CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

# Noncommutative Cryptography using Extra Special Group and Galois Field

by

Khurram Ali

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

in the

Faculty of Computing

Department of Mathematics

2019

*To my parents, brother and sisters for their support and love.*

## CERTIFICATE OF APPROVAL

# Noncommutative Cryptography using Extra Special Group and Galois Field

by

Khurram Ali

(MMT171012)

## THESIS EXAMINING COMMITTEE

| S. No. | Examiner | Name | Organization |
|---|---|---|---|
| (a) | External Examiner | Dr. Waqas Mehmood | QAU, Islamabad |
| (b) | Internal Examiner | Dr. Dur-e-Shewar | CUST, Islamabad |
| (c) | Supervisor | Dr. Rashid Ali | CUST, Islamabad |

Thesis Supervisor
Dr. Rashid Ali
September, 2019

Dr. Muhammad Sagheer
Head
Dept. of Mathematics
September, 2019

Dr. Muhammad Abdul Qadir
Dean
Faculty of Computing
September, 2019

# *Author's Declaration*

I, **Khurram Ali** hereby state that my M. Phil thesis titled "**Noncommutative Cryptography using Extra Special Group and Galois Field**" 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 M. Phil Degree.

**(Khurram Ali)**
Registration No: MMT171012

# *Plagiarism Undertaking*

I solemnly declare that research work presented in this thesis titled **Noncommutative Cryptography using Extra Special Group and Galois Field** 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 M. Phil Degree, the University reserves the right to withdraw/revoke my M. Phil 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.

**(Khurram Ali)**
Registration No: MMT171012

# *Acknowledgements*

First of all, I wold like to thanks **Allah Almighty** for is countless blessing 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 supervisor **Dr. Rashid Ali** for his motivation. His unfailing patience and encouragement kept me in good stead. i will never be able to forget his key contribution to one of the most fruitful endeavor of my life. I have appreciated the guidance for my supervisor and feeling proud to be a student of such great teacher.

Also,many thanks are due to all teachers of CUST Islamabad Dr.Muhammad Sagheer, Dr.Abdul Rehman Kashif, Dr.Shafqat Hussain, Dr. M. Afzal, Dr. Samina Rashid, Dr. Dur-e-Shewar Sagheer and Dr. Rashid Ali for their appreciation and support.

I am grateful to the management staff of Capital university of science and technology, Islamabad for providing a friendly environment for studies.

I am greatful to My father **Ghazanfar Ali** and My mother **Rukhsana Ali** for their prayers, love and motivation. I would like to thanks my brother **Nayyer Ali** and sister **Sehrish Ali, Savera Ali** and my sister in law **Rummana Naz** for their support in completing my degree program. They supported and encouraged me throughout my life. I would like to thanks my all family members for their continuous support and patience during my research work.

I also feel honored to have such supporting friends. I would like to specially thank my friend **Ali Asghar**, Malik Ayoub and M.Haris for providing the strength to get focused to ward my main objective.

Finally, I am obliged to all people who have shared their knowledge and supported me all along.

**(Khurram Ali )**

Registration No: MMT171012

# *Abstract*

Non-commutative cryptography (NCC) is truly fascinating with the great hope of advancing performance and security for high end applications. It provides a high level of safety measure. We have modified the scheme of Kumar and Saini using Galois Field ($GF(p^n)$). We have proposed to use of matrices from Galois field $GF(p^n)$. In order to improve the security of the scheme, we have used conjugacy search problem together with symmetrical decomposition problem. The working principal is based on the random polynomial chosen by the communicating parties to secure key exchange, encryption and decryption. The projected approach is exclusively based on the typical sparse matrices, and an analysis report in presenting fulfilling the central cryptographic requirements.

# Contents

# List of Figures

# List of Tables

# Abbreviations

| | |
|---|---|
| **AES** | Advance Encryption Standard |
| **DES** | Data Encryption Standard |
| **DLP** | Discrete Logarithm Problem |
| **ECC** | Elliptic Curve Cryptography |
| **GF** | Galois Field |
| **IFP** | Integer Fctorization Problem |
| **RSA** | Rivest Shamir Adleman |
| **2DES** | Double Data Encryption Standard |
| **3DES** | Triple Data Encryption Standard |

# Symbols

| | |
|---|---|
| $M$ | Plaintext or Message |
| $C$ | Ciphertext |
| $K$ | Key |
| $E$ | Encryption Algorithm |
| $D$ | Decryption Algorithm |
| $R$ | Ring |
| $M_R$ | Matrix Ring |
| $\mathbb{N}$ | Natural Numbers |
| $\mathbb{Z}$ | Set of Integers |
| $\mathbb{R}$ | Real Numbers |
| $\mathbf{H}$ | Hash Function |
| $\mathbb{G}$ | Group |

