $\mathcal{A}$ has guess the correct ciphertext. The first lemma is also divided into two parts if Algorithm $\mathcal{B}$ produced normal ciphertext on the queries on adversary $\mathcal{A}$ then this lemma is called real security game and if $\mathcal{B}$ produce the semi-functional ciphertext on the queries of $\mathcal{A}$ then it is called as Game$_0$. In lemma 2,3 and 4 algorithm $\mathcal{B}$ change p number of private key one by one to semi-functional and in last lemma $\mathcal{B}$ make semi-functional private key at that time the $\mathcal{A}$ decrypt the semi-function ciphertext with semi-functional key so the decryption will fail. Now first of all we summarizing Lemma 1.

Lemma 4.5.1 The Algorithm $\mathcal{B}$ is given

$$\mathcal{D} = (g^{b_1}, g^{b_2}, g^{b_3}, g^{b_4}, g^{\eta b_1^*}, g^{\eta b_2^*}, g^{\eta b_3^*}, g^{\eta b_4^*}, g^{b_5^*}, g^{b_6^*}, U_1, U_2, \mu_3)$$

and T_1 and T_2. The goal of $\mathcal{B}$ is to decide whether $T_1 = g^{\tau_1 \eta b_1^* + \tau_2 \beta b_3^* + \tau_3 b_5^*}$ and $T_2 = g^{\tau_1 \eta b_2^* + \tau_2 \beta b_4^* + \tau_3 b_6^*}$ or $T_1 = g^{\tau_1 \eta b_1^* + \tau_2 \beta b_3^*}$ and $T_2 = g^{\tau_1 \eta b_2^* + \tau_2 \beta b_4^*}$.
The system changes it's basis by choosing the invertible matrix $A \in \mathbb{Z}_p^{2\times 2}$ and define the new orthonormal basis as
$F = (f_1, f_2, f_3, f_4, f_5, f_6)$ and $F^* = (f_1^*, f_2^*, f_3^*, f_4^*, f_5^*, f_6^*)$ as follow.
$f_1 = \eta b_1^*,\ f_2 = \eta b_2^*,\ f_3 = \beta b_3^*\ f_4 = \beta b_4^*,\ f_5 = b_5^*,\ f_6 = b_6^*$
$f_1^* = \eta^{-1} b_1,\ f_2^* = \eta^{-1} b_2,\ f_3^* = \beta^{-1} b_3\ f_4 = \beta^{-1} b_4,\ f_5^* = b_5,\ f_6^* = b_6$
Now we apply matrix A to change the basis matrix to $f_5,\ f_6$ and $(A^{-1})^T$ change the basis matrix to $f_5^*,\ f_6^*$ and $\mathcal{B}$ set original basis $D = F_A$ and $D^* = F_A^*$.
Now the $\mathcal{B}$ chooses some random number $\alpha',\ \theta',\ \sigma'$ and sets $\theta = \theta'\eta,\ \sigma = \sigma'\beta$, and calculate the master key as $\mathcal{K} = \{g^{\alpha b_1 \theta'},\ g^{b_1\theta'}, g^{b_2\theta'}, g^{b_3\sigma'},\ g^{b_4\sigma'}\}$

Now the adversary $\mathcal{A}$ target identity $ID_i \in \Omega$. The $\mathcal{B}$ generate the $\mathcal{K}_1,\ \mathcal{K}_2$ by using extract algorithm and send it to adversary $\mathcal{A}$.
The $\mathcal{A}$ has two outputs M_0 and M_1 and challenge set $\Omega^* = \{ID_1^*, ID_2^*, \ldots ID_m^*$.
Then $\mathcal{B}$ chooses a bit $b \in \{0, 1\}$ and compute the ciphertexts as.

$$C_1 = M_b.\phi(T_1, g^{b_1})^{\alpha\theta'} \text{ and } C_2 = T_1.(T_2)^{ID_1^*, ID_2^*, \ldots, ID_n^*}$$

$\mathcal{A}$ continue to issue the queries on all ID_i but constraint that $ID_i \notin \Omega^*$. At the end, $\mathcal{A}$ guess a bit b' and obtain the correct ciphertext if $b' = b$

If $T_1 = g^{\tau_1 \eta b_1^* + \tau_2 \beta b_3^*}$ and $T_2 = g^{\tau_1 \eta b_2^* + \tau_2 \beta b_4^*}$ (where $\eta, \beta, \tau_1, \tau_2$ are chosen randomly) then $C = (C_1, C_2)$ is the normal ciphertext and if the ciphertext is normal then game is called real security game. If $T_1 = g^{\tau_1 \eta b_1^* + \tau_2 \beta b_3^* + \tau_3 b_5^*}$ and $T_2 = g^{\tau_1 \eta b_2^* + \tau_2 \beta b_4^* + \tau_3 b_6^*}$ then $C = (C_1, C_2)$ is the semi-functional ciphertexts if the ciphertext is semi-functional then it is Game$_0$.

Now, after simulating Game$_0$ proceeding further and summarizing Lemma 2, Lemma 3 and Lemma 4 of [37] in which we change the p (order of group) number of private keys and one by one to semi-functional so in this way the adversary $\mathcal{A}$ cannot decrypt semi-functional ciphertext that is calculated in Game$_0$ with semi-functional key.

Lemma 4.5.2 Proceeding in the same way as above, the algorithm $\mathcal{B}$ is given the same parameters as in lemma 1 and also goal of $\mathcal{B}$ is same as in lemma 1. The system changes it's basis by choosing the invertible matrix $A \in \mathbb{Z}_p^{2\times 2}$ and define the orthonormal basis as $d_1 = b_1,\ d_2 = b_2,\ d_3 = b_3, d_4 = b_4, d_1^* = b_1^*, d_2^* = d_2^*,\ d_3^* = b_3^*,\ d_4 = b_4^*$
Now we apply matrix A to change the basis matrix to $d_5,\ d_6$ and $(A^{-1})^T$ change the basis matrix to $d_5^*,\ d_6^*$ and $\mathcal{B}$ set original basis $D = B_A$ and $D^* = B_A^*$.
Setup generate the master key as $\mathcal{K} = \{g^{\alpha \eta b_1^*}, g^{\eta b_1}, g^{\eta b_2^*}, g^{\beta b_3^*},\ g^{\beta b_4^*}\}$ and adversary chooses $ID_i \in \Omega$ then $\mathcal{B}$ set $(k, m) = (2, 6)$ then following possibilities take place.

- If $i < k$, challenger start the extract algorithm and generate the normal keys for ID_i. As system knows the $g^{b_5^*}$ and $g^{b_6^*}$ where b_5^*, b_6^* are the component of basis. It can also generate semi-functional keys by taking the linear combination of $g^{b_5^*}$ and $g^{b_6^*}$.

- If $i > k$, challenger start the extract algorithm and generate the normal keys for ID_i.

- If $i = k$, $\mathcal{B}$ randomly chooses $r_1^1, r_1^2, \ldots r_1^m, r_2^1, r_2^2, \ldots r_2^m \in \mathbb{Z}_p^*$ then keys are calculated as follow:

$$\mathcal{K}_1 = (g^{\eta b_1^*})^{\alpha} T_1^{ID_i} T_2^{-1}$$

$$\mathcal{K}_2 = \begin{cases} (g^{\eta b_1^*})^{(r_1^1+r_1^2+\ldots r_1^m)(ID_1+ID_2+\ldots,ID_m)}. \\ (g^{\eta b_2^*})^{-(r_1^1+r_1^2+\ldots r_1^m)}. \\ (g^{\beta b_3^*})^{(r_2^1+r_2^2+\ldots r_2^m)(ID_1+ID_2+\ldots,ID_m)}. \\ (g^{\beta b_4^*})^{-(r_2^1+r_2^2+\ldots r_2^m)}.T_1^{(ID_1+ID_2+\ldots,ID_m)} \end{cases}$$

If $T_1 = g^{\tau_1\eta b_1^*+\tau_2\beta b_3^*}$ and $T_2 = g^{\tau_1\eta b_2^*+\tau_2\beta b_4^*}$ (where $\eta, \beta, \tau_1, \tau_2$ are chosen randomly) then $C = (C_1, C_2)$ is the normal keys. If $T_1 = g^{\tau_1\eta b_1^*+\tau_2\beta b_3^*+\tau_3 b_5^*}$ and $T_2 = g^{\tau_1\eta b_2^*+\tau_2\beta b_4^*+\tau_3 b_6^*}$ then $C = (C_1, C_2)$ is the semi-functional keys.

The $\mathcal{A}$ has two outputs M_0 and M_1 and challenge set $\Omega^* = \{ID_1^*, ID_2^*, \ldots ID_m^*$. Then $\mathcal{B}$ chooses a bit $b \in \{0, 1\}$ and compute the semi-functional ciphertexts as.

$$C_1 = M_b.\phi(g^{\eta b_1^*}, U_1)^\alpha \text{ and } C_2 = U_1.(U_2)^{ID_1^*, ID_2^* \ldots ID_m^*}$$

$\mathcal{A}$ continue to issue queries on $ID_i \in \Omega$ and finally end the algorithm with guess a bit b' guesses the ciphertext if $b' = b$.

In lemma 3 and 4 the assumption is $(k, m) = (1, 6)$ and the game is same as above but the only change is that here it take U_1 and calculate the polynomial T_1.

Lemma 4.5.3

The $\mathcal{B}$ is given $\mathcal{D} = (g^{b_1}, g^{b_2}, g^{b_4}, g^{b_5}, g^{b_6}, g^{\eta b_1^*}, g^{\beta b_2^*}, g^{b_3^*}, g^{b_4^*}, g^{b_5^*}, g^{b_6^*},$
$U_1 = g^{\mu_1 b_1+\mu_1 b_2+\mu_1 b_3}, \mu_3)$. The goal of $\mathcal{B}$ is decided whether $T_1 = g^{\tau_1\eta b_1^*+\tau_2\beta b_2^*}$ or
$T_1 = g^{\tau_1\eta b_1^*+\tau_2\beta b_2^*+\tau_3 b_3^*}$

First of all, $\mathcal{B}$ choose the invertible matrix $A \in \mathbb{Z}_p^{2\times 2}$ and define the bases as
$d_1 = b_6^*,\ d_2 = b_3^*,\ d_3 = b_5^*, d_4 = b_4^*, d_5 = b_2^*, d_6 = b_1^*,$
$d_1^* = b_6, d_2^* = b_3, d_3^* = b_5,\ d_4^* = b_4, d_5^* = b_2, d_6^* = b_1$

Now $\mathcal{B}$ chooses $\theta, \sigma, \alpha \in \mathbb{Z}_p^*$ and send the public parameters $\mathcal{PP}$ to $\mathcal{A}$. Now, adversary issues the queries on $ID_i \in \Omega$ and challenger calculates the private keys for every ID_i. $\mathcal{B}$ chooses the random values $r_1^{i'}, t_5', t_6', t_5'', t_6'', r_1^1, r_1^2, \ldots r_1^{i-1}, r_1^{i+1}$
$\ldots, r_1^m, r_2^1, r_2^2, r_2^3 \ldots, r_2^m \in \mathbb{Z}_p^*$ and calculate the keys as follow.

$$\mathcal{K}_1 = (U_1)^{(-\theta r_1^{i'})\theta}(g^{b_6})^{(\alpha+\mu_3 r_1' ID_i)\theta}(g^{b_5})^{r_2^i ID_i\sigma}(g^{b_4})^{-r_2^{i'}\sigma}(g^{b_2})^{t_5'}.(g^{b_1})^{t_6'}$$

$$\mathcal{K}_2 = (g^{b_6})^{(r_1^1, r_1^2, \ldots r_1^{i-1}, r_1^{i+1} \ldots r_1^m)(ID_1, ID_2, \ldots ID_m)\theta}.$$
$$(g^{b_6})^{\mu_3 r_1^{i'}(ID_1, ID_2, \ldots ID_{i-1} + ID_{i+1} \ldots ID_m)\theta}.$$
$$(U_1)^{-(r_1^1, r_1^2, \ldots r_1^{i-1}, r_1^{i+1} \ldots, r_1^m)\theta/\mu_3}.$$
$$(g^{b_5})^{(r_2^2, r_2^3, \ldots r_2^{i-1}, r_2^{i+1} \ldots, r_2^m)(ID_1, ID_2, \ldots ID_m)\sigma}$$
$$(g^{b_5})^{(r_2^i)(ID_1, ID_2, \ldots ID_{i-1} + ID_{i+1} \ldots, ID_m)\sigma}$$
$$(g^{b_6})^{-(r_2^1, r_2^2 \ldots r_2^{i-1}, r_2^{i+1} \ldots, r_2^m)\sigma}.(g^{b_2})^{t_5^{i''}}.(g^{b_1})^{t_6''}$$

The $\mathcal{A}$ has two outputs M_0 and M_1 and challenge set $\Omega^* = \{ID_1^*, ID_2^*, \ldots ID_m^*$. Then $\mathcal{B}$ chooses a bit $b \in \{0,1\}$ and compute the semi-functional ciphertexts choosing $s_1, s_2\mathbb{Z}_p^*$

$$C_1 = M_b.\phi(g^{g,g}, U_1)^{\alpha\theta s_1} \text{ and}$$
$$C_2 = (g^{b_6^*})_1^s (g^{b_3^*})^{s_1(ID_1^*, ID_2^* \ldots, ID_m^*)} (g^{b_5^*})^{s_2} (g^{b_4^*})^{s_2(ID_1^*, ID_2^* \ldots, ID_m^*)} T_1$$

If $T_1 = g^{\tau_1\eta b_1^* + \tau_2\beta b_2^*} = g^{\tau_1\eta b_5^* + \tau_2\beta b_6^*}$ then exponent of T_1 is the linear combination of d_5 and d_6 this make the semi-functional ciphertext. If

$$T_1 = g^{\tau_1\eta b_1^* + \tau_2\beta b_2^* + \tau_3 b_3^*} = g^{\tau_1\eta b_6^* + \tau_2\beta b_5^* + \tau_3 b_2^*}$$

then τ_3 randomize d_2 and making the semi functional ciphertext.