# Chapter 1

# Introduction

Cryptography is a mean of concealing or hiding true information in the form that are incomprehensible to others. The main purpose of cryptography is to maintain confidentiality of information. Within its history, the cryptographic schemes have developed from simple form of conversion into complex methods that we have today. Simple conversion includes rearranging of letters, or replacing or shifting letters. Some notable personalities in the history who made use of these forms of cryptography includes J.Ceaser [1] who used concealing of three letters, to communicate with his generals, and T. Jefferson who developed a wheel cipher that was used in the U.S. Navy during the World War II [2]. On the other hand, complex method are the result of modern technology such as data encryption, digital signature, authentication of senders/receivers, public key cryptography, and secure computation among others.

In the world War I and II German's and the Japanese made use of cryptography in the battle field. The German's Enigma and the Japanese Purple machine are two of the famous machines used in the War. In 1976 Diffie and Hellman [3] introduced public key cryptography. Later on different Public Key Cryptographic (PKC) schemes have been suggested and various others have been broken. The one-way trapdoor functions are central to the idea of PKC. These days most thriving PKC schemes are based on the known difficulty of specific problems, specially the large finite commutative rings. For instance, the difficulty of solving integer factorization problem(IFP) defined over the ring $\mathbb{Z}_{\ltimes}$, which forms a basis of RSA cryptosystem. The detailed multi-faceted RSA cryptosysrem [4], which can adequately oppose low exponent attacks, is also explained by the commutative ring $\mathbb{Z}_{\ltimes}$. Other examples of cryptographic algorithm based on commutative ring are ElGamal-like PKC schemes [5], DSS and McCurley scheme [6].

## 1.1 Non-Commutative Cryptography

“The cryptosystems dependent on the Integer Factorization Problem (IFP) [7, 8], the Discrete Logarithm Problem (DLP) [7] and the Elliptic Curve Discrete Logarithm Problem (ECDLP) are basically the main three sorts of down to earth open key cryptosystems being used. The security of these cryptosystems relies intensely on these three infeasible issues. Quantum computers are thought to be able to solve these problems thousands of times faster than classical computers. Scientists have been working on developing them for more than a decade. Quantum PCs will break the present most well known open key cryptographic frameworks, including RSA , DSA, and ECDSA [4]”. Therefore analyst move to utilize a non-commutative cryptography as secure choice.

A field of cryptology where the cryptographic traditional techniques and system based on noncommutative algebraic structures like semi groups, groups and rings is known as Noncommutative cryptography. Traditionally, braid groups [9], were used to develope cryptographic protocols in noncommutative algebraic structures. Other noncommutative structures Thompson groups [10], matrix groups [11], polycyclic groups [12] and Grig-orchuk groups were afterwards regarded important for cryptographic applications. Currently mostly used PKC like RSA cryptosystem, Diffie-Helman key exchange and Elliptic curve cryptography are based on commutative algebraic structure.

To solve many cryptographic complexities like key-exchange, encryption/decryption and authentication, noncommutative cryptographic protocols have been introduced.

## 1.2 Current Research

In this research, we focused on **"Non-commutative Cryptography Scheme Using Extra Special Group"** introduced by Kumar and Siani [13]. They proposed it with matrices from $\mathbb{Z}_{\mathbb{N}}$. Also, they used symmetrical decomposition problem in their scheme. We mainly focused on the modification of noncommutative cryptographic scheme using extra special group. For this purpose, we have used matrices from $GF(p^n)$. Also we have used conjugacy search problem in our scheme. Our scheme has become more secure as the attacker would have to solve symmetrical decomposition problem as well as congugacy search problem, to get access to secret key, which is computationally infeasible. Using Galois field, we constructed examples for the illustration of our modified scheme.

## 1.3 Thesis Layout

The composition of rest of the thesis is as follows:

1. In **Chapter 2**, we will explain the fundamental ideas and definition of cryptography. Then we discussed mathematical background, Galois Field and arithmetic in Galois field. Later on, we have discussed Fermat's theorem, hash functions and their properties.

2. In **Chapter 3**, we have presented the review of "Non-commutative Cryptography Scheme Using Extra Special Groups" by Saini and Kumar [13]. Also we have presented a cryptanalysis of the scheme. Furthermore, we have described the concepts on noncommutative cryptography scheme with the help of examples.

3. In **Chapter 4**, we have discussed the modified form of the key exchange on" Non-commutative Cryptography Scheme Using Extra Special Groups". In the modified scheme, we have used Heisenberg group of matrices over Galois Field. Also, we have used conjugacy search problem, together with

symmetrical decomposition problem, to improve security of the algorithm. The modified scheme is illustrated with examples and the last section is devoted to the security analysis.

# Chapter 2

# Preliminaries

In this chapter we will describe the fundamental ideas, mathematical background and definitions related to the thesis.

## 2.1 Cryptography

Cryptography is the art and science for transforming the secret messages into an unreadable format, called ciphertext. Only those who have a secret key can decipher the ciphertext into original message. Cryptography can also be used for user authentication. This is traditionally based on mathematical foundation. In cryptography we develop a secure cryptosystem. A system in which we convert data or message into secret codes using encryption algorithm and convert secret codes back into messsage using decryption algorithm is know as cryptosystem. There are five basic components in cryptosystem:

i Plaintext space $M$

ii Ciphertext space $C$

iii Encryption algorithm $E$

iv Decryption algorithm $D$

v Key $K$

Cryptography have the following types

- Symmetric Key Cryptography(secret key cryptography)
- Public Key Cryptography

### 2.1.1 Symmetric Key Cryptography

"A system in which related keys is used for both Encryption and Decryption is called symmetric key cryptography [14]. For example, Data Encryption Standard (DES)[15], Double Data Encryption Standard [16], Triple Data Encryption (3DES) [16] and Advance Encryption Standard (AES) [17]. A model of symmetric key cryptography is shown in the FIGURE 2.1"

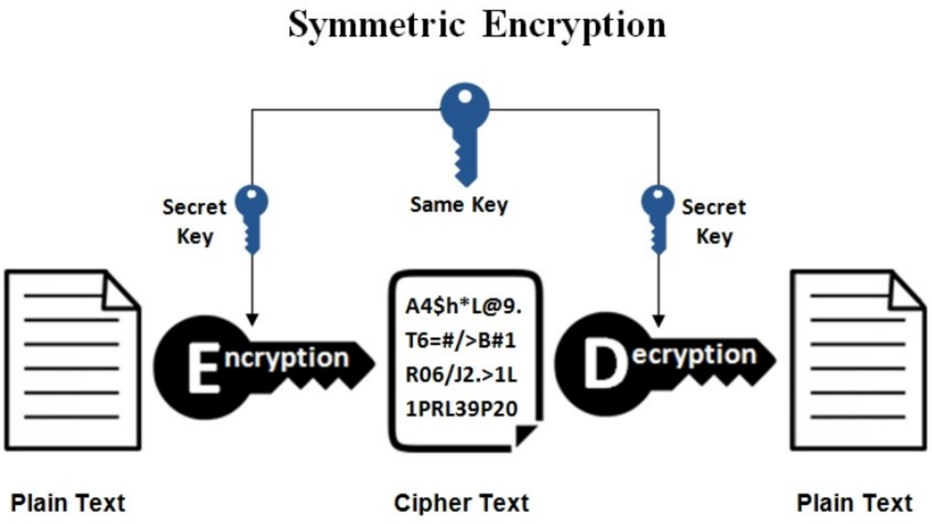

FIGURE 2.1: Symmetric Key

The main disadvantage of symmetric key cryptography is key sharing which means that the secret key is to be transmitted to each party involved in the communication. Electronic communication used for this purpose may not be a secure way of exchanging keys because anyone can access to the communication channels. The only protected ways of switching keys will be to exchange them privately but it could be a very difficult task.

### 2.1.2 Public Key Cryptography

"Public key cryptosystem is proposed by Diffie-Hellman in 1976 [3]. In public key cryptography [16], two keys are used for encryption and decryption, one is called public key which is known to everybody and the other is called secret key which is kept secret by user". The public key cryptography is shown in the FIGURE 2.2. Here sender encrypt original text using public key and encryption algorithm to obtain the cipher-text. The secret key and decryption algorithm are used by the receiver end to obtain orignal text.