Lemma 4.5.4 The same procedure is follow in Lemma 4 but only that the system change the bases as follow:

$$d_1 = b_3^*, d_2 = b_4^*, d_3 = b_5^*, d_4 = b_6^*, d_5 = b_1^*, d_6 = b_2^*$$
$$d_1^* = b_3, d_2^* = b_4, d_3^* = b_5, d_4^* = b_6, d_5^* = b_1, d_6^* = b_2$$

The $\mathcal{A}$ make queries about $ID_i \in \Omega$ and $\mathcal{B}$ chooses random integers same as above. Therefore, the keys are calculated as follow:

$$\mathcal{K}_1 = (U_1)^{(\alpha' + r_1' ID_i)\theta} (g^{b_4})^{-r_1^{i'}\mu_3\theta} (g^{b_5})^{r_2^i ID_i \sigma} (g^{b_6})^{-r_2^{i'}\sigma} (g^{b_1})^{t_5'}.(g^{b_2})^{t_6'}$$

$$\begin{aligned}
\mathcal{K}_2 = {} & (U_1)^{(r_1^1, r_1^2, \ldots r_1^{i-1}, r_1^{i+1}, \ldots, r_1^m)(ID_1, ID_2, \ldots ID_m)\theta/\mu_3}. \\
& (U_1)^{\mu_3 r_1^{i'}(ID_1, ID_2, \ldots, ID_{i-1}, ID_{i+1}, \ldots ID_m)\theta} 5. \\
& (g^{b_4})^{-(r_1^1, r_1^2, \ldots r_1^{i-1}, r_1^{i+1}, \ldots, r_1^m)\theta}. \\
& (g^{b_5})^{(r_2^1, r_2^2 \ldots r_2^{i-1}, r_2^{i+1}, \ldots, r_2^m)(ID_1, ID_2, \ldots ID_n)\sigma} \\
& (g^{b_5})^{(r_2^i)(ID_1, ID_2, \ldots, ID_{i-1}, ID_{i+1}, \ldots, ID_m)\sigma} \\
& (g^{b_6})^{-(r_2^1, r_2^2 \ldots r_2^{i-1}, r_2^{i+1}, \ldots, r_2^m)\sigma}.(g^{b_1})^{t_5^{''}}.(g^{b_2})^{t_6''}
\end{aligned}$$

The $\mathcal{A}$ output two challenge message M_0, M_1 and challenge set ψ . Then $\mathcal{B}$ choose $b \in \{0, 1\}$,$s_1, s_2, \omega \in \mathbb{Z}_p^*$. therefore it sets the semi-functional ciphertexts of M_b.

$$\begin{aligned}
C_1 &= M_b \phi(g^{b_4}, g^{b_4^*}) \\
C_2 &= (g^{b_3^*})^{s_1}.(g^{b_4^*})^{\omega}.(g^{b_5^*})^{s_2}.(g^{b_6^*})^{s_2(ID_1, ID_2, \ldots ID_n)}.T_1
\end{aligned}$$

$\mathcal{A}$ continue to issue private keys query on ID_i but $\mathcal{A}$ is not allowed to generates queries for $ID_i \in \Omega^*$. $\mathcal{A}$ guess b' and wins the game if $b' = b$.

If $T_1 = g^{\tau_1 \eta b_1^* + \tau_2 \beta b_2^*} = g^{\tau_1 \eta d_5^* + \tau_2 \beta d_6^*}$ then exponent of T_1 is the linear combination of d_5 and d_6 this make the semi-functional ciphertext.

If $T_1 = g^{\tau_1 \eta b_1^* + \tau_2 \beta b_2^* + \tau_3 b_3^*} = g^{\tau_1 \eta d_5^* + \tau_2 \beta d_6^* + \tau_3 d_1}$ then exponent of T_1 is the linear combination of d_5 and d_6 this make the semi-functional ciphertext.

From the all above discussion we conclude that the IBBE scheme is secure against chosen plaintext attack having negligible advantage that adversary gain in real security game.

4.5.3 Analysis and Conclusion

The authors compare the efficiency and security of four schemes[17, 24, 41, 53] with their proposed scheme [37]. They analysis is summarized.

- **Identity-based broadcast encryption with constant size ciphertexts and private keys Scheme:** The system parameters size depend upon the maximum number of receivers t but the private key and ciphertext size are constant. This scheme does not provide full security in group of prime order and uses the D-GDHE assumption see literature [17].

- **Adaptive security in broadcast encryption systems (with short ciphertexts) scheme:** Authors provide three schemes of IBBE, we label them as S_1, S_2, S_3. In S_1 scheme, size of system parameter and private key depend upon the maximum possible size of receivers but having constant size ciphertext. It does not provide the full security in group of prime order and uses decision-BDHE assumption see literature [24]. In S_2 scheme, size of system parameters depend upon the maximum possible size of receivers but having constant size ciphertext and private keys. It does not provide the full security in group of prime order and uses decision-BDHE assumption see literature [**24**]. In S_3 scheme, size of system parameters depend upon the maximum possible size of receivers but having constant size of private key and the execution time of ciphertext grow slowly with the maximum possible size of receivers. It provides the full security in group of prime order and uses decision-BDHE assumption see literature [24].

- **Fully CCA secure identity based broadcast encryption without random oracles scheme:** The size system parameters and ciphertext space are constant but the size of private key depends upon the maximum number of receivers $|\Omega|$. This scheme provide full security in group of prime order and uses D-TBDHE assumption see literature [41].

- **Adaptively secure identity-based broadcast encryption with constant size private keys and ciphertexts from the subgroups scheme:** The size of system parameters depend upon the maximum number of receivers t but the private key and ciphertext size are constant. This scheme provide full security in composite order group using SD assumption see literature [53].

- **Identity Based Broadcast Encryption with Group of Prime Order scheme:** The size of system parameters, private keys and ciphertext are constant. It provides the full security in the group of prime order using decisional linear assumption.

From the analysis, it is concluded that the proposed scheme [37] is secure and achieve constant size system parameter which means that execution of algorithm does not depends on the time or execution time is constant.

Chapter 5

Implementation of Identity Based Broadcast Encryption Scheme using Weil pairing

In this chapter, we will discuss the implementation of Yang's scheme[37]. For this purpose, we aim to use modified weil pairing [34] together with group of points on an elliptic curve $E_{\mathbb{F}_p}(a, b)$. We explain the implementation by a toy example. Implemented code is given in Appendix.

5.1 Our construction

In this Section, we will built an example of IBBE scheme that is defined in Chapter 4. For this purpose, first we will choose the bilinear mapping that is used in Yang's IBBE scheme [37]. The good example of bilinear mapping is weil pairing as discussed in Chapter 3. So, we will implement weil pairing in computer algebra system ApCoCoA [1] and with the help of this code we will implement weil and IBBE scheme.

5.1.1 Assumptions

In this Section, we discuss assumption that is helpful in developing the example.

1. We select the field $\mathbb{F}_p = \mathbb{Z}_p$ of order prime p where p satisfying the condition
$$p = 2 \mod 3.$$

2. We define elliptic curve by the equation $y^2 = x^3 + 1$ over field $\mathbb{Z}_p$.

3. Let $A \in E_{\mathbb{F}_p}(a, b)$ be the point of order $q = \frac{(p+1)}{6}$ where $q > 3$ be the some prime factor of $p + 1$ and q satisfies the condition $q \mid p + 1$ but $q^2 \nmid p + 1$. Here we denote $\mathbb{G}_1$ be the subgroup of point generated by A.

4. Now we choose extension field $\mathbb{F}_{p^2}$ by using the fact that $1 \neq \beta \in \mathbb{F}_{p^2}$ but $\beta^3 = 1$ that defines the mapping $\omega(x_1, y_1) = (\beta x_1, y_1)$. Note that
$$\omega(A) \in E_{\mathbb{F}_{p^2}}(0, 1) \text{ but } \omega(A) \notin E_{\mathbb{F}_p}(0, 1)$$
.

5. We take $\mathbb{G}_2$ be the subgroup of $\mathbb{F}_{p^2}$ containing all elements of order $q = \frac{(p+1)}{6}$ which is the group that contains all irreducible polynomials in mod $\beta^2 + \beta + 1$

6. When we discuss weil pairing on $\mathbb{F}_{p^2}$ its mapping is defined as
$$\phi : E_{\mathbb{F}_p}(a, b) \times E_{\mathbb{F}_p}(a, b) \longrightarrow \mathbb{G}_2$$
and modified weil pairing is defined as $\hat{\phi_n} : \mathbb{G}_1 \times \mathbb{G}_2 \longrightarrow \mathbb{G}_2$ which is defined as:
$$\hat{\phi_n}(A, A) = \phi_n(A, \omega(A))$$

Now we give the an example of prime order group that we use in our construction

Example 5.1.1 We choose $p = 29$ that satisfies the condition $p = 2 \mod 3$. Now we choose prime number $q = 5$ which is the order subgroup $\mathbb{G}_1$ and satisfying $q = \frac{p+1}{6} = \frac{30}{6} = 5$ also check that $q = 5$ is prime factor of $p + 1$ and $5 \mid 30$ but $25 \nmid 30$.

Using these facts we take field $\mathbb{F}_p = \mathbb{F}_{29}$ and subgroup is $E_5(0, 1)$ and elliptic curve is defined by equation $y^2 = x^3 + 1$. Let $\mathbb{G}_1$ be the subgroup of $\mathbb{F}_{29}$ having order 5. Here we take $\mathbb{G}_2$ is also group of order q which is the subgroup of field extension $\mathbb{F}_{29^2}$ taking mod $\beta^2 + \beta + 1$. So the mapping defined as:

$$\hat{\phi_n} : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$$

We choose $A = (8, 7) \in E_{29}(0, 1)$ is the point of order $q = 5$

Now, we calculate the Modified weil pairing using ApCoCoA tool see Appendix.

$$\hat{\phi_5}((8,7),(8,7)) = \phi_5((8,7),\phi((8,7))) = \phi_5((8,7),(8\beta,7)) = 15\beta + 10$$
$$\text{where } (15\beta + 10)^5 = 1 \mod p$$

Here we choose $C = (0, 28)$.

Before going to construct the example, we will verify the properties that are used in Ming and Wang's IBBE scheme [37].

1. **Bilinear Mapping:** In IBBE the bilinear mapping is defined as, let $u, v \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_p$ then $\phi(u^a, v^b) = \phi(u, v)^{ab}$. But, we will use $\mathbb{G}_1 = E_{\mathbb{F}}(a, b)$ and elliptic curve is addition group so exponent is written as

   $$\phi(au, bv) = \phi(u, v)^{ab}$$

   . For example, let $a = 2, b = 3$ then $u = v = (8, 7)$ therefore,

   $$\phi(2(8,7),3(8,7)) = \phi((8,7),(8,7))^{2\cdot 3}$$
   $$\phi((4,23),(4,6)) = \phi((8,7),(8,7))^{6}$$
   $$15\beta + 10 = 15\beta + 10$$

2. **Non-Degenerate:** $\phi(g, g) \neq 1$, As we use elliptic curve so we take $g = (8, 7)$ and modified weil pairing. Therefore, $\phi((8,7),(8,7)) = 15\beta + 10 \neq 1$.

3. Let $v = (v_1, v_2 \dots, v_m)$ and $g \in \mathbb{G}_1$ and g^v is m-tuple of element of $\mathbb{G}_1$. In elliptic curve these m-tuples is equal to

$$g^v = \{g^{v_1}, g^{v_2}, \dots, g^{v_1}\} = \{v_1 g, v_2 g, \dots, v_m g\}$$

. For example, let $v = (2, 3)$ and $g = (8, 7)$ so

$$\{(8, 7)^2, (8, 7)^3\} = \{2(8, 7), 3(8, 7)\}$$

.

Now, we discuss how our weil pairing is work on Ming and Wang IBBE scheme.

5.1.2 Example 1

IBBE has four algorithm that is discussed in Chapter 4

1. **Setup:** Setup algorithm calculate the master key of IBBE system. For that purpose we take $m = 6$, the number of receivers, now we assume that PKG first chooses standard bases of $\mathbb{Z}_{29}^6$.
$D = \{(1, 0, 0, 0, 0, 0), (0, 1, 0, 0, 0, 0), (0, 0, 1, 0, 0, 0),$
$(0, 0, 0, 1, 0, 0), (0, 0, 0, 1, 0, 0), (0, 0, 0, 0, 1, 0), (0, 0, 0, 0, 0, 1)\} = D^*$
Here D is basis and D^* is orthonormal basis of system both bases are equal to each other and satisfies the condition.

$$b_i^* \cdot (b_j) = \begin{cases} 0 \mod p & \text{if } i \neq j \\ r \mod p & \text{if } i = j \end{cases}$$

Where $p = 29$ and as we take the standard bases then $r = 1$ otherwise r is any random number belongs to $\mathbb{Z}_p^*$.
Now, master key that is defined as

$$\mathcal{K} = \{g^{\alpha\theta d_1^*}, g^{\theta d_1^*}, g^{\theta d_2^*}, g^{\sigma d_3^*}, g^{\sigma d_4^*}\}$$

We will choose $\alpha = 2, \theta = 4, \sigma = 7 \in \mathbb{Z}_{29}^*$.

First we will calculate the public parameters that are made to be public and defined as

$$\mathcal{PP} = \{\mathbb{G}_1, \mathbb{G}_2, g, p, \phi(g,g)^{\alpha\theta d_1 d_1^*}, g^{d_1}, g^{d_2}, g^{d_3}, g^{d_4}\}$$

In this example pubic parameter are

$$\begin{aligned}
\mathcal{PP} = \{&\mathbb{G}_1 = E_{29}(0,1), \mathbb{G}_2 = \mathbb{F}_{29^2}, g = (8,7), p = 29, \\
&\phi(g,g)^{\alpha\theta d_1 d_1^*} = \phi((8,7),(8,7))^{2\times 4(1,0,0,0,0,0)(1,0,0,0,0,0)} = 12\beta + 3, \\
&g^{d_1} = (8,7)^{(1,0,0,0,0,0)} = \{((8,7), \mathcal{O}, \mathcal{O}, \mathcal{O}, \mathcal{O}, \mathcal{O}), \} \\
&g^{d_2} = g^{(0,1,0,0,0,0)} = \{\mathcal{O}, (8,7), \mathcal{O}, \mathcal{O}, \mathcal{O}, \mathcal{O}\} \\
&g^{d_3} = g^{(0,0,1,0,0,0)} = \{\mathcal{O}, \mathcal{O}, (8,7), \mathcal{O}, \mathcal{O}, \mathcal{O}\}, \\
&g^{d_4} = g^{(0,0,0,1,0,0)} = \{\mathcal{O}, \mathcal{O}, \mathcal{O}, (8,7), \mathcal{O}, \mathcal{O}\}
\end{aligned}$$

Therefore, calculation for master key is

$$\begin{aligned}
&\mathcal{K} = \{(8,7)^{(2)(4)(1,0,0,0,0,0)}, (8,7)^{(4)(1,0,0,0,0,0)}, (8,7)^{(4)(0,1,0,0,0,0)} \\
&\quad , (8,7)^{(7)(0,0,1,0,0,0)}, (8,7)^{(7)(0,0,0,1,0,0)}\} \\
&\mathcal{K} = \{(8,7)^{(8,0,0,0,0,0)}, (8,7)^{(4,0,0,0,0,0)}, (8,7)^{(0,4,0,0,0,0)}, (8,7)^{(0,0,7,0,0,0)}, (8,7)^{(0,0,0,7,0,0)}\}
\end{aligned}$$

Now, using the scalar multiplication of elliptic curve, the above equation is written as:

$$\begin{aligned}
\mathcal{K} = \{&(8(8,7), 0(8,7), 0(8,7), 0(8,7), 0(8,7), 0(8,7)),\\
&(4(8,7), 0(8,7), 0(8,7), 0(8,7), 0(8,7), 0(8,7)),\\
&(0(8,7), 4(8,7), 0(8,7), 0(8,7), 0(8,7), 0(8,7)),\\
&(0(8,7), 0(8,7), 7(8,7), 0(8,7), 0(8,7), 0(8,7)),\\
&(0(8,7), 0(8,7), 0(8,7), 7(8,7), 0(8,7), 0(8,7))\}\\
\mathcal{K} = \{&((4,6), \mathcal{O}, \mathcal{O}, \mathcal{O}, \mathcal{O}, \mathcal{O}), ((8,22), \mathcal{O}, \mathcal{O}, \mathcal{O}, \mathcal{O}, \mathcal{O}),\\
&(\mathcal{O}, (8,22), \mathcal{O}, \mathcal{O}, \mathcal{O}, \mathcal{O}), (\mathcal{O}, \mathcal{O}, (4,23), \mathcal{O}, \mathcal{O}, \mathcal{O}),\\
&(\mathcal{O}, \mathcal{O}, \mathcal{O}, (4,23), \mathcal{O}, \mathcal{O})\}
\end{aligned}$$

2. **Extract:** Now the system calculate the private key of the corresponding identity. For this we take set of Identities $\Omega = \{2, 5, 7, 3, 14, 13\}$ and PKG randomly chooses

$$\begin{aligned}
r_1^1, r_1^2, r_1^3, r_1^4, r_1^5, r_1^6 &= 4, 3, 7, 9, 2, 6 \in \mathbb{Z}_{29}^*\\
r_2^1, r_2^2, r_2^3, r_2^4, r_2^5, r_2^6 &= 6, 4, 2, 13, 8, 1 \in \mathbb{Z}_{29}^*
\end{aligned}$$

$\mathcal{K}_1$ is defined as

$$\mathcal{K}_1 \quad = g^{\alpha\theta d_1^* + r_1^i ID_i \theta d_1^* - r_1^i \theta d_2^* + r_2^i ID_i \sigma d_3^* - r_2^i \sigma d_4^*}$$

Let $i = 1$ so pick up first identity $ID_1 = 2 \in \Omega$ and $r_1^1 = 4, r_2^1 = 6$

$$\begin{aligned}
\mathcal{K}_1 = (8,7)&^{(2)(4)(1,0,0,0,0,0)+(4)(2)(4)(1,0,0,0,0,0)-(4)(4)(0,1,0,0,0,0)}\\
(8,7)&^{(6)(2)(7)(0,0,1,0,0,0)-(6)(7)(0,0,0,1,0,0)}
\end{aligned}$$

Simplifying, the exponents we will get

$$\begin{aligned}
\mathcal{K}_1 &= (8,7)^{(8,0,0,0,0,0)+(32,0,0,0,0,0)+(0,-8,0,0,0,0)+(0,0,84,0,0,0)+(0,0,0,-42,0,0)}\\
\mathcal{K}_1 &= (8,7)^{(40,-16,84,-42,0,0)}
\end{aligned}$$

Using the scalar multiplication of elliptic curve, above equation can be written as

$$\begin{aligned}
\mathcal{K}_1 &= (40(8,7), -16(8,7), 84(8,7), -42(8,7), 0(8,7), 0(8,7)) \\
\mathcal{K}_1 &= \{\mathcal{O}, (8,22), (8,22), (4,6), \mathcal{O}, \mathcal{O}\}
\end{aligned}$$

$\mathcal{K}_2$ is defined as

$$\mathcal{K}_2 \quad = \begin{cases}
g^{(r_1^1+r_1^2+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)(ID_1+ID_2+\ldots,ID_6)\theta d_1^*}. \\
g^{r_1^i(ID_1+ID_2+\cdots+ID_{i-1}+ID_{i+1}+\ldots ID_6)\theta d_1^*}. \\
g^{-(r_1^1+r_1^2+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)\theta d_2^*}. \\
g^{(r_2^1+r_2^2+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)(ID_1+ID_2+\ldots,ID_6)\sigma d_3^*}. \\
g^{r_2^i(ID_1+ID_2+\cdots+ID_{i-1}+ID_{i+1}+\cdots+ID_6)\sigma d_3^*}. \\
g^{-(r_2^1+r_2^2+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)\sigma d_4^*}
\end{cases}$$

Putting values

$$\mathcal{K}_2 = \begin{cases}
(8,7)^{(3+7+9+2+6)(2+5+7+3+14+13)(4)(1,0,0,0,0,0)} & \text{Skiping the } r_1^1 = 4 \\
.(8,7)^{(4)(5+7+3+14+13)(4)(1,0,0,0,0,0)} & \text{Skiping the } ID_1 = 2 \\
.(8,7)^{-(3+7+9+2+6)(4)(0,1,0,0,0,0)} & \text{Skiping the } r_1^1 = 4 \\
.(8,7)^{(4+2+13+8+1)(2+5+7+3+14+13)(7)(0,0,1,0,0,0)} & \text{Skiping the } r_2^1 = 6 \\
.(8,7)^{(3)(5+7+3+14+13)(7)(0,0,1,0,0,0)} & \text{Skiping the } ID_1 = 2 \\
.(8,7)^{-(4+2+13+8+1)(7)(0,0,0,1,0,0)} & \text{Skiping the } r_2^1 = 6
\end{cases}$$

Adding the identities and random number series of r_1^i and r_2^i, where $1 \leq i \leq m$

$$\begin{aligned}
\mathcal{K}_2 = (8,7)^{(27)(44)(4)(1,0,0,0,0,0)}.(8,7)^{(4)(42)(4)(1,0,0,0,0,0)} \\
.(8,7)^{-(27)(4)(0,1,0,0,0,0)}.(8,7)^{(28)(44)(7)(0,0,1,0,0,0)} \\
.(8,7)^{(3)(42)(7)(0,0,1,0,0,0)}.(8,7)^{-(28)(7)(0,0,0,1,0,0)}
\end{aligned}$$

Now, adding them component wise

$$\mathcal{K}_2 = (8,7)^{(5424,-108,10388,-196,0,0)}$$

Writing the above equation as scalar multiple of point, and using scalar multiplication of elliptic curve, we will get

$$\mathcal{K}_2 = \{5424(8,7), -108(8,7), 10388(8,7), -196(8,7), 0(8,7), 0(8,7)\}$$
$$\mathcal{K}_2 = \{(8,22), (4,23), (4,6), (8,22)\mathcal{O}, \mathcal{O}\}$$

Similarly, by using the above procedure one can calculate private keys for all other identities $\Omega = \{5,7,3,14,13\}$

3. **Encrypt:** Choose $s_1 = 21, s_2 = 6 \in \mathbb{Z}_{29}^*$ and take message $M = 16$. Ciphertext C_1 is defined as:

$$C_1 = M.\phi(g,g)^{\alpha\theta s_1 d_1 \cdot d_1^*}$$

For calculation of C_1 we need to calculate the bilinear mapping ϕ. As we use the modified weil pairing and choose point $(8,7)$ of order 5. Therefore, we calculate $\hat{\phi_n}(A,A) = \hat{\phi_5}((8,7),(8,7))$. Then putting the values in (4.9), we get.

$$C_1 = 16.\hat{\phi_5}((8,7),(8,7))^{(2)(4)(21)(1,0,0,0,0,0)(1,0,0,0,0,0)}$$

Using ApCoCoA tool the modified weil pairing $\hat{\phi_5}((8,7),(8,7)) = (15\beta + 10)$

$$C_1 = 16.(15\beta + 10)^{(2)(4)(21)(1,0,0,0,0,0)(1,0,0,0,0,0)}$$

Multiplying the exponent and taking the dot product of exponents and reduced it into $\mod 29$.

$$C_1 = 16.((15\beta + 10)^{168} \mod \beta^2 + \beta + 1) \mod 29$$
$$C_1 = 16.(12\beta + 3) \mod 29$$
$$C_1 = 18\beta + 19 \mod 29$$

Now C_2 is defined as

$$C_2 = g^{s_1 d_1 + s_1(ID_1, ID_2, \ldots, ID_6)d_2 + s_2 d_3 + s_2(ID_1, ID_2, \ldots, ID_6)d_4}$$

Putting values, we get

$$C_2 = (8,7)^{(21,0,0,0,0,0)+(21)(2+5+7+3+14+13)(0,1,0,0,0,0)}.$$
$$(8,7)^{(6)(0,0,1,0,0,0)+(6)(2+5+7+3+14+13)(0,0,0,1,0,0)}$$
$$C_2 = (8,7)^{(21,924,6,264,0,0)} \mod 29$$

Therefore, using scalar multiplication of elliptic curve above equation can be rewritten as

$$C_2 = (21(8,7), 924(8,7), 6(8,7), 264(8,7), 0(8,7), 0(8,7)) \mod 29$$
$$C_2 = \{(8,7), (8,22), (8,7), (8,22), \mathcal{O}, \mathcal{O}\}$$

4. **Decrypt:** Message decryption is define as $M = C_1/\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2)$. For decryption, we use both decryption keys and ciphertexts that is calculated as below.

$$\mathcal{K}_1 = \{\mathcal{O}, (8,22), (8,22), (4,6), \mathcal{O}, \mathcal{O}\}$$
$$\mathcal{K}_2 = \{(8,22), (4,23), (4,6), (8,22), \mathcal{O}, \mathcal{O}\}$$
$$C_1 = 18\beta + 19 \mod 29$$
$$C_2 = \{(8,7), (8,22), (8,7), (8,22), \mathcal{O}, \mathcal{O}\}$$

From the definition of message decryption, first of all we will calculate the denominator, for that purpose we need to calculate the product of pairing that is denoted by $\phi_m = \phi_6$ between both decryption keys and ciphertext C_2 as defined as $\phi_5(\mathcal{K}_1\mathcal{K}_2, C_2)$.

But, elliptic curve is group of addition so we will add both keys $\mathcal{K}_1 + \mathcal{K}_2$ by using addition Algorithm 3.1.1

$$\mathcal{K}_1 + \mathcal{K}_2 = ((8,22),(8,7),(4,23),(4,23),\mathcal{O},\mathcal{O})$$

We use the modified weil pairing between keys $\mathcal{K}_1 + \mathcal{K}_2$ and ciphertext C_2 which is denoted by $\hat{\phi_m} = \hat{\phi_6}$. Applying the pairing component wise between six tuples

$$\hat{\phi_6}(\mathcal{K}_1 + \mathcal{K}_2, C_2) = \begin{cases} \hat{\phi}((8,22),(8,7)).\hat{\phi}((8,7),(8,22)).\hat{\phi}((4,23),(8,7)). \\ \hat{\phi}((4,23),(8,22)).\hat{\phi}(\mathcal{O},\mathcal{O}).\hat{\phi}(\mathcal{O},\mathcal{O}) \end{cases}$$

Applying the definition of modified weil pairing, we get

$$\phi(\mathcal{K}_1 + \mathcal{K}_2, C_2) = \begin{cases} \hat{\phi}((8,22),(8\beta,7)).\hat{\phi}((8,7),(8\beta,22)).\hat{\phi}((4,23),(8\beta,7)). \\ \hat{\phi}((4,23),(8\beta,22)).\hat{\phi}(\mathcal{O},\mathcal{O}).\hat{\phi}(\mathcal{O},\mathcal{O}) \end{cases}$$

Using ApCoCoA code, we will calculate the modified weil pairing (see Appendix B).

$$\begin{aligned} \phi(\mathcal{K}_1 + \mathcal{K}_2, C_2) &= ((14\beta+24)\cdot(14\beta+24)\cdot(17\beta+20)\cdot \\ &\quad (12\beta+3)\cdot(1)\cdot(1) \mod \beta^2+\beta+1) \mod 29 \\ \phi(\mathcal{K}_1 + \mathcal{K}_2, C_2) &= -76896\beta - 96132 \mod 29 \\ \phi(\mathcal{K}_1 + \mathcal{K}_2, C_2) &= 12\beta + 3 \end{aligned}$$

By putting values, we get

$$M = (18\beta + 19)/(12\beta + 3)$$

Using extended euclidean Algorithm 2.5.14, we have $(12\beta+3)^{-1} = (17\beta+20)$. So, message will be

$$M = ((18\beta + 19) * (17\beta + 20) \mod \beta^2 + \beta + 1) \mod 29$$

$$M = 377\beta + 74 \mod 29$$

$$M = 16 \mod 29$$

5.1.3 Example 2

In this example, we show that our scheme work on the randomly chooses orthonormal bases that satisfies the condition (4.2). Assume that PKG first chooses the bases as

$$\begin{aligned} D = \{&(1,2,3,4,5,6),(1,0,2,3,5,2),(2,3,0,5,2,7),\\ &(2,3,4,0,2,1),(1,0,3,4,0,1),(2,3,1,7,0,0)\} \end{aligned}$$

We also check that D is linearly independent by using the method define in 2.5.21. Now we calculate the orthonormal basis D^* that fulfil the condition (4.2).

Example 5.1.2 Let $D = \{(1,2,3,4,5,6),(1,0,2,3,5,2),(2,3,0,5,2,7),$ $(2,3,4,0,2,1),(1,0,3,4,0,1),(2,3,1,7,0,0)\}$ be the basis of $\mathbb{Z}_{29}^6$. Let

$$\begin{aligned} D^* = \{&(d^*_{11},d^*_{12},d^*_{13},d^*_{14},d^*_{15},d^*_{16}),(d^*_{21},d^*_{22},d^*_{23},d^*_{24},d^*_{25},d^*_{26}),(d^*_{31},d^*_{32},d^*_{33},d^*_{34},d^*_{35},d^*_{36}),\\ &(d^*_{41},d^*_{42},d^*_{43},d^*_{44},d^*_{45},d^*_{46}),(d^*_{51},d^*_{52},d^*_{53},d^*_{54},d^*_{55},d^*_{56}),(d^*_{61},d^*_{62},d^*_{63},d^*_{64},d^*_{65},d^*_{66})\} \end{aligned}$$

be the orthonormal basis. By using condition (4.2), we will make the system of linear equations by multiplying the first component of D^* by the components of D. We will choose the random $r = 3$ in this case.Therefore, first system of linear equations are:

$$\begin{cases} 1d_{11}^{*} + 2d_{12}^{*} + 3d_{13}^{*} + 4d_{14}^{*} + 5d_{15}^{*} + 6d_{16} = 3 \\ 1d_{11}^{*} + 0d_{12}^{*} + 2d_{13}^{*} + 3d_{14}^{*} + 5d_{15}^{*} + 2d_{16} = 0 \\ 2d_{11}^{*} + 3d_{12}^{*} + 0d_{13}^{*} + 5d_{14}^{*} + 2d_{15}^{*} + 7d_{16} = 0 \\ 2d_{11}^{*} + 3d_{12}^{*} + 4d_{13}^{*} + 0d_{14}^{*} + 2d_{15}^{*} + 1d_{16} = 0 \\ 1d_{11}^{*} + 0d_{12}^{*} + 3d_{13}^{*} + 4d_{14}^{*} + 0d_{15}^{*} + 1d_{16} = 0 \\ 2d_{11}^{*} + 3d_{12}^{*} + 1d_{13}^{*} + 7d_{14}^{*} + 0d_{15}^{*} + 0d_{16} = 0 \end{cases} \tag{5.1}$$

Solving them by transforming into the augmented matrix and solve by reduced echelon form.

$$\begin{bmatrix} d_{11}^{*} \\ d_{12}^{*} \\ d_{13}^{*} \\ d_{14}^{*} \\ d_{15}^{*} \\ d_{16}^{*} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & \mid & 3 \\ 1 & 0 & 2 & 3 & 5 & 2 & \mid & 0 \\ 2 & 3 & 0 & 5 & 2 & 7 & \mid & 0 \\ 2 & 3 & 4 & 0 & 2 & 1 & \mid & 0 \\ 1 & 0 & 3 & 4 & 0 & 1 & \mid & 0 \\ 2 & 3 & 1 & 7 & 0 & 0 & \mid & 0 \end{bmatrix}$$

Using the following row operation: $(R_1+(-1)R_2)$, $(R_1+(-2)R_3)$, $(R_1+(-2)R_4)$, $(R_1+(-1)R_5)$, $(R_1+(-2)R_6)$ and reduce their result in $\mod 29$. We get

$$\begin{bmatrix} d_{11}^{*} \\ d_{12}^{*} \\ d_{13}^{*} \\ d_{14}^{*} \\ d_{15}^{*} \\ d_{16}^{*} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & \mid & 3 \\ 0 & 27 & 28 & 28 & 0 & 25 & \mid & 26 \\ 0 & 28 & 23 & 26 & 21 & 24 & \mid & 23 \\ 0 & 28 & 27 & 21 & 21 & 18 & \mid & 23 \\ 0 & 27 & 0 & 0 & 24 & 24 & \mid & 26 \\ 0 & 28 & 24 & 28 & 19 & 17 & \mid & 23 \end{bmatrix} \mod 29$$

Now multiplying second row by inverse of 27 by using extended euclidean inverse see Section 2.5.14 i.e $(27)^{-1} = 14 \mod 29$.