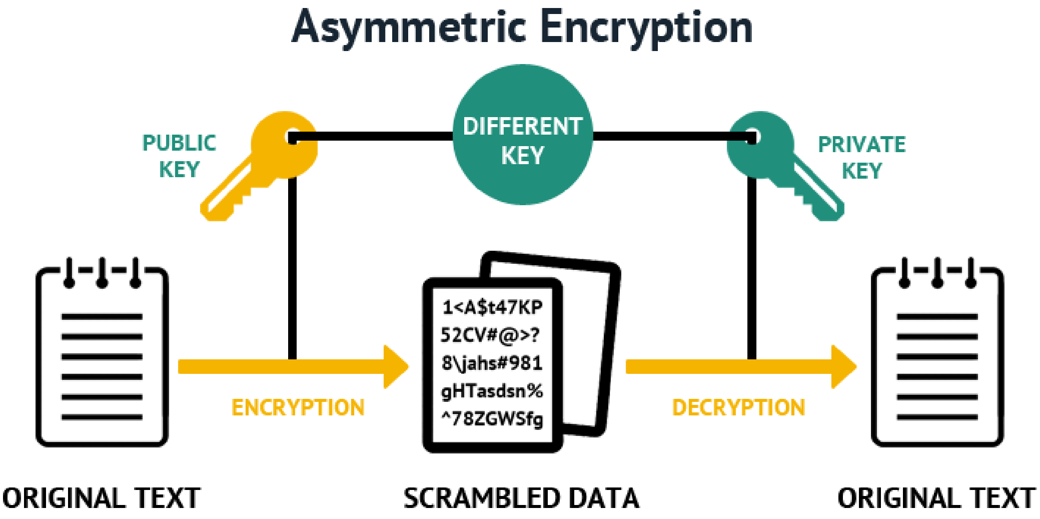

FIGURE 2.2: Asymmetric Key

RSA cryptosystem [4] and ElGamal cryptosystem [5] are examples of asymmetric key cryptography. Diffie and Hellman version of the cryptosystem based on trapdoor function (which is easy to calculate in one direction but hard to calculate in other direction). Diffie-Hellman protocol relies on some hard problems which will be discussed after the mathematical background.

## 2.2 Mathematical Background

In this section, we recall some tools in mathematics that are used in the thesis.

**Definition 2.2.1. (Group)**

"Let $\mathbb{G}$ be a non empty set and $*$ be a binary operation on $\mathbb{G}$. Then $(\mathbb{G}, *)$ is called a group [18] if it satisfies the following properties:

i. **Closure**: For all $a, b \in \mathbb{G}$, $a * b \in \mathbb{G}$,

ii. **Associative**: For all $a, b, c \in \mathbb{G}$ $(a * b) * c = a * (b * c)$,

iii. **Identity**: There is element $e \in \mathbb{G}$ such that $a * e = e * a = a$,

iv. **Inverse**: If $p \in \mathbb{G}$, then there is an element $p_1 \in \mathbb{G}$ such that $p * p_1 = p_1 * p = e$"

**Example 2.2.1.** The following are examples of group

i. Set of integers $\mathbb{Z}$ is a group with respect to addition of integers.

ii. Set of all invertible matrices with ordinary matrix multiplication form a group.

iii. Set of real numbers (only non zero elements) $\mathbb{R}$ form a group under multiplication.

**Definition 2.2.2.** (**Abelain Group**)

"A group $\mathbb{G}$ is called abelian group [18], if binary operation "*" is commutative that is

$$a * b = b * a \ \forall \ a, b \in \mathbb{G}\text{''}.$$

**Definition 2.2.3.** (**Ring**)

"A non-empty set together with two binary operations, one is addition (+) and other is multiplication $(\cdot)$, denoted by (R,+,$\cdot$) is said to be a ring [19] if it satisfies the following properties:

i. (R,+) is an **abelian group.**

ii. (R,$\cdot$) is a **monoid.**

iii. **Distributive property** of multiplication over addition holds.

That is $\forall\ p, m, n \in R$

$$p.(m+n) = p.m + p.n \text{ and}$$
$$(p+m).n = p.n + m.n\text{''}$$

**Example 2.2.2.** "Followings are the examples of ring

i. $\mathbb{Z}$ , $\mathbb{Q}$ , $\mathbb{R}$ and $\mathbb{C}$ all form ring under usual addition and multiplication.

ii. $M_n(\mathbb{R})$ set of all $n \times n$ matrices over the ring $\mathbb{R}$ is also a ring under addition and multiplication .

iii. If $p$ is a prime than the set $\mathbb{Z}_p$ of integer mod $p$ is a ring.

iv. Set of odd integer is not a ring because it does not satisfied closure property under multiplication."

**Definition 2.2.4. (Commutative Ring )**

"A ring is known as commutative ring [20], if commutative property of multiplication holds, that is $u \times v = v \times u$"

**Example 2.2.3.** The non-commutative ring $M_n(R)$ is the set of all $n \times n$ matrices over a ring $R$ is not commutative ring because matrix multiplication is not commutative.