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & | & 3 \\ 0 & 1 & 15 & 15 & 0 & 2 & | & 16 \\ 0 & 28 & 23 & 26 & 21 & 24 & | & 23 \\ 0 & 28 & 27 & 21 & 21 & 18 & | & 23 \\ 0 & 27 & 0 & 0 & 24 & 24 & | & 26 \\ 0 & 28 & 24 & 28 & 19 & 17 & | & 23 \end{bmatrix} \mod 29$$

Now use the following row operations that is $(28R_2 - R_3)$, $(28R_2 - R_4)$, $(27R_2 - R_5)$, $(28R_2 - R_6)$.

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & | & 3 \\ 0 & 1 & 15 & 15 & 0 & 2 & | & 16 \\ 0 & 0 & 9 & 12 & 21 & 26 & | & 10 \\ 0 & 0 & 13 & 7 & 21 & 20 & | & 10 \\ 0 & 0 & 1 & 1 & 24 & 28 & | & 0 \\ 0 & 0 & 10 & 14 & 19 & 19 & | & 10 \end{bmatrix} \mod 29$$

Now multiplying third row by inverse of 9 by using extended euclidean inverse see Section 2.5.14 i.e $(9)^{-1} = 13 \mod 29$.

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & | & 3 \\ 0 & 1 & 15 & 15 & 0 & 2 & | & 16 \\ 0 & 0 & 1 & 11 & 12 & 19 & | & 14 \\ 0 & 0 & 13 & 7 & 21 & 20 & | & 10 \\ 0 & 0 & 1 & 1 & 24 & 28 & | & 0 \\ 0 & 0 & 10 & 14 & 19 & 19 & | & 10 \end{bmatrix} \mod 29$$

Using the following row operations: $(13R_3 - R_4),\ (R_3 - R_5),\ (10R_3 - R_6)$

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & | & 3 \\ 0 & 1 & 15 & 15 & 0 & 2 & | & 16 \\ 0 & 0 & 1 & 11 & 12 & 19 & | & 14 \\ 0 & 0 & 0 & 20 & 19 & 24 & | & 27 \\ 0 & 0 & 0 & 10 & 17 & 20 & | & 14 \\ 0 & 0 & 0 & 9 & 14 & 26 & | & 14 \end{bmatrix} \mod 29$$

Now multiplying $4th$ row by inverse of 20 by using extended euclidean inverse see Section 2.5.14 i.e $(20)^{-1} = 16 \mod 29$.

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & | & 3 \\ 0 & 1 & 15 & 15 & 0 & 2 & | & 16 \\ 0 & 0 & 1 & 11 & 12 & 19 & | & 14 \\ 0 & 0 & 0 & 1 & 14 & 7 & | & 26 \\ 0 & 0 & 0 & 10 & 17 & 20 & | & 14 \\ 0 & 0 & 0 & 9 & 14 & 26 & | & 14 \end{bmatrix} \mod 29$$

Using the following row operation: $(10R_4 - R_5),\ (9R_4 - R_6)$

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & | & 3 \\ 0 & 1 & 15 & 15 & 0 & 2 & | & 16 \\ 0 & 0 & 1 & 11 & 12 & 19 & | & 14 \\ 0 & 0 & 0 & 1 & 14 & 7 & | & 26 \\ 0 & 0 & 0 & 0 & 7 & 21 & | & 14 \\ 0 & 0 & 0 & 0 & 25 & 8 & | & 17 \end{bmatrix} \mod 29$$

Now multiplying $5th$ row by inverse of 7 by using extended euclidean inverse see Section 2.5.14 i.e $(9)^{-1} = 13 \mod 29$.

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & \mid & 3 \\ 0 & 1 & 15 & 15 & 0 & 2 & \mid & 16 \\ 0 & 0 & 1 & 11 & 12 & 19 & \mid & 14 \\ 0 & 0 & 0 & 1 & 14 & 7 & \mid & 26 \\ 0 & 0 & 0 & 0 & 1 & 3 & \mid & 2 \\ 0 & 0 & 0 & 0 & 1 & 2 & \mid & 3 \end{bmatrix} \mod 29$$

Apply the following operation $(25R_5 - R_6)$. After this $6th$ row becomes $[0\,0\,0\,0\,0\,9|4]$ multiplying resulting $6th$ row by inverse of 9 using extended euclidean inverse 2.5.14 i.e. $\frac{-519}{4091}$

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & \mid & 3 \\ 0 & 1 & 15 & 15 & 0 & 2 & \mid & 16 \\ 0 & 0 & 1 & 11 & 12 & 19 & \mid & 14 \\ 0 & 0 & 0 & 1 & 14 & 7 & \mid & 26 \\ 0 & 0 & 0 & 0 & 1 & 3 & \mid & 2 \\ 0 & 0 & 0 & 0 & 0 & 1 & \mid & 23 \end{bmatrix} \mod 29$$

Use the following row operations $3R_6 - R_5,\ 7R_6 - R_4,\ 19R_6 - R_3,\ 2R_6 - R_2,\ 6R_6 - R_1$

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 28 & 27 & 26 & 25 & 24 & 0 & \mid & 19 \\ 0 & 28 & 14 & 14 & 0 & 0 & \mid & 1 \\ 0 & 0 & 28 & 18 & 17 & 0 & \mid & 17 \\ 0 & 0 & 0 & 28 & 15 & 0 & \mid & 19 \\ 0 & 0 & 0 & 0 & 28 & 0 & \mid & 9 \\ 0 & 0 & 0 & 0 & 0 & 1 & \mid & 23 \end{bmatrix} \mod 29$$

Multiplying $5th$ row by inverse of 28 i.e $(28)^{-1} = 28 \mod 29$

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 28 & 27 & 26 & 25 & 24 & 0 & | & 19 \\ 0 & 28 & 14 & 14 & 0 & 0 & | & 1 \\ 0 & 0 & 28 & 18 & 17 & 0 & | & 17 \\ 0 & 0 & 0 & 28 & 15 & 0 & | & 19 \\ 0 & 0 & 0 & 0 & 1 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 0 & 1 & | & 23 \end{bmatrix} \mod 29$$

Using the row operation: $15R_5 - R_4, 17R_5 - R_3,\ 24R_5 - R_1$

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 0 & 0 & | & 26 \\ 0 & 28 & 14 & 14 & 0 & 0 & | & 1 \\ 0 & 0 & 1 & 11 & 0 & 0 & | & 4 \\ 0 & 0 & 0 & 1 & 0 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 1 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 0 & 1 & | & 23 \end{bmatrix} \mod 29$$

Using the following row operations $11R_4 - R_3,\ 14R_4 - R_2,\ 4R_4 - R_1$.

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 28 & 27 & 26 & 0 & 0 & 0 & | & 25 \\ 0 & 1 & 15 & 0 & 0 & 0 & | & 18 \\ 0 & 0 & 28 & 0 & 0 & 0 & | & 13 \\ 0 & 0 & 0 & 1 & 0 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 1 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 0 & 1 & | & 23 \end{bmatrix} \mod 29$$

Multiplying third row by inverse of 28 i.e $(28)^{-1} = 28 \mod 29$

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 28 & 27 & 26 & 0 & 0 & 0 & | & 25 \\ 0 & 1 & 15 & 0 & 0 & 0 & | & 18 \\ 0 & 0 & 1 & 0 & 0 & 0 & | & 16 \\ 0 & 0 & 0 & 1 & 0 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 1 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 0 & 1 & | & 23 \end{bmatrix} \mod 29$$

Using the following row operations $26R_3 - R_1,\ 15R_3 - R_2$.

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 0 & 0 & 0 & 0 & | & 14 \\ 0 & 28 & 0 & 0 & 0 & 0 & | & 19 \\ 0 & 0 & 1 & 0 & 0 & 0 & | & 16 \\ 0 & 0 & 0 & 1 & 0 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 1 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 0 & 1 & | & 23 \end{bmatrix} \mod 29$$

Multiplying second row by inverse of 28 i.e $(28)^{-1} = 28 \mod 29$

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 2 & 0 & 0 & 0 & 0 & | & 14 \\ 0 & 1 & 0 & 0 & 0 & 0 & | & 10 \\ 0 & 0 & 1 & 0 & 0 & 0 & | & 16 \\ 0 & 0 & 0 & 1 & 0 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 1 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 0 & 1 & | & 23 \end{bmatrix} \mod 29$$

Now, use the following operation $2R_2 - R_1$. After this 1^{st} row becomes $[28\ 0\ 0\ 0\ 0\ 0|6]$ multiplying resulting 1^{st} row by inverse of 28 using extended euclidean inverse

2.5.14 i.e. $(28)^{-1} = 28 \mod 29$. Therefore,

$$\begin{bmatrix} d^*_{11} \\ d^*_{12} \\ d^*_{13} \\ d^*_{14} \\ d^*_{15} \\ d^*_{16} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & | & 23 \\ 0 & 1 & 0 & 0 & 0 & 0 & | & 10 \\ 0 & 0 & 1 & 0 & 0 & 0 & | & 16 \\ 0 & 0 & 0 & 1 & 0 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 1 & 0 & | & 20 \\ 0 & 0 & 0 & 0 & 0 & 1 & | & 23 \end{bmatrix} \mod 29$$

Therefore, the values of $(d^*_{11}, d^*_{12}, d^*_{13}, d^*_{14}, d^*_{15}, d^*_{16})$ are:

$$\begin{aligned} d^*_{11} &= 23 \mod 29 & \qquad d^*_{12} &= 10 \mod 29 \\ d^*_{13} &= 16 \mod 29 & \qquad d^*_{14} &= 20 \mod 29 \\ d^*_{15} &= 20 \mod 29 & \qquad d^*_{16} &= 23 \mod 29 \end{aligned}$$

To find the second component of D^*, we will multiply the second component of orthonormal basis D^* with all components of D one by one and making the second system of linear equations.

$$\begin{cases} 1d^*_{21} + 2d^*_{22} + 3d^*_{23} + 4d^*_{24} + 5d^*_{25} + 6d_{26} = 0 \\ 1d^*_{21} + 0d^*_{22} + 2d^*_{23} + 3d^*_{24} + 5d^*_{25} + 2d_{26} = 3 \\ 2d^*_{21} + 3d^*_{22} + 0d^*_{23} + 5d^*_{24} + 2d^*_{25} + 7d_{26} = 0 \\ 2d^*_{21} + 3d^*_{22} + 4d^*_{23} + 0d^*_{24} + 2d^*_{25} + 1d_{26} = 0 \\ 1d^*_{21} + 0d^*_{22} + 3d^*_{23} + 4d^*_{24} + 0d^*_{25} + 1d_{26} = 0 \\ 2d^*_{21} + 3d^*_{22} + 1d^*_{23} + 7d^*_{24} + 0d^*_{25} + 0d_{26} = 0 \end{cases} \tag{5.2}$$

Similarly, by using the above procedure we can solve the following system (5.2). and their calculated values are,

$$\begin{aligned}
d^*_{21} &= 21 \mod 29\\
d^*_{22} &= 9 \mod 29\\
d^*_{23} &= 18 \mod 29\\
d^*_{24} &= 0 \mod 29\\
d^*_{25} &= 25 \mod 29\\
d^*_{26} &= 12 \mod 29
\end{aligned}$$

To find the third component of D^*, we will multiply the third component of orthonormal basis D^* with all components of D one by one and making the third system of linear equations.

$$\begin{cases}
1d^*_{31} + 2d^*_{32} + 3d^*_{33} + 4d^*_{34} + 5d^*_{35} + 6d_{36} = 0\\
1d^*_{31} + 0d^*_{32} + 2d^*_{33} + 3d^*_{34} + 5d^*_{35} + 2d_{36} = 0\\
2d^*_{31} + 3d^*_{32} + 0d^*_{33} + 5d^*_{34} + 2d^*_{35} + 7d_{36} = 3\\
2d^*_{31} + 3d^*_{32} + 4d^*_{33} + 0d^*_{34} + 2d^*_{35} + 1d_{36} = 0\\
1d^*_{31} + 0d^*_{32} + 3d^*_{33} + 4d^*_{34} + 0d^*_{35} + 1d_{36} = 0\\
2d^*_{31} + 3d^*_{32} + 1d^*_{33} + 7d^*_{34} + 0d^*_{35} + 0d_{36} = 0
\end{cases} \tag{5.3}$$

Similarly, the values of following equation (5.3) are,

$$\begin{aligned}
d^*_{31} &= 11 \mod 29\\
d^*_{32} &= 26 \mod 29\\
d^*_{33} &= 12 \mod 29\\
d^*_{34} &= 13 \mod 29\\
d^*_{35} &= 19 \mod 29\\
d^*_{36} &= 17 \mod 29
\end{aligned}$$

To find the forth component of D^*, we will multiply the fourth component of orthonormal basis D^* with all components of D one by one and making the fourth system of linear equations.

$$\begin{cases} 1d_{41}^* + 2d_{42}^* + 3d_{43}^* + 4d_{44}^* + 5d_{45}^* + 6d_{46} = 0 \\ 1d_{41}^* + 0d_{42}^* + 2d_{43}^* + 3d_{44}^* + 5d_{45}^* + 2d_{46} = 0 \\ 2d_{41}^* + 3d_{42}^* + 0d_{43}^* + 5d_{44}^* + 2d_{45}^* + 7d_{46} = 0 \\ 2d_{41}^* + 3d_{42}^* + 4d_{43}^* + 0d_{44}^* + 2d_{45}^* + 1d_{46} = 3 \\ 1d_{41}^* + 0d_{42}^* + 3d_{43}^* + 4d_{44}^* + 0d_{45}^* + 1d_{46} = 0 \\ 2d_{41}^* + 3d_{42}^* + 1d_{43}^* + 7d_{44}^* + 0d_{45}^* + 0d_{46} = 0 \end{cases} \tag{5.4}$$

Similarly, the values of following equation (5.4) are,

$$\begin{aligned} d_{41}^* &= 24 \mod 29 \\ d_{42}^* &= 28 \mod 29 \\ d_{43}^* &= 19 \mod 29 \\ d_{44}^* &= 24 \mod 29 \\ d_{45}^* &= 15 \mod 29 \\ d_{46}^* &= 26 \mod 29 \end{aligned}$$

To find the fifth component of D^*, we will multiply the fifth component of orthonormal basis D^* with all components of D one by one and making the fifth system of linear equations.

$$\begin{cases} 1d_{51}^* + 2d_{52}^* + 3d_{53}^* + 4d_{54}^* + 5d_{55}^* + 6d_{56} = 0 \\ 1d_{51}^* + 0d_{52}^* + 2d_{53}^* + 3d_{54}^* + 5d_{55}^* + 2d_{56} = 0 \\ 2d_{51}^* + 3d_{52}^* + 0d_{53}^* + 5d_{54}^* + 2d_{55}^* + 7d_{56} = 0 \\ 2d_{51}^* + 3d_{52}^* + 4d_{53}^* + 0d_{54}^* + 2d_{55}^* + 1d_{56} = 0 \\ 1d_{51}^* + 0d_{52}^* + 3d_{53}^* + 4d_{54}^* + 0d_{55}^* + 1d_{56} = 3 \\ 2d_{51}^* + 3d_{52}^* + 1d_{53}^* + 7d_{54}^* + 0d_{55}^* + 0d_{56} = 0 \end{cases} \tag{5.5}$$