**Definition 2.2.5. (Semiring)**

"A set $S$ [21], together with two binary operation "+" and "$\cdot$" is called the semiring if it satisfies the following conditions:

i. $S$ is semi group under "+"

ii. $S$ is semi group under "$\cdot$"

iii. Multiplication is distributive over addition in either side. That is, for all $u, v, w \in S$ we have

$$u \cdot (v + w) = (u \cdot v) + (u \cdot w)$$
$$(u + v) \cdot \text{w} = (u \cdot \text{w}) + (v \cdot \text{w})$$

**Example 2.2.4.** Following are the examples of semiring.

i Every ring is a semiring therefore set of integers $\mathbb{Z}$, rational number $\mathbb{Q}$, real number $\mathbb{R}$ and complex number $\mathbb{C}$ all are semirings.

ii Set of whole number $\mathbb{W}$ is a semiring.

iii Set of all non-negative integers, non-negative rational numbers and non-negative real numbers are examples of semiring.

iv The set of polynomial with natural numbers as coefficients, denoted by $\mathbb{N}[\mathbb{X}]$, forms a semi-ring. In fact,this is the commutative semiring on a single generator $X$."

**Definition 2.2.6. (Commutative Semiring)**

"A semiring $S$ is known as commutative semiring [22] if commutative property of multiplication holds i.e.,

$$u \cdot v = v \cdot u \qquad \forall\, u, v \in S."$$

**Definition 2.2.7. (Field)**

"A nonempty set $\mathbb{F}$ with two binary operation addition $(+)$ and $(\cdot)$ is called a field [21], if it satisfies the following properties:

i. $(\mathbb{F},+)$ is an abelian group.

ii. $(\mathbb{F},\cdot)$ is an abelian group.

iii. Distributivity of addition over multiplication. "

**Example 2.2.5.** Examples of field are

i. Set of real and complex numbers are fields under usual addition and multiplication.

ii. Set of integers $\mathbb{Z}$ is not a field as there are no multiplicative inverses in $\mathbb{Z}$".

**Definition 2.2.8. (Finite Field)**

Finite field is a field that contains finite numbers of elements.

**Example 2.2.6.** Following are the examples of finite field

i. $\mathbb{Z}$ under mod $p$ where $p$ is prime is a field.

ii. Galois fields are finite field. For example $GF(2)$, $GF(2^3)$ and $GF(3)$.

-Galois Field "A finite field whose order is the form of $p^n$, where $n$ is any integer and $p$ is prime number is called Galois Field denoted by $GF(p^n)$ [23]. In Galois field, elements are defined as

$GF(p^n) = (0, 1, 2, ...., p-1) \cup (p, p+1, p+2, ..., p+p-1) \cup (p^2, p^2+1, p^2+2, ..., p^2+p-1) \cup .... \cup (p^{n-1}, p^{n-1}+1, p^{n-1}+2, ..., p^{n-1}+p-1)$.

The order of Galois field is given by $p^n$ while $p$ is characteristics of field and the degree of the polynomials in $GF(p^n)$ is less than $n$, while coefficients is at most $p-1$."

**Example 2.2.7.** $GF(3^2)$=$(0, 1, 2, x, x+1, x+2, 2x, 2x+1, 2x+2)$ consist $3^2 = 9$ elements where each of the polynomials have degree less than 2 and coefficients are less than 3.

**Example 2.2.8.** Finite field $\mathbb{F}_2$ i.e., $\{0,1\}$ with addition and multiplication is defined in TABLE 2.1 and TABLE 2.2 below.

| + | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 0 |

TABLE 2.1: Addition

| . | 0 | 1 |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |

TABLE 2.2: Multiplication

## 2.3 Computation in Galois Field

In this section, we will explain algebraic operation in Galois field. In Galois field algebraic expression needs some additional steps. Disparate functioning in the Euclidean space, algebraic operations (+,-,×) in Galois Field need some additional steps.

### 2.3.1 Addition and Subtraction in Galois Field

In Galois field, the operation of addition is quite simple. If $f_1(x)$, $g_1(x)$ are any two polynomials in $GF(p^n)$ and $h_1(x) = f_1(x) + g_1(x)$ with the coefficients of $f_1(x)$, $g_1(x)$ and $h_1(x)$ are $A = a_{n-1}, a_{n-2}, ...., a_1 a_0, B = b_{n-1}, b_{n-2}, ...b_1, b_0$, and $C = c_{n-1}, c_{n-2}, ...c_1, c_0$ respectively. Let $a_k$, $b_k$ and $c_k$ are the coefficients of $f_1(x)$, $g_1(x)$ and $h_1(x)$ respectively then

$$c_k = a_k + b_k \bmod p \text{ for } k = 0, 1, 2, 3, ....n - 1$$

Likewise if $h_1(x) = f_1(x) - g_1(x)$ then $c_k = a_k - b_k \bmod p$ where $k \in \{0, 1, 2, 3......n-1\}$.

Note that in Galois field $GF(2^n)$ addition can be performed using XOR operation. The element of Galois field can be represent by a unique $n$-bit pattern. We can transform polynomials of Galois field in binary number system from which we can convert it into any number system.