Similarly, the values of following equation (5.5) are,

$$d^*_{51} = 9 \mod 29$$
$$d^*_{52} = 2 \mod 29$$
$$d^*_{53} = 26 \mod 29$$
$$d^*_{54} = 26 \mod 29$$
$$d^*_{55} = 1 \mod 29$$
$$d^*_{56} = 15 \mod 29$$

To find the last component of D^*, we will multiply the sixth component of orthonormal basis D^* with all components of D one by one and making the following system of linear equations.

$$\begin{cases} 1d^*_{61} + 2d^*_{62} + 3d^*_{63} + 4d^*_{64} + 5d^*_{65} + 6d_{66} = 0 \\ 1d^*_{61} + 0d^*_{62} + 2d^*_{63} + 3d^*_{64} + 5d^*_{65} + 2d_{66} = 0 \\ 2d^*_{61} + 3d^*_{62} + 0d^*_{63} + 5d^*_{64} + 2d^*_{65} + 7d_{66} = 0 \\ 2d^*_{61} + 3d^*_{62} + 4d^*_{63} + 0d^*_{64} + 2d^*_{65} + 1d_{66} = 0 \\ 1d^*_{61} + 0d^*_{62} + 3d^*_{63} + 4d^*_{64} + 0d^*_{65} + 1d_{66} = 0 \\ 2d^*_{61} + 3d^*_{62} + 1d^*_{63} + 7d^*_{64} + 0d^*_{65} + 0d_{66} = 3 \end{cases} \tag{5.6}$$

Similarly, the values of following equation (5.6) are,

$$d^*_{61} = 27 \mod 29$$
$$d^*_{62} = 8 \mod 29$$
$$d^*_{63} = 26 \mod 29$$
$$d^*_{64} = 27 \mod 29$$
$$d^*_{65} = 1 \mod 29$$
$$d^*_{66} = 19 \mod 29$$

Therefore,

$$\begin{aligned} D^* = \{&(23,10,16,20,20,23),(21,9,18,0,25,12),(11,26,12,13,19,17),\\ &(24,28,19,24,15,26),(9,2,26,26,1,15),(27,8,26,27,1,19)\} \end{aligned} \tag{5.7}$$

Verification of orthonormal condition (4.2)

Now, we will verify the orthonormal condition (4.2), as we choose the random $r = 3$ so the dot product of bases are as follow:

$$\begin{aligned}
d_1 \cdot d_1^* &= (1,2,3,4,5,6) \cdot (23,10,16,20,20,23)\\
&= (1 \cdot 23 + 2 \cdot 10 + 3 \cdot 16 + 4 \cdot 20 + 5 \cdot 20 + 6 \cdot 23) = 409 \mod 29 = 3\\
d_1 \cdot d_2^* &= (1,2,3,4,5,6) \cdot (21,9,18,0,25,12)\\
&= (1 \cdot 21 + 2 \cdot 9 + 3 \cdot 18 + 4 \cdot 0 + 5 \cdot 25 + 6 \cdot 12) = 290 \mod 29 = 0\\
d_1 \cdot d_3^* &= (1,2,3,4,5,6) \cdot (11,26,12,13,19,17)\\
&= (1 \cdot 11 + 2 \cdot 26 + 3 \cdot 12 + 4 \cdot 13 + 5 \cdot 19 + 6 \cdot 17) = 348 \mod 29 = 0\\
d_1 \cdot d_4^* &= (1,2,3,4,5,6) \cdot (24,28,19,24,15,26)\\
&= (1 \cdot 24 + 2 \cdot 28 + 3 \cdot 19 + 4 \cdot 24 + 5 \cdot 15 + 6 \cdot 26) = 464 \mod 29 = 0\\
d_1 \cdot d_5^* &= (1,2,3,4,5,6) \cdot (9,2,26,26,1,15)\\
&= (1 \cdot 9 + 2 \cdot 2 + 3 \cdot 26 + 4 \cdot 26 + 5 \cdot 1 + 6 \cdot 15) = 290 \mod 29 = 0\\
d_1 \cdot d_6^* &= (1,2,3,4,5,6) \cdot (27,8,26,27,1,19)\\
&= (1 \cdot 27 + 2 \cdot 8 + 3 \cdot 26 + 4 \cdot 27 + 5 \cdot 1 + 6 \cdot 19) = 348 \mod 29 = 0
\end{aligned}$$

$$\begin{aligned}
d_2 \cdot d_1^* &= (1,0,2,3,5,2) \cdot (23,10,16,20,20,23)\\
&= (1 \cdot 23 + 0 \cdot 10 + 2 \cdot 16 + 3 \cdot 20 + 5 \cdot 20 + 2 \cdot 23) = 261 \mod 29 = 0\\
d_2 \cdot d_2^* &= (1,0,2,3,5,2) \cdot (21,9,18,0,25,12)\\
&= (1 \cdot 21 + 0 \cdot 9 + 2 \cdot 18 + 3 \cdot 0 + 5 \cdot 25 + 2 \cdot 12) = 206 \mod 29 = 3\\
d_2 \cdot d_3^* &= (1,0,2,3,5,2) \cdot (11,26,12,13,19,17)\\
&= (1 \cdot 11 + 0 \cdot 26 + 2 \cdot 12 + 3 \cdot 13 + 5 \cdot 19 + 2 \cdot 17) = 203 \mod 29 = 0\\
d_2 \cdot d_4^* &= (1,0,2,3,5,2) \cdot (24,28,19,24,15,26)\\
&= (1 \cdot 24 + 0 \cdot 28 + 2 \cdot 19 + 3 \cdot 24 + 5 \cdot 15 + 2 \cdot 26) = 261 \mod 29 = 0\\
d_2 \cdot d_5^* &= (1,0,2,3,5,2) \cdot (9,2,26,26,1,15)\\
&= (1 \cdot 9 + 0 \cdot 2 + 2 \cdot 26 + 3 \cdot 26 + 5 \cdot 1 + 2 \cdot 15) = 174 \mod 29 = 0\\
d_2 \cdot d_6^* &= (1,0,2,3,5,2) \cdot (27,8,26,27,1,19)\\
&= (1 \cdot 27 + 0 \cdot 8 + 2 \cdot 26 + 3 \cdot 27 + 5 \cdot 1 + 2 \cdot 19) = 203 \mod 29 = 0
\end{aligned}$$

$$\begin{aligned}
d_3 \cdot d_1^* &= (2,3,0,5,2,7) \cdot (23,10,16,20,20,23)\\
&= (2 \cdot 23 + 3 \cdot 10 + 0 \cdot 16 + 5 \cdot 20 + 2 \cdot 20 + 7 \cdot 23) = 377 \mod 29 = 0\\
d_3 \cdot d_2^* &= (2,3,0,5,2,7) \cdot (21,9,18,0,25,12)\\
&= (2 \cdot 21 + 3 \cdot 9 + 0 \cdot 18 + 5 \cdot 0 + 2 \cdot 25 + 7 \cdot 12) = 203 \mod 29 = 0\\
d_3 \cdot d_3^* &= (2,3,0,5,2,7) \cdot (11,26,12,13,19,17)\\
&= (2 \cdot 11 + 3 \cdot 26 + 0 \cdot 12 + 5 \cdot 13 + 2 \cdot 19 + 7 \cdot 17) = 322 \mod 29 = 3\\
d_3 \cdot d_4^* &= (2,3,0,5,2,7) \cdot (24,28,19,24,15,26)\\
&= (2 \cdot 24 + 3 \cdot 28 + 0 \cdot 19 + 5 \cdot 24 + 2 \cdot 15 + 7 \cdot 26) = 464 \mod 29 = 0\\
d_3 \cdot d_5^* &= (2,3,0,5,2,7) \cdot (9,2,26,26,1,15)\\
&= (2 \cdot 9 + 3 \cdot 2 + 0 \cdot 26 + 5 \cdot 26 + 2 \cdot 1 + 7 \cdot 15) = 261 \mod 29 = 0\\
d_3 \cdot d_6^* &= (2,3,0,5,2,7) \cdot (27,8,26,27,1,19)\\
&= (2 \cdot 27 + 3 \cdot 8 + 0 \cdot 26 + 5 \cdot 27 + 2 \cdot 1 + 7 \cdot 19) = 348 \mod 29 = 0
\end{aligned}$$

$$\begin{aligned}
d_4 \cdot d_1^* &= (2,3,4,0,2,1) \cdot (23,10,16,20,20,23) \\
&= (2 \cdot 23 + 3 \cdot 10 + 4 \cdot 16 + 0 \cdot 20 + 2 \cdot 20 + 1 \cdot 23) = 203 \mod 29 = 0 \\
d_4 \cdot d_2^* &= (2,3,4,0,2,1) \cdot (21,9,18,0,25,12) \\
&= (2 \cdot 21 + 3 \cdot 9 + 4 \cdot 18 + 0 \cdot 0 + 2 \cdot 25 + 1 \cdot 12) = 203 \mod 29 = 0 \\
d_4 \cdot d_3^* &= (2,3,4,0,2,1) \cdot (11,26,12,13,19,17) \\
&= (2 \cdot 11 + 3 \cdot 26 + 4 \cdot 12 + 0 \cdot 13 + 2 \cdot 19 + 1 \cdot 17) = 203 \mod 29 = 0 \\
d_4 \cdot d_4^* &= (2,3,4,0,2,1) \cdot (24,28,19,24,15,26) \\
&= (2 \cdot 24 + 3 \cdot 28 + 4 \cdot 19 + 0 \cdot 24 + 2 \cdot 15 + 1 \cdot 26) = 264 \mod 29 = 3 \\
d_4 \cdot d_5^* &= (2,3,4,0,2,1) \cdot (9,2,26,26,1,15) \\
&= (2 \cdot 9 + 3 \cdot 2 + 4 \cdot 26 + 0 \cdot 26 + 2 \cdot 1 + 1 \cdot 15) = 145 \mod 29 = 0 \\
d_4 \cdot d_6^* &= (2,3,4,0,2,1) \cdot (27,8,26,27,1,19) \\
&= (2 \cdot 27 + 3 \cdot 8 + 4 \cdot 26 + 0 \cdot 27 + 2 \cdot 1 + 1 \cdot 19) = 203 \mod 29 = 0
\end{aligned}$$

$$\begin{aligned}
d_5 \cdot d_1^* &= (1,0,3,4,0,1) \cdot (23,10,16,20,20,23) \\
&= (1 \cdot 23 + 0 \cdot 10 + 3 \cdot 16 + 4 \cdot 20 + 0 \cdot 20 + 1 \cdot 23) = 174 \mod 29 = 0 \\
d_5 \cdot d_2^* &= (1,0,3,4,0,1) \cdot (21,9,18,0,25,12) \\
&= (1 \cdot 21 + 0 \cdot 9 + 3 \cdot 18 + 4 \cdot 0 + 0 \cdot 25 + 1 \cdot 12) = 87 \mod 29 = 0 \\
d_5 \cdot d_3^* &= (1,0,3,4,0,1) \cdot (11,26,12,13,19,17) \\
&= (1 \cdot 11 + 0 \cdot 26 + 3 \cdot 12 + 4 \cdot 13 + 0 \cdot 19 + 1 \cdot 17) = 116 \mod 29 = 0 \\
d_5 \cdot d_4^* &= (1,0,3,4,0,1) \cdot (24,28,19,24,15,26) \\
&= (1 \cdot 24 + 0 \cdot 28 + 3 \cdot 19 + 4 \cdot 24 + 0 \cdot 15 + 1 \cdot 26) = 203 \mod 29 = 0 \\
d_5 \cdot d_5^* &= (1,0,3,4,0,1) \cdot (9,2,26,26,1,15) \\
&= (1 \cdot 9 + 0 \cdot 2 + 3 \cdot 26 + 4 \cdot 26 + 0 \cdot 1 + 1 \cdot 15) = 206 \mod 29 = 3 \\
d_5 \cdot d_6^* &= (1,0,3,4,0,1) \cdot (27,8,26,27,1,19) \\
&= (1 \cdot 27 + 0 \cdot 8 + 3 \cdot 26 + 4 \cdot 27 + 0 \cdot 1 + 1 \cdot 19) = 232 \mod 29 = 0
\end{aligned}$$

$$
\begin{aligned}
d_6 \cdot d_1^* &= (2,3,1,7,0,0) \cdot (23,10,16,20,20,23) \\
&= (2 \cdot 23 + 3 \cdot 10 + 1 \cdot 16 + 7 \cdot 20 + 0 \cdot 20 + 0 \cdot 23) = 232 \mod 29 = 0 \\
d_6 \cdot d_2^* &= (2,3,1,7,0,0) \cdot (21,9,18,0,25,12) \\
&= (2 \cdot 21 + 3 \cdot 9 + 1 \cdot 18 + 7 \cdot 0 + 0 \cdot 25 + 0 \cdot 12) = 87 \mod 29 = 0 \\
d_6 \cdot d_3^* &= (2,3,1,7,0,0) \cdot (11,26,12,13,19,17) \\
&= (2 \cdot 11 + 3 \cdot 26 + 1 \cdot 12 + 7 \cdot 13 + 0 \cdot 19 + 0 \cdot 17) = 203 \mod 29 = 0 \\
d_6 \cdot d_4^* &= (2,3,1,7,0,0) \cdot (24,28,19,24,15,26) \\
&= (2 \cdot 24 + 3 \cdot 28 + 1 \cdot 19 + 7 \cdot 24 + 0 \cdot 15 + 0 \cdot 26) = 319 \mod 29 = 0 \\
d_6 \cdot d_5^* &= (2,3,1,7,0,0) \cdot (9,2,26,26,1,15) \\
&= (2 \cdot 9 + 3 \cdot 2 + 1 \cdot 26 + 7 \cdot 26 + 0 \cdot 1 + 0 \cdot 15) = 232 \mod 29 = 0 \\
d_6 \cdot d_6^* &= (2,3,1,7,0,0) \cdot (27,8,26,27,1,19) \\
&= (2 \cdot 27 + 3 \cdot 8 + 1 \cdot 26 + 7 \cdot 27 + 0 \cdot 1 + 0 \cdot 19) = 293 \mod 29 = 3
\end{aligned}
$$

From above calculations, it is verified that (5.7) are the orthonormal bases. Now, by using the orthonormal bases, we will calculate message encryption and decryption by using the IBBE scheme defined in Chapter 4.