**Example 2.3.1.** Conversion of polynomials into different number systems
Let $x^7 + x^5 + x^4 + x + 1$ is the polynomial in $GF(2^8)$.
The binary representation

$$x^7 + x^5 + x^4 + x + 1 = (10110011)_2$$

In hexadecimal representation

$$x^7 + x^5 + x^4 + x + 1 = (B3)_{16}$$

In decimal representation

$$x^7 + x^5 + x^4 + x + 1 = (10110011)_2 = 179$$

**Example 2.3.2.** Suppose we are working in $GF(2^4)$, then we have compute $f(x)+g(x)$ if $f(x) = x^3 + x^2 + x + 1$, $g(x) = x^2 + 1$ under the mod $m(x)$ where $m(x) = x^4 + x^3 + x + 1$ then

$$f(x) + g(x) = (x^3 + x^2 + x + 1) + (x^2 + 1)$$

$$f(x) + g(x) = (x^3 + x) \mod \quad (x^4 + x^3 + x + 1)$$

Alternatively, from binary number system

$$f(x) = x^3 + x^2 + x + 1 = (1111)_2 \text{ and } g(x) = x^2 + 1 = (0101)_2$$

$$f(x) + g(x) = 1111 \oplus 0101$$

$$= 1010$$

$$= x^3 + x$$

### 2.3.2 Multiplication and Multiplicative Inverse

In Galois Field, multiplication involves more hard work. Suppose $f_1(x)$, $g_1(x)$ be any two polynomials in $GF(p^n)$ and suppose $m_1(x)$ be irreducible polynomial. The degree of product of $f_1(x)$ and $g_1(x)$ should be less than $n$ in $GF(p^n)$. If $h_1(x)$ represent the product of $f_1(x)$, $g_1(x)$ then

$$h_1(x) = f_1(x) \cdot g_1(x) \mod p$$

Suppose $a_1(x)$ represent the multiplicative inverse of $f_1(x)$ then

$$f_1(x) \cdot a_1(x) = 1 \mod p$$

Note that in evaluating the multiplication of any two polynomials and their inverses need both reducing polynomial $m_1(x)$ and coefficients in modulo $p$. The most feasible method to calculate the multiplicative inverse of polynomials is Extended Euclidean Algorithm.

**Example 2.3.3.** Let $f_1(x) = x^2 + 1$ and $g_1(x) = x^2 + x + 1$ with irreducible polynomial $m_1(x) = x^3 + x^2 + 1$ in $GF(2^3)$ evaluate $f(x).g(x)$

$$\begin{aligned} f_1(x).g_1(x) &= (x^2+1)(x^2+x+1) \\ &= (x^2+1)(x^2+x+1) \\ &= x^4+x^3+x^2+x^2+x+1 \\ &= x^4+x^3+x+1 \\ &= 1 \mod (x^3+x+1) \end{aligned}$$

### 2.3.3 Fermat's Little Theorem

"If $p$ is any prime and $a$ is any integer such that $p$ does not divide $a$, then $a^{p-1} \equiv 1 \bmod (p)$ [24]."

**Example 2.3.4.** Following are applications of Fermat's little theorem
(i): $a^{37}\equiv a \pmod 5$, (ii): $a^{37} \equiv a \pmod 7$ and (iii): $a^{37} \equiv a \pmod 3$
Solution: (i)
Since Fermat's little theorem $a^{p-1} \equiv 1 \bmod (p)$ therefore

$$a^{5-1} \equiv 1 \bmod (5)$$

$$a^4 \equiv 1 \bmod (5)$$

$$a^{37} \equiv (a^4)^9 \cdot a \equiv 1^9 \cdot a \equiv a \pmod 5$$

(ii)

$$a^{7-1} \equiv 1 \bmod (7)$$

$$a^{37} \equiv (a^6)^6 \cdot a \equiv 1^6 \cdot a \equiv a \bmod (7)$$

(iii)

$$a^{3-1} \equiv 1 \bmod (3)$$

$$a^{37} \equiv (a^2)^{18} \cdot a \equiv 1^{18} \cdot a \equiv a \bmod (3)$$

## 2.4 Polynomial

In this section we will explain the different types of polynomials, arithmetic of the polynomials and multiplicative inverses of the polynomials. There are three different types of polynomials such as:

i. Usual polynomials,

ii. Polynomials based on modulo prime,

iii. Polynomials based on modulo prime defined on other polynomials which have some power $n$, where $n$ is an integer.

### 2.4.1 Usual Polynomial

An expression of the form

$$f(y) = b_m y^m + b_{m-1} y^{m-1} + ..... + b_1 y + b_0$$

for $b_m \neq 0$ is called usual polynomial with $\forall\ b_i \in \mathbb{R}$ and $m_i$ are non negative integers.

### 2.4.2 Polynomial Based on Modulo Prime

An expression of the form:

$$f(x) = f_0 + f_1 x + f_2 x^2 +, \cdot \cdot \cdot, f_m x^m$$

where the coefficient are taken from a finite field $\mathbb{F}$ is called polynomial over $\mathbb{F}$. With polynomial over $\mathbb{F}$, the coefficients should be reduced under modulo prime $p$.

### 2.4.3 Polynomials Based on Modulo Prime Defined on Other Polynomials (which have some power $n$)

A polynomial with coefficients reduced by modulo prime $p$ and degree of the polynomial is decreased by modulo an irreducible polynomial of degree $n$, where $n$ is an integer.

## 2.5 Polynomial Arithmetic

In this section we will explain the ordinary polynomial arithmetic, polynomial arithmetic with coefficient mod $n$ and modular polynomial arithmetic.

### 2.5.1 Ordinary Polynomial Arithmetic

"In ordinary polynomial arithmetic we add or subtract corresponding coefficients of polynomial then multiply all term by each other" e.g, let $f(y) = y^4 + y^2 + 1$ and $g(y) = y^3 - y + 2$

$$\begin{aligned} f(y) + g(y) &= y^4 + y^2 + 1 + y^3 - y + 2 \\ &= y^4 + y^3 + y^2 - y + 3, \\ f(y) - g(y) &= y^4 + y^2 + 1 - y^3 + y + 2 \\ &= y^4 - y^3 + y^2 - 1 \end{aligned}$$

and

$$\begin{aligned} f(y) \cdot g(y) &= (y^4 + y^2 + 1)(y^3 - y + 2) \\ &= y^7 + 2y^5 - 2y^4 + 2y^3 - 2y^2 + y - 2 \end{aligned}$$

### 2.5.2 Polynomial Arithmetic with Coefficient Mod $n$

i. Reduce each coefficient modulo any integer.

ii. Polynomials form a ring if coefficient are in $GF(p)$

**Example 2.5.1.** Following are examples of addition and multiplication of polynomials mod $p$.
Let modulo prime $p = 2$ i.e., all coefficient are in $GF(2)$.
Suppose $f(y) = y^3 + y^2 + y + 1$ and $g(y) = y^3 + y$

$$f(y) + g(y) = y^3 + y^2 + y + 1 + y^3 + y$$

$$= 2y^3 + y^2 + 2y + 1$$

$$= y^2 + 1$$

$$f(y) \times g(y) = (y^3 + y^2 + y + 1)(y^3 + y)$$

$$= y^6 + y^5 + 2y^4 + y^3 + y^2 + y$$

$$= y^6 + y^5 + y^3 + y^2 + y$$

### 2.5.3 Modular Polynomial Arithmetic

"The polynomial [25] $r(x)$ is called the remainder of $f(x)$ modulo $g(x)$. For polynomials $a(x)$, $b(x)$ and $g(x)$ which are over the same field, we say $a(x)$ is congruent to $b(x)$ modulo $g(x)$ written $a(x) \equiv b(x)$ mod $g(x)$, if $m(x)$ divides $a(x) - b(x)$.

**Example 2.5.2. arithmetic in $GF(3^2)$ mod $(a'^2 + 1\ )$:**

| $+$ | $0$ | $1$ | $2$ | $a'$ | $a'+1$ | $a'+2$ | $2a'$ | $2a'+1$ | $2a'+2$ |
|---|---|---|---|---|---|---|---|---|---|
| $0$ | $0$ | $1$ | $0$ | $0$ | $0$ | $0$ | $0$ | $0$ | $0$ |
| $1$ | $0$ | $1$ | $2$ | $a'$ | $a'+1$ | $a'+2$ | $2a'$ | $2a'+1$ | $2a'+2$ |
| $2$ | $0$ | $2$ | $1$ | $2a'$ | $2a'+2$ | $2a'+1$ | $a'$ | $a'+2$ | $a'+1$ |
| $a'$ | $0$ | $a'$ | $2a'$ | $2$ | $a'+2$ | $2a'+2$ | $1$ | $a'+1$ | $2a'+1$ |
| $a'+1$ | $0$ | $a'+1$ | $2a'+2$ | $a'+2$ | $2a'$ | $1$ | $2a'+1$ | $2$ | $a'$ |
| $a'+2$ | $0$ | $a'+2$ | $2a'+1$ | $2a'+2$ | $1$ | $a'$ | $a'+2$ | $2a'$ | $2$ |
| $2a'$ | $0$ | $2a'$ | $a'$ | $1$ | $2a'+1$ | $a'+1$ | $2$ | $2a'+2$ | $a'+2$ |
| $2a'+1$ | $0$ | $2a'+1$ | $a'+2$ | $a'+1$ | $2$ | $2a'$ | $2a'+2$ | $a'$ | $1$ |
| $2a'+2$ | $0$ | $2a'+2$ | $a'+1$ | $2a'+1$ | $a'$ | $2$ | $a'+2$ | $1$ | $2a'$ |