1. **Setup:** Setup algorithm calculate the master key of IBBE system. For that purpose we take $m = 6$ denotes the number of receivers, now we assume that PKG first randomly chooses orthonormal bases of $\mathbb{Z}_{29}^6$ that are calculated above
   $D = \{(1,2,3,4,5,6), (1,0,2,3,5,2), (2,3,0,5,2,7),$
   $(2,3,4,0,2,1), (1,0,3,4,0,1), (2,3,1,7,0,0)\}$
   $D^* = \{(23,10,16,20,20,23), (21,9,18,0,25,12), (11,26,12,13,19,17),$
   $(24,28,19,24,15,26), (9,2,26,26,1,15), (27,8,26,27,1,19)\}$ The master key is defined as
   $$\mathcal{K} = \{g^{\alpha\theta d_1^*}, g^{\theta d_1^*}, g^{\theta d_2^*}, g^{\sigma d_3^*}, g^{\sigma d_4^*}\}$$

PKG chooses randomly $\alpha = 3, \theta = 7, \sigma = 9 \in \mathbb{Z}_{29}$. First we will calculate the public parameters that are defined as

$$\mathcal{PP} = \{\mathbb{G}_1, \mathbb{G}_2, g, p, \phi(g,g)^{\alpha\theta d_1 d_1^*}, g^{d_1}, g^{d_2}, g^{d_3}, g^{d_4}\}$$

In this example pubic parameter are

$$\begin{aligned}
\mathcal{PP} = \{&\mathbb{G}_1 = E_{29}(0,1), \mathbb{G}_2 = \mathbb{F}_{29^2}, g = (8,7), p = 29,\\
&\phi(g,g)^{\alpha\theta d_1 d_1^*} = \phi((8,7),(8,7))^{3\times 7(1,2,3,4,5,6)(23,10,16,20,20,23)} = 12\beta + 3,\\
&g^{d_1} = (8,7)^{(1,2,3,4,5,6)} = \{((8,7),(4,23),(4,6),(8,22),(8,7),(8,7)),\}\\
&g^{d_2} = g^{(1,0,2,3,5,2)} = \{(8,7),(8,7),(4,23),(4,6),(8,7),(4,23)\}\\
&g^{d_3} = g^{(2,3,0,5,2,7)} = \{(4,23),(4,6),(8,7),(8,7),(4,23),(4,23)\},\\
&g^{d_4} = g^{(2,3,4,0,2,1)} = \{(4,23),(4,6),(8,22),(8,7),(4,23),(8,7)\}
\end{aligned}$$

So calculation for master key is as follow:

$$\begin{aligned}
\mathcal{K} = \{&(8,7)^{(3)(7)(23,10,16,20,20,23)}, (8,7)^{(7)(23,20,16,20,20,23)}, (8,7)^{(7)(21,9,18,0,25,12)}\\
&, (8,7)^{(9)(11,26,12,13,19,17)}, (8,7)^{(9)(24,28,19,24,15,26)}\}\\
\mathcal{K} = \{&(8,7)^{(21)(23,10,16,20,20,23)}, (8,7)^{(7)(23,20,16,20,20,23)}, (8,7)^{(7)(21,9,18,0,25,12)}\\
&, (8,7)^{(9)(11,26,12,13,19,17)}, (8,7)^{(9)(24,28,19,24,15,26)}\}\\
\mathcal{K} = \{&(8,7)^{(483,210,336,420,420,483)}, (8,7)^{(161,70,112,140,140,161)}\\
&, (8,7)^{(147,63,126,0,175,84)}, (8,7)^{(99,234,108,117,171,153)}, (8,7)^{(216,252,171,216,135,234)}\}
\end{aligned}$$

As we work on elliptic curve so we write the above equation as:

$$\begin{aligned}
\mathcal{K} = \{&(483(8,7), 210(8,7), 336(8,7), 420(8,7), 420(8,7), 483(8,7)), (161(8,7),\\
&70(8,7), 112(8,7), 140(8,7), 140(8,7), 161(8,7)), (147(8,7), 63(8,7), 126(8,7)\\
&0(8,7), 175(8,7), 84(8,7)), (99(8,7), 234(8,7), 108(8,7), 117(8,7), 171(8,7),\\
&153(8,7)), (216(8,7), 252(8,7), 171(8,7), 216(8,7), 135(8,7), 234(8,7))\}
\end{aligned}$$

Using the scalar multiplication of elliptic curve, calculating them using ApCoCoA program see Appendix A. The above equation can be rewritten as.

$$\begin{aligned}\mathcal{K} = \{&((4,6),\mathcal{O},(8,7),\mathcal{O},\mathcal{O},(4,6)),((8,7),\mathcal{O},(4,23),\mathcal{O},\mathcal{O},(8,7)),\\ &((4,23),(4,6),(8,7),\mathcal{O},\mathcal{O},(8,22),((8,22),(8,22),(4,6),(4,23),(8,7),(4,6))\\ &,((8,7),(4,23),(8,7),(8,7),\mathcal{O},(8,22))\}\end{aligned}$$

2. **Extract:** Now the PKG calculates the private key of the corresponding identity. $\mathcal{K}_1$ is defined as

$$\mathcal{K}_1 = g^{\alpha\theta d_1^* + r_1^i ID_i \theta d_1^* - r_1^i \theta d_2^* + r_2^i ID_i \sigma d_3^* - r_2^i \sigma d_4^*}$$

Let set of Identities be $\Omega = \{2,5,7,3,14,13\} \in \mathbb{Z}_{29}$ and choose random integers

$$\begin{aligned} r_1^1, r_1^2, r_1^3, r_1^4, r_1^5 r_1^6 &= 2^1, 2^2, 2^3, 2^4, 2^5, 2^6 \in \mathbb{Z}_{29}^* \\ &= (2,4,8,16,32,64) \mod 29 \\ &= (2,4,8,16,3,6) \mod 29 \end{aligned}$$

$$\begin{aligned} r_2^1, r_2^2, r_2^3, r_2^4, r_2^5, r_2^6 &= 3^1, 3^2, 3^3, 3^4, 3^5, 3^6 \in \mathbb{Z}_{29}^* \\ &= (3,9,27,81,243,729) \mod 29 \\ &= (3,9,27,23,11,4) \mod 29 \end{aligned}$$

Pick up the second identity $ID_2 = 5 \in \Omega$ where $i = 2$.Therefore, calculation for $\mathcal{K}_1$ is as follow:

$$\begin{aligned} \mathcal{K}_1 = (8,7)^{(3)(7)(23,10,16,20,20,23)+(4)(5)(7)(23,10,16,20,20,23)-(4)(7)(21,9,18,0,25,12)} \\ .(8,7)^{(9)(5)(9)(11,26,12,13,19,17)-(9)(9)(24,28,19,24,15,26)} \end{aligned}$$

Multiplying the exponents, we get

$$\mathcal{K}_1 = (8,7)^{(483,210,336,420,420,483)+(3220,1400,2240,2800,2800,3220)+(-588,-252,-504,0,-700,-336)}$$
$$(8,7)^{(4455,10530,4860,5265,7695,6885)+(-1944,-2268,-1539,-1944,-1215,-2106)} \mod 29$$

As base is same then by using the law of exponent $x^i.x^j = x^{i+j}$ and adding them component wise, and reducing the answer in $\mod 29$ therefore we get

$$\mathcal{K}_1 = (8,7)^{(5626,9620,5393,6541,9000,8146)} \mod 29$$

Using Scalar multiplication of elliptic curve the above equation can be rewritten as.

$$\mathcal{K}_1 = 5626(8,7), 9620(8,7), 5393(8,7), 6541(8,7), 9000(8,7), 8146(8,7)$$

Now using scalar multiplication of elliptic curve using ApCoCoA tool see Appendix A, we get

$$\mathcal{K}_1 = \{(8,7), \mathcal{O}, (4,6), (8,7), \mathcal{O}, (8,7)\}$$

$\mathcal{K}_2$ is defined as

$$\mathcal{K}_2 = \begin{cases} g^{(r_1^1+r_1^2+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)(ID_1+ID_2+\ldots,ID_6)\theta d_1^*}. \\ g^{r_1^i(ID_1+ID_2+\cdots+ID_{i-1}+ID_{i+1}+\ldots ID_6)\theta d_1^*}. \\ g^{-(r_1^1+r_1^2+\cdots+r_1^{i-1}+r_1^{i+1}+\cdots+r_1^6)\theta d_2^*}. \\ g^{(r_2^1+r_2^2+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)(ID_1+ID_2+\ldots,ID_6)\sigma d_3^*}. \\ g^{r_2^i(ID_1+ID_2+\cdots+ID_{i-1}+ID_{i+1}+\cdots+ID_6)\sigma d_3^*}. \\ g^{-(r_2^1+r_2^2+\cdots+r_2^{i-1}+r_2^{i+1}+\cdots+r_2^6)\sigma d_4^*} \end{cases}$$

The calculation for $\mathcal{K}_2$ given below, we assume that $i = 2$

$$\mathcal{K}_2 = \begin{cases} (8,7)^{(2+8+16+3+6)(2+5+7+3+14+13)(7)(23,10,16,20,20,23)} & \text{Skiping the } r_1^2 = 4 \\ .(8,7)^{(4)(2+7+3+14+13)(7)(23,10,16,20,20,23)} & \text{Skiping the } ID_2 = 5 \\ .(8,7)^{-(2+8+16+3+6)(7)(21,9,18,0,25,12)} & \text{Skiping the } r_1^2 = 4 \\ .(8,7)^{(3+27+23+11+4)(2+5+7+3+14+13)(9)(11,26,12,13,19,17)} & \text{Skiping the } r_2^2 = 9 \\ .(8,7)^{(9)(2+7+3+14+13)(9)(11,26,12,13,19,17)} & \text{Skiping the } ID_2 = 5 \\ .(8,7)^{-(3+27+23+11+4)(9)(24,28,19,24,15,26)} & \text{Skiping the } r_2^2 = 9 \end{cases}$$

Solving the exponents

$$\mathcal{K}_2 = \begin{cases} (8,7)^{(247940,107800,172480,215600,215600,247940)}.(8,7)^{(25116,10920,17472,21840,21840,25116)} \\ .(8,7)^{(-5145,-2205,-4410,0,-6125,-2940)}.(8,7)^{(296208,700128,323136,350064,511632,457776)} \\ .(8,7)^{(34749,82134,37908,41067,60021,53703)}.(8,7)^{(-14688,-17136,-11628,-14688,-9180,-15912)} \end{cases}$$

As the base is same which is $(8,7)$ then using the law of exponent that is stated as $x^i.x^j = x^{i+j}$ we add them component wise.

$$\mathcal{K}_2 = (8,7)^{(584180,881641,534958,613883,793788,765683)} \mod 29$$

By using the scalar multiplication of elliptic curve

$$\mathcal{K}_2 = \{584180(8,7), 881641(8,7), 534958(8,7), 613883(8,7), 793788(8,7), 765683(8,7)\}$$
$$\mathcal{K}_2 = \{(8,22), (4,23), (4,6), (8,22), (8,22), (8,22)\}$$

Similarly, by using the above procedure one can calculate private keys for all other identities $\Omega = \{5, 7, 3, 14, 13\}$

3. **Encrypt:** C_1 is defined as

$$C_1 = M.\phi(g,g)^{\alpha\theta s_1 d_1 \cdot d_1^*}$$

Choose randomly $s_1 = 21, s_2 = 23 \in \mathbb{Z}^*_{29}$ and take message $M = 9$.

For calculation of C_1 we need to calculate the bilinear mapping ϕ. As we use the modified weil pairing and choose point $(8, 7)$ of order 5. Therefore, we calculate $\hat{\phi_n}(A, A) = \hat{\phi_5}((8, 7), (8, 7))$. Then putting the values in (4.9), we get.

$$C_1 = 9.\hat{\phi_5}((8,7),(8,7))^{(3)(7)(21)(1,2,3,4,5,6)\cdot(23,10,16,20,20,23)}$$

Using ApCoCoA program see Appendix B the modified weil pairing $\hat{\phi_5}((8, 7), (8, 7)) = (15\beta + 10)$

$$C_1 = 9.(15\beta + 10)^{(3)(7)(21)(1,2,3,4,5,6)\cdot(23,10,16,20,20,23)}$$

The dot product $d_1 \cdot d_1^* = 3 \mod 29$, then multiplying the exponents in mod 29 we get

$$C_1 = 9.(15\beta + 10)^{(3)(7)(21)(3)}$$
$$C_1 = (9.(15\beta + 10)^{1323} \mod \beta^2 + \beta + 1) \mod 29$$
$$C_1 = 9.(12\beta + 3) \mod 29$$
$$C_1 = 108b + 27 \mod 29$$
$$C_1 = 21\beta + 27 \mod 29$$

C_2 is define as

$$C_2 = g^{s_1 d_1 + s_1(ID_1, ID_2, ..., ID_6)d_2 + s_2 d_3 + s_2(ID_1, ID_2, ..., ID_6)d_4}$$

Now we have to calculate C_2 by using the equation (4.10). Putting values in (4.10) we get

$$C_2 = (8,7)^{21(1,2,3,4,5,6)+(21)(2+5+7+3+14+13)(1,0,2,3,5,2)}.$$
$$(8,7)^{(23)(2,3,0,5,2,7)+(23)(2+5+7+3+14+13)(2,3,4,0,2,1)}$$

Solving the exponents and adding them component wise.

$$\begin{aligned} C_2 &= (8,7)^{(21,42,63,84,105,126)+(924,0,1848,2772,4620,1848)}\cdot \\ &\quad (8,7)^{(46,69,0,115,46,161)+(2024,3036,4048,0,2024,1012)} \\ C_2 &= (8,7)^{(3015,3147,5959,2971,6795,3147)} \end{aligned}$$

By using scalar multiplication of elliptic curve

$$\begin{aligned} C_2 &= (3015(8,7), 3147(8,7), 5959(8,7), 2971(8,7), 6795(8,7), 3147(8,7)) \mod 29 \\ C_2 &= \{\mathcal{O}, (4,23), (8,22), (8,7), \mathcal{O}, (4,23)\} \end{aligned}$$

4. **Decrypt:** Message decryption is define as $M = C_1/\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2)$. For decryption, we use both decryption keys and ciphertexts that is calculated as below.

$$\begin{aligned} \mathcal{K}_1 &= \{(8,7), (8,7), (4,6), (8,7), \mathcal{O}, (8,7)\} \\ \mathcal{K}_2 &= \{(8,22), (4,23), (8,22), (8,7), (8,7), \mathcal{O}\} \\ C_1 &= 21\beta + 27 \\ C_2 &= \{\mathcal{O}, (4,23), (8,22), (8,7), \mathcal{O}, (4,23)\} \end{aligned}$$

From the definition of message decryption (4.11). First we calculate the denominator of equation (4.11) for that purpose we need to calculate the product of pairing that is denoted by $\phi_m = \phi_6$ between both decryption keys and second ciphertext C_2 as defined as $\phi_6(\mathcal{K}_1\mathcal{K}_2, C_2)$.
As a first step we calculate the product of decryption keys but as we are using elliptic curves that is group of addition then we add the both keys component wise using the ApCoCoA (see Appendix A for point addition). The result of addition is given below.

$$\mathcal{K}_1 + \mathcal{K}_2 = \{(8,7), (8,7), (8,7), (8,22), (4,6), (8,22)\}$$

We use the modified weil pairing between the keys and ciphertext which is denoted by $\hat{\phi_m} = \hat{\phi_6}$. pairing component wise

$$\hat{\phi_m}(\mathcal{K}_1 + \mathcal{K}_2, C_2) = \begin{cases} \hat{\phi}((8,7), \mathcal{O}).\hat{\phi}((8,7),(4,23)).\hat{\phi}((8,7),(8,22)). \\ \hat{\phi}((8,22),(8,7)).\hat{\phi}((4,6),\mathcal{O}).\hat{\phi}((8,22),(4,23)) \end{cases}$$

Applying the definition of modified weil pairing, we get

$$\phi(\mathcal{K}_1 + \mathcal{K}_2, C_2) = \begin{cases} \phi((8,7), \mathcal{O}) \cdot \phi((8,7),(4\beta,23)) \cdot \phi((8,7),(8\beta,22)) \cdot \\ \phi((8,22),(8\beta,7)) \cdot \phi((4,6),\mathcal{O}) \cdot \phi((8,22),(4\beta,23)) \end{cases}$$

Calculate the pairing between the points and multiplying the answer by using usual multiplication. As we work on extension field we reduced our answer in $\mod \beta^2 + \beta + 1$, we get

$$\begin{aligned} \phi(\mathcal{K}_1 + \mathcal{K}_2, C_2) &= ((1) \cdot (17\beta) \cdot (14\beta + 24) \cdot \\ &\qquad (14\beta + 24) \cdot (1) \cdot (12\beta + 3) \mod \beta^2 + \beta + 1) \mod 29 \\ \phi(\mathcal{K}_1 + \mathcal{K}_2, C_2) &= (12\beta + 3) \end{aligned}$$

Putting value in message, we will get

$$M = (21\beta + 27)/(12\beta + 3)$$

Using extended euclidean Algorithm, we have $(17\beta + 20)^{-1} = 12\beta + 3 \mod \beta^2 + \beta + 1$. So,

$$\begin{aligned} M &= (21\beta + 27) \cdot (17\beta + 20) \\ M &= (357\beta^2 + 879\beta + 540 \mod \beta^2 + \beta + 1) \mod 29 \\ M &= 522\beta + 183 \mod 29 = 9 \end{aligned}$$

5.2 Security Analysis

- **Discrete log problem:** The discrete log problem states that if $g, a \in G_1$ there exist an integer $b \in \{0, 1 \ldots p-1\}$, where p is prime and order of group G_1
$$g^b = a \mod p$$
. If we have g^b then it is hard to find b. Some small size of group it is easy by using brute force attack but it is still a problem for large order group.

- **Elliptic Curve Discrete log problem** Given point $A \in E_{\mathbb{F}}(a, b)$ and $y \in \mathbb{Z}$ there exist an integer $n \in \mathbb{Z}$ such that
$$y = nA$$
if someone has information of y and A but it is still difficult to calculate the n for the information.

In our construction Weil pairing apply the discrete log problem as if we given calculated value of $\phi(g, g)^{\alpha\theta d_1 d_1^*}$ then it is difficult to calculate the exponent=$\alpha\theta d_1 d_1^*$. Similarly, In our construction, given $g, y \in E_{\mathbb{F}}(a, b)$ and calculate $y = \alpha\theta d_1^* g$ it is very difficult to calculate $\alpha\theta d_1^*$ by only knowing y, g.

5.3 Conclusion

In this thesis, we review the research paper of "IBBE with group of prime order" [37] this scheme was introduced by Ming and Wang that is based on bilinear groups. We modified this scheme replaces group of prime order with group of points of elliptic curve. In this view setting we use weil pairing as a considerable for bilinear mapping. for the implementation, we develop computer programs for computation with elliptic curve using the platform of computer algebra system ApCoCoA[1]. Using weil pairing, we give the two examples of proposed scheme and efficiently compute the encryption and decryption of message.

One can extend our work by changing the type of curve that is $y^2 = x^3 + ax$ where $a \in \mathbb{Z}$ so that the distortion mapping can also change and defined as $\phi : (x, y) = (-x, iy)$ that is available in [29]. One can also use tate pairing see literature [23, 33].

Appendix A

Weil Pairing

A.1 ApCoCoA Code for Weil Pairing

This section contain the ApCoCoA code for calculation of weil pairing in finite field $\mathbb{F}_p$. It consists of **ModInv**, **ECadd**,**NP**, **OrderP**,**MillerFtn**, **MillerCal**, **Weil**.

ModInv calculate the inverse of a number under the mod . It require the input N, P where P is number and N is mod . This function uses the extended euclidean inverse algorithm.

```
Define ModInv(N,P)
A1:=1;A2:=0;A3:=P;
B1:=0;B2:=1;B3:=N;
While B3<0 Do
B3:=B3+P;
EndWhile;
While B3<>1 Do
Q:=Div(A3,B3);
If Q=0 Then Error(" Q is 0");EndIf;
T1:=A1-Q*B1;T2:=A2-Q*B2;T3:=A3-Q*B3;
A1:=B1;A2:=B2;A3:=B3;
```

```
B1:=T1;B2:=T2;B3:=T3;
If B2<0 Then B2:=B2+P; EndIf;
If B3=1  Then Return B2;EndIf;
If B3=0 Then Return("Not Invertible!"); EndIf;
EndWhile;
Return B2;
EndDefine;
```

Dec2Bin(D) converts the decimal number into binary. It calls in **MillerFtn** and **ModMillerFtn**

```
Define Dec2Bin(D)
L:=[];Q:=1;Rem:=0;
While Q<>0 Do
Q:=Div(D,2);Rem:=Mod(D,2);
Append(L,Rem);
D:=Q;
EndWhile;
Return Reversed(L);
EndDefine;
```

ECadd is use to add the points on P1 and P2 on curve C mod M. It require P1, P2, C, M where P1, P2 should be given as the list of integers x and y coordinates of the points. C is list containing a and b of elliptic curve.

```
Define ECadd(P1,P2,C,M)
If P1="Infinity"  Then Return P2; EndIf;
If P2="Infinity"  Then Return P1; EndIf;
If Mod(P1[2]^2,M)<>Mod(P1[1]^3+C[1]*P1[1]+C[2],M) Then
Error("First Point is not on the Elliptic Curve");
EndIf;
If Mod(P2[2]^2,M)<>Mod(P2[1]^3+C[1]*P2[1]+C[2],M) Then
Error("Second Point is not on the Elliptic Curve");
EndIf;
```

```
If P1[1]=P2[1] AND P1[2]=Mod(-P2[2],M)
Then Return "Infinity";
EndIf;
If P1=P2 Then
S:=Mod((3P1[1]^2+C[1])*ModInv(2P1[2],M),M);
Else
S:=Mod((P2[2]-P1[2])*ModInv(P2[1]-P1[1],M),M);
EndIf;
Y0:=P1[2]-S*P1[1];
XR:=Mod(S^2-P1[1]-P2[1],M);
YR:=Mod(-S*XR-Y0,M);
Return [XR,YR];
EndDefine;
```

NP calculates the scalar product of point P on curve C mod M. It require the input N,P,C,M where N is integer P is the point. It calculate N times P.

```
Define NP(N,P,C,M)
S:=P;
For I:=1 To N-1 Do
S:=ECadd(S,P,C,M);
EndFor;
Return S;
EndDefine;
```

OrderP calculates the order of point P on curve C mod M.

```
Define OrderP(P,C,M)
N:=1;
P1:=P;
While P1<> "Infinity" Do
P1:=ECadd(P1,P,C,M);
N:=N+1;
PrintLn("P^",N,"= ",P1);
```

```
EndWhile;
Return N;
EndDefine;
```

MillerFtn find the miller function $f_{A,B}$ define in section 3.2.2. In this code A=P1 and B=P2 points should be given as the list of integers x and y coordinates of the points.

```
Define MillerFtn(P1,P2,C,M)
If P1="Infinity"  Then Return (x - Mod(P2[1],M)); EndIf;
If P2="Infinity"  Then Return (x - Mod(P2[1],M)); EndIf;
If Mod(P1[2]^2,M)<>Mod(P1[1]^3+C[1]*P1[1]+C[2],M) Then
Error("First Point is not on the Elliptic Curve");
EndIf;
If Mod(P2[2]^2,M)<>Mod(P2[1]^3+C[1]*P2[1]+C[2],M) Then
Error("Second Point is not on the Elliptic Curve");
EndIf;
If P1[1]=P2[1] AND P1[2]=Mod(-P2[2],M) Then Return (x - Mod(P1[1],M));
EndIf;
If P1=P2  Then
S := Mod((3 * P1[1]^2 + C[1]) * ModInv(2 * P1[2],M),M);
F:=(y - P1[2] - S*(x - P1[1]))/(x + P1[1] + P2[1] - Mod(S^2,M));
EndIf;
If P1<>P2 Then S:=Mod((P2[2] - P1[2]) * ModInv(P2[1] - P1[1],M),M);
F:=(y - P1[2] - S*(x - P1[1]))/(x + P1[1] + P2[1] - Mod(S^2,M));
EndIf;
PrintLn("F = ",PolyMod(F,M));
Return PolyMod(F,M);
EndDefine;
```

MillerCal calculate the miller function by using Miller Algorithm 3.2.3. It requires the input N, P, C, M where P is point on Curve C mod M and N is order of P.

```
Define MillerCal(N,P,C,M)
F:=1;
T:=P;
K:=Reversed(Dec2Bin(N));
PrintLn("K=",K,"Len(K)",Len(K));
For I := Len(K)-1 To 1 Step -1 Do
F:=F * F * MillerFtn(T,T,C,M);
PrintLn("I = ",I, " and N = ",N);
F:=PolyMod(F,M);
PrintLn("F = ",F);
T:=NP(2,T,C,M);
PrintLn("T = ",T);
If K[I]=1 Then
PrintLn("KI = ",K[I]);
F:=F*MillerFtn(T,P,C,M);
F:=PolyMod(F,M);
PrintLn("F = ",F);
T:= ECadd(T,P,C,M);
PrintLn("T = ",T);
EndIf;
EndFor;
Return PolyMod(F,M);
EndDefine;
```

Finally,**Weil** uses the **MillerCal** function and evaluate the Weil pairing between A and B se section 3.2. In this code, we take $A = P$, $B = Q$ and $C = S$ points on Curve C mod M and N is order of P and Q.

```
Define Weil(N,P,Q,S,C,M);
Sinv:=[S[1],-S[2]];
PrintLn("Calculations for Point P=",P);
A:=MillerCal(N,P,C,M);
QS:=ECadd(Q,S,C,M);
If Type(A)=RATFUN Then
```

```
Fp1:=Mod(Eval(A.Num, QS),M)*ModInv(Mod(Eval(A.Den,QS),M),M);

Fp2:=Mod(Eval(A.Num, S),M)*ModInv(Mod(Eval(A.Den,S),M),M);

WP1:=Mod(Fp1*ModInv(Fp2,M),M);

Else

Fp1:=Mod(Eval(A,QS),M); Fp2:=Mod(Eval(A, S),M);

WP1:=Mod(Fp1*ModInv(Fp2,M),M);

EndIf;

PrintLn("NumWP =",WP1);

PrintLn("Calculations for Point Q=",Q);

B:=MillerCal(N,Q,C,M);

QS2:=ECadd(P,Sinv,C,M);

If Type(B)=RATFUN Then

Fq1:=Mod(Eval(B.Num, QS2),M)*ModInv(Mod(Eval(B.Den,QS2),M),M);

Fq2:=Mod(Eval(B.Num, Sinv),M)*ModInv(Mod(Eval(B.Den,Sinv),M),M);

WP2:=Mod(Fq1*ModInv(Fq2,M),M);

Else

Fq1:=Mod(Eval(B, QS2),M); Fq2:=Mod(Eval(B, Sinv),M);

WP2:=Mod(Fq1*ModInv(Fq2,M),M);

EndIf;

PrintLn("DenWP =",WP2);

WP:=Mod(WP1*ModInv(WP2,M),M);

Return WP;

EndDefine;
```

Appendix B

Modified Weil Pairing

B.1 ApCoCoA Code for Modified Weil Pairing

This section contain the ApCoCoA code for calculation of modified Weil pairing in finite field extension $\mathbb{F}_{p^2}$. It consists of **PolyMod**, **PolyInvM**, **ModEcAdd**, **ModNP**, **OrderP**, **ModMillerFtn**, **ModMillerCal**, **ModWeil** all function have same purpose that are define in **Appendix A**. But it work on finite field extension $\mathbb{F}_{p^2}$ instead of finite field $\mathbb{F}_p$. Where p is mod, in this code we take $p = M$.

OrderP calculates the order of any point P on curve C mod M in extension field $\mathbb{F}_{p^2}$.

```
Define OrderP(P,C,M)
N:=1;
P1:=P;
While P1<> "Infinity" Do
P1:=ModEcAdd(P1,P,C,M);
N:=N+1;
PrintLn("P^",N,"= ",P1);
EndWhile;
Return N;
```

```
EndDefine;
```

IsInEM use to check the point P lei on curve C mod M.

```
Define IsInEM(P,C,M)
Return PolyMod(NR(P[2]^2,[b^2+b+1]),M)=PolyMod(NR(P[1]^3+C[1]
*P[1]+C[2],[b^2+b+1]),M);
EndDefine;
```

PolyMod gives the polynomial F that is reduced on mod M.

```
Define PolyMod(F,M)
If Type(F)=RATFUN Then
If Mod(Den(LC(F.Num)),M)=0 Then
D:=Den(LC(F.Num));
D2:=D*F.Den-D*LPP(F.Den);
If D2= 0 Then Error("Zero Denominator . . .");EndIf;
F:=D*F.Num/(D2);
Return PolyMod(F,M);
EndIf;
CoefNum:=Coefficients(F.Num);CoefDen:=Coefficients(F.Den);
For I:= 1 To Len(CoefNum) Do
If Type(CoefNum[I])=RAT Then
CoefNum[I]:=Mod(CoefNum[I].Num*ModInv(CoefNum[I].Den,M),M);
Else
CoefNum[I]:=Mod(CoefNum[I],M);
EndIf;
EndFor;
For I:= 1 To Len(CoefDen) Do
If Type(CoefDen[I])=RAT Then
CoefDen[I]:=Mod(CoefDen[I].Num*ModInv(CoefDen[I].Den,M),M);
Else
CoefDen[I]:=Mod(CoefDen[I],M);
EndIf;
```

```
EndFor;
NewNum:=ScalarProduct(CoefNum,Support(F.Num));
NewDen:=ScalarProduct(CoefDen,Support(F.Den));
If NewDen= 0 Then Error("Zero Denominator . . .");EndIf;
Return NewNum/NewDen;
EndIf;
Coef:=Coefficients(F);
For I:= 1 To Len(Coef) Do
If Type(Coef[I])=RAT Then
Coef[I]:=Mod(Coef[I].Num*ModInv(Coef[I].Den,M),M);
Else
Coef[I]:=Mod(Coef[I],M);
EndIf;
EndFor;
Return ScalarProduct(Coef,Support(F));
EndDefine;
```

PolyInvM calculates inverse of polynomial F on polynomial M under mod Md using Extended Euclidean Inverse.

```
Define PolyInvM(F,M,Md)
F:=NR(F,[M]);
If MakeSet(Log(F))=[0] Then Return ModInv(LC(F),Md); EndIf;
A1:=1;A2:=0;A3:=PolyMod(M,Md);
B1:=0;B2:=1;B3:=PolyMod(F,Md);
While MakeSet(Log(B3))<>[0] Do
D:=DivAlg(A3,[B3]);
Q:=D.Quotients[1];
Coef:=Coefficients(Q);
For I:= 1 To Len(Coef) Do
C:=Coef[I];
Coef[I]:=Mod(C.Num*ModInv(C.Den,Md),Md);
EndFor;
Q:= ScalarProduct(Coef,Support(Q));
```

```
If Q=0 Then Error(" Q is 0");EndIf;
T1:=PolyMod(A1-Q*B1,Md);
T2:=PolyMod(A2-Q*B2,Md);
T3:=PolyMod(A3-Q*B3,Md);
A1:=B1;A2:=B2;A3:=B3;
B1:=T1;B2:=T2;B3:=T3;
If B3=1  Then
Return PolyMod(B2,Md);
EndIf;
EndWhile;
If B3<>1 Then
Return PolyMod(NR(ModInv(LC(B3),Md)*B2,[M]),Md);
Else
Return PolyMod(B2,Md);
EndIf;
EndDefine;
```

ModEcAdd is use to add the points on P1 and P2 on curve C mod M. It require P1, P2, C, M where P1, P2 should be given as the list of integers x and y coordinates of the points. C is list containing a and b of elliptic curve. Note that one point should be polynomial and written in form of say $P1 := [bx, y]$

```
Define ModEcAdd(P1,P2,C,M)
If P1="Infinity"  Then Return P2; EndIf;
If P2="Infinity"  Then Return P1; EndIf;
If PolyMod(NR(Poly(P1[2]^2),[b^2+b+1]),M)<>PolyMod(NR(Poly(P1[1]^3+
C[1]*P1[1]+C[2]),[b^2+b+1]),M) Then
Error("First Point is not on the Elliptic Curve");
EndIf;
If PolyMod(NR(Poly(P2[2]^2),[b^2+b+1]),M)<>PolyMod(NR(Poly(P2[1]^3+
C[1]*P2[1]+C[2]),[b^2+b+1]),M) Then
Error("Second Point is not on the Elliptic Curve");
EndIf;
If P1[1]=P2[1] AND PolyMod(Poly(P1[2]),M)=PolyMod(Poly(-P2[2]),M)
```

```
Then Return "Infinity";
EndIf;
If P1=P2 Then
S := PolyMod(Poly(3 * P1[1]^2 + C[1])*PolyInvM(Poly(2 *
   P1[2]),b^2+b+1,M),M);
Else
If Poly(P1[1])=Poly(P2[1]) AND Poly(P1[2])=PolyMod(Poly(-P2[2]),M)
Then Return "Infinity"; EndIf;
S:=PolyMod(Poly(P2[2] - P1[2])*PolyInvM(Poly(P2[1] -
   P1[1]),b^2+b+1,M),M);
EndIf;
Y0:=P1[2]-S*P1[1];
XR:=PolyMod(NR(S^2-P1[1]-P2[1],[b^2+b+1]),M);
YR:=PolyMod(NR(-S*XR-Y0,[b^2+b+1]),M);
Return [XR,YR];
EndDefine;
```

ModNP calculates the scalar product of point P on curve C mod M. It require the input N,P,C,M where N is integer P is the point. It calculate N times P.

```
Define ModNP(N,P,C,M)
S:=P;
For I:=1 To N-1 Do
S:=ModEcAdd(S,P,C,M);
EndFor;
Return S;
EndDefine;
```

ModMillerFtn find the miller function $f_{A,B}$ define in section 3.2.2. In this code A=P1 and B=P2 points should be given as the list of integers x and y coordinates of the points.

```
Define ModMillerFtn(P1,P2,C,M);
If P1="Infinity"  Then Return PolyMod(x - Poly(P2[1]),M); EndIf;
If P2="Infinity"  Then Return PolyMod(x - Poly(P1[1]),M); EndIf;
If PolyMod(NR(Poly(P1[2]^2),[b^2+b+1]),M)<>PolyMod(NR(Poly(P1[1]^3+
C[1]*P1[1]+C[2]),[b^2+b+1]),M)
Then Error("First Point is not on the Elliptic Curve");
EndIf;
If PolyMod(NR(Poly(P2[2]^2),[b^2+b+1]),M)<>PolyMod(NR(Poly(P2[1]^3+
C[1]*P2[1]+C[2]),[b^2+b+1]),M)
Then Error("Second Point is not on the Elliptic Curve");
EndIf;

If P1[1]=P2[1] AND P1[2]=PolyMod(Poly(-P2[2]),M)
Then Return PolyMod(x - Poly(P1[1]),M);
EndIf;
If P1=P2  Then
S := PolyMod(Poly(3 * P1[1]^2 + C[1]) * PolyInvM(Poly(2 *
   P1[2]),b^2+b+1,M),M);
Return (NR((y - P1[2] - S*(x - P1[1])),[b^2+b+1])/NR((x + P1[1] + P2[1]
   - PolyMod(Poly(S^2),M)),[b^2+b+1]));
EndIf;
If P1<>P2 Then S:=PolyMod(Poly(P2[2] - P1[2]) * PolyInvM(Poly(P2[1] -
   P1[1]),b^2+b+1,M),M);
Return (NR((y - P1[2] - S*(x - P1[1])),[b^2+b+1])/NR((x + P1[1] + P2[1]
   - PolyMod(Poly(S^2),M)),[b^2+b+1]));
EndIf;
EndDefine;
```

ModMillerCal calculate the miller function by using Miller Algorithm 3.2.3. It requires the input N, P, C, M where P is point on Curve C mod M and N is order of P.

```
Define ModMillerCal(N,P,U,C,M)
F:=1;
T:=P;
K:=Reversed(Dec2Bin(N));
```

```
J:=Len(K);
For I:=J-1 To 1 Step -1  Do
F:=F*F * ModMillerFtn(T,T,C,M);
F:=PolyMod(F,M);
PrintLn("F = ",F);
F:=PolyMod(Subst(F, [[x,U[1]],[y,U[2]]]),M);
T:=ModNP(2,T,C,M);
PrintLn("I = ",I, " and N = ",N);
If K[I]=1 Then
F:=F*ModMillerFtn(T,P,C,M);
F:=PolyMod(F,M);
PrintLn("F = ",F);
F:=PolyMod(Subst(F, [[x,U[1]],[y,U[2]]]),M);
T:= ModEcAdd(T,P,C,M);
EndIf;
EndFor;
Return PolyMod(F,M);
EndDefine;
```

Finally,**ModWeil** uses the **ModMillerCal** function and evaluate the Modified Weil pairing between A and B se section 3.3. In this code, we take $A = P$, $B = Q$ and $C = S$ points on Curve C mod M and N is order of P and Q.

```
Define ModWeil(N,P,Q,S1,C,M);
NS:=[S1[1],PolyMod(Poly(-S1[2]),M)];
If P="Infinity"  Then Return PrintLn("Calculated  Pairing Between
    ",P,"and ",Q,"= ",1); EndIf;
If Q="Infinity"  Then Return PrintLn("Calculated  Pairing Between
    ",P,"and ",Q,"= ",1); EndIf;
QS:=ModEcAdd(Q,S1,C,M);
A:=ModMillerCal(N,P,QS,C,M);
B:=ModMillerCal(N,P,S1,C,M);
PS:=ModEcAdd(P,NS,C,M);
C1:=ModMillerCal(N,Q,PS,C,M);
```

```
D:=ModMillerCal(N,Q,NS,C,M);
WP1:=PolyMod(A*D,M);
WP2:=PolyMod(B*C1,M);
WP:=PolyMod(WP1/WP2,M);
Return PrintLn("Calculated Weil Pairing Between ",P,"and ",Q,"=
   ",PolyMod(NR(WP.Num*PolyInvM(WP.Den,b^2+b+1,M),[b^2+b+1]),M));
EndDefine;
```

Bibliography

[1] “Apcoca team,ApCoCoA”. ApCoCoA :Applied computation in commutative algebra. Available at http://www.apcocoa.org.

[2] R. M. Abobeah, M. M. Ezz, & H. M. Harb, “Public-key cryptography techniques evaluation”. *International Journal of Computer Networks and Applications*, 2, 2015.

[3] A. E. Aftuck, “The Weil Pairing on Elliptic Curves and its Cryptographic Applications”. *UNF Theses and Dissertations Paper 139*, 2011. http://digitalcommons.unf.edu/etd/139/.

[4] G. Agarwal & S. Singh, “A comparison between public key authority and certification authority for distribution of public key”. *International Journal of Computer Science and Information Technologies*, 1(5):332–336, 2010.

[5] J. Anzai, N. Matsuzaki, & T. Matsumoto, “A quick group key distribution scheme with entity revocation”. *Advances in Cryptology-ASIACRYPT99*, pages 333–347, 1999.

[6] J. Baek, R. Safavi-Naini, & W. Susilo, “Efficient multi-receiver identity-based encryption and its Application to Broadcast Encryption”. In *Public Key Cryptography*, volume 3386, pages 380–397. Springer, 2005.

[7] W. G. Barker, “Introduction to the analysis of the Data Encryption Standard (DES)”. Aegean Park Press, 1991.

[8] J. Baylis, “Contemporary abstract algebra , by joseph a. gallian.”. *The Mathematical Gazette*, 75(473):374–375, 1991.

[9] S. Berkovits, "How to broadcast a secret". In *Advances in CryptologyEUROCRYPT91*, pages 535–541. Springer, 1991.

[10] D. Boneh & M. Franklin, "Identity-based encryption from the Weil pairing". In *Advances in CryptologyCRYPTO 2001*, pages 213–229. Springer, 2001.

[11] D. Boneh & M. Hamburg, "Generalized identity based and broadcast encryption schemes". *Advances in Cryptology-ASIACRYPT 2008*, pages 455–470, 2008.

[12] R. Canetti, S. Halevi, & J. Katz, "Chosen-ciphertext security from identity-based encryption". In *Advances in Cryptology-eurocrypt 2004*, pages 207–222. Springer, 2004.

[13] J. Cassels et al., "Joseph H. Silverman, The arithmetic of elliptic curves". *Bulletin (New Series) of the American Mathematical Society*, 17(1):148–149, 1987.

[14] S. Chatterjee & P. Sarkar, "Identity-based encryption". Springer Science & Business Media, 2011.

[15] C. Cipher & M. Cipher, "Introduction to cryptography". *EEC*, 484:584, 2004.

[16] J. D. da Silva, "Multilinear algebra: Recent applications". *Linear algebra and its applications*, 241:211–223, 1996.

[17] C. Delerablée, "Identity-based broadcast encryption with constant size ciphertexts and private keys". In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 200–215. Springer, 2007.

[18] Y. Desmedt & J.-J. Quisquater, "Public-key systems based on the difficulty of tampering (Is there a difference between DES and RSA?)". In *Advances in CryptologyCRYPTO86*, pages 111–117. Springer, 1987.

[19] W. Diffie & M. Hellman, "New directions in cryptography". *IEEE transactions on Information Theory*, 22(6):644–654, 1976.

[20] Y. Dodis & N. Fazio, “Public key broadcast encryption for stateless receivers”. In *Digital Rights Management Workshop*, volume 2696, pages 61–80. Springer, 2002.

[21] Y. Dodis & N. Fazio, “Public key trace and revoke scheme secure against adaptive chosen ciphertext attack”. In *International Workshop on Public Key Cryptography*, pages 100–115. Springer, 2003.

[22] A. Fiat & M. Naor, “Broadcast encryption”. In *Annual International Cryptology Conference*, pages 480–491. Springer, 1993.

[23] G. Frey & H.-G. Rück, “A remark concerning -divisibility and the discrete logarithm in the divisor class group of curves”. *Mathematics of computation*, 62(206):865–874, 1994.

[24] C. Gentry & B. Waters, “Adaptive security in broadcast encryption systems (with short ciphertexts)”. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 171–188. Springer, 2009.

[25] D. Halevy & A. Shamir, “The LSD broadcast encryption scheme”. *Advances in CryptologyCRYPTO 2002*, pages 145–161, 2002.

[26] C. Heinrich, “Pretty good privacy (PGP)”. In *Encyclopedia of Cryptography and Security*, pages 955–958. Springer, 2011.

[27] K. Hoffman & R. Kunze, “Linear algebra”. *Englewood Cliffs, New Jersey*, 1971.

[28] M. S. Iqbal, S. Singh, & A. Jaiswal, “Symmetric key cryptography: Technological developments in the field”. *International Journal of Computer Applications*, 117(15), 2015.

[29] A. Joux & K. Nguyen, “Separating decision Diffie–Hellman from computational Diffie–Hellman in cryptographic groups”. *Journal of cryptology*, 16(4): 239–247, 2003.

[30] L. M. Kohnfelder, “Towards a practical public-key cryptosystem”. PhD thesis, Massachusetts Institute of Technology, 1978.

[31] A. Lewko & B. Waters, "New proof methods for attribute-based encryption: Achieving full security through selective techniques". In *Advances in Cryptology–CRYPTO 2012*, pages 180–198. Springer, 2012.

[32] A. B. Lewko & B. Waters, "New techniques for Dual System Encryption and fully secure HIBE with short ciphertexts". In *TCC*, volume 5978, pages 455–479. Springer, 2010.

[33] S. Lichtenbaum, "Duality theorems for curves overp-adic fields". *Inventiones mathematicae*, 7(2):120–136, 1969.

[34] M. Maas, "Pairing-based cryptography". *Master's thesis, Technische Universiteit Eindhoven*, 2004.

[35] U. Maurer & Y. Yacobi, "Non-interactive public-key cryptography". In *Advances in CryptologyEUROCRYPT91*, pages 498–507. Springer, 1991.

[36] V. Miller et al., "Short programs for functions on curves". *Unpublished manuscript*, 97:101–102, 1986.

[37] Y. Ming & Y. Wang, "Identity Based Broadcast Encryption with group of prime order". *International Arab Journal of Information Technology (IAJIT)*, 13(5), 2016.

[38] M. A. Musa, E. F. Schaefer, & S. Wedig, "A simplified aes algorithm and its linear and differential cryptanalyses". *Cryptologia*, 27(2):148–177, 2003.

[39] D. Naor, M. Naor, & J. Lotspiech, "Revocation and tracing schemes for stateless receivers". In *Advances in CryptologyCRYPTO 2001*, pages 41–62. Springer, 2001.

[40] T. Okamoto & K. Takashima, "Hierarchical Predicate Encryption for inner-products". In *Asiacrypt*, volume 5912, pages 214–231. Springer, 2009.

[41] Y. Ren & D. Gu, "Fully CCA2 secure identity based broadcast encryption without random oracles". *Information Processing Letters*, 109(11):527–533, 2009.

[42] R. L. Rivest, A. Shamir, & L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems”. *Communications of the ACM*, 21 (2):120–126, 1978.

[43] J. J. Rotman, “A first course in abstract algebra”. Pearson College Division, 2000.

[44] R. Sakai & J. Furukawa, “Identity-based broadcast encryption”. *IACR Cryptology ePrint Archive*, 2007:217, 2007.

[45] A. Shamir et al., “Identity-based cryptosystems and signature schemes”. In *Crypto*, volume 84, pages 47–53. Springer, 1984.

[46] J. H. Silverman, “The arithmetic of elliptic curves”, volume 106. Springer Science & Business Media, 2009.

[47] R. Singh & S. Kumar, “Elgamals algorithm in cryptography”. *International Journal of Scientific and Engineering Research*, 3(12):1–4, 2012.

[48] W. Stallings, “Cryptography and network security: principles and practices”. Pearson Education India, 2006.

[49] H. Tanaka, “A realization scheme for the identity-based cryptosystem”. In *Advances in CryptologyCRYPTO87*, pages 340–349. Springer, 2006.

[50] S. Tsujii & T. Itoh, “An ID-based cryptosystem based on the discrete logarithm problem”. *IEEE Journal on Selected Areas in Communications*, 7(4): 467–473, 1989.

[51] L. C. Washington, “Elliptic curves: number theory and cryptography”. CRC press, 2008.

[52] B. Waters et al., “Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions”. In *Crypto*, volume 5677, pages 619–636. Springer, 2009.

[53] L. Zhang, Y. Hu, & Q. Wu, “Adaptively secure identity-based broadcast encryption with constant size private keys and ciphertexts from the subgroups”. *Mathematical and computer Modelling*, 55(1):12–18, 2012.