TABLE 2.3: Addition in $GF(3^2)$

| $\cdot$ | $0$ | $1$ | $2$ | $a'$ | $a'+1$ | $a'+2$ | $2a'$ | $2a'+1$ | $2a'+2$ |
|---|---|---|---|---|---|---|---|---|---|
| $0$ | $0$ | $1$ | $2$ | $a'$ | $a'+1$ | $a'+2$ | $2a'$ | $2a'+1$ | $2a'+2$ |
| $1$ | $1$ | $2$ | $0$ | $a'+1$ | $a'+2$ | $a'$ | $2a'+1$ | $2a'+2$ | $2a'$ |
| $2$ | $2$ | $0$ | $1$ | $a'+2$ | $a'$ | $a'+1$ | $2a'+2$ | $2a'$ | $2a'+1$ |
| $a'$ | $a'$ | $a'+1$ | $a'+2$ | $2a'$ | $2a'+1$ | $2a'+2$ | $0$ | $1$ | $2$ |
| $a'+1$ | $a'+1$ | $a'+2$ | $a'$ | $2a'+1$ | $2a'+2$ | $2a'$ | $1$ | $2$ | $0$ |
| $a'+2$ | $a'+2$ | $a'+3$ | $a'+1$ | $2a'+2$ | $2a'$ | $2a'+1$ | $2$ | $0$ | $a'+1$ |
| $2a'$ | $2a'$ | $2a'+1$ | $2a'+2$ | $0$ | $1$ | $2$ | $a'$ | $a'+1$ | $a'+2$ |
| $2a'+1$ | $2a'+1$ | $2a'+2$ | $2a'$ | $1$ | $2$ | $0$ | $a'+1$ | $a'+2$ | $a'$ |
| $2a'+2$ | $2a'+2$ | $2a'$ | $2a'+1$ | $2$ | $0$ | $1$ | $a'+2$ | $a'$ | $a'+1$ |

TABLE 2.4: Multiplication in $GF(3^2)$

The above tables are the examples of addition and multiplication in Galois field $GF(3^2)$ ”

## 2.6 Modular Multiplicative Inverse

In this section we will explain how to find multiplicative inverses modulo some integer $n$.

**Definition 2.6.1.** Given any two integer $r$ and $s$, the problem is to find an integer $t$ such $r.t \equiv 1 \bmod s$ and $r^{-1} \equiv t \bmod s$, where $1 \leq t \leq s-1$.

The multiplicative inverse of $r$ mod $s$ are relatively prime that is, $gcd(r, m) = 1$.

**Algorithm 2.5.1 (Multiplicative inverse in finite field)**

To find the multiplicative inverse in $\mathbb{Z}_p$, we can implement Euclidean Algorithm [26] in the computer algebra system ApCoCoA [27].

Following is the method of finding the inverse of $r$ mod $s$.

**Input**: An integer $r$ and an irreducible integer $s$.

**Output**: $r^{-1}$ mod $s$

i. Initialize six integers $U_i$ and $V_i$ for i=1,2,3 as
$(V_1, V_2, V_3) = (1, 0, m)$
$(W_1, W_2, W_3) = (0, 1, r)$

ii. If $W_3$=0, return $V_3$=gcd$(r, s)$; no inverse of $r$ exist in mod $s$

iii. If $W_3$=1 then return $W_3$ = gcd $(r, s)$ and $W_2 = r^{-1}$ mod $s$

iv. Now divide $V_3$ by $W_3$ and find the quotient $Q$ when $V_3$ is divided by $W_3$

v. Set $(P_1, P_2, P_3) = ((V_1 - QW_1), (V_2 - QW_2), (V_3 - QW_3))$

vi. Set $(V_1, V_2, V_3) = (W_1, W_2, W_3)$

vii. Set $(W_1, W_2, W_3) = (P_1, P_2, P_3)$

viii. Go to step (ii)

**Definition 2.6.2. (Monic Polynomial)**:

"A monic polynomial [28], is a mathematical expression that consists of coefficients and a single variable, with the leading coefficient equal to one. The leading coefficient is found in the term that contains the variable with the highest degree or exponent.”

**Definition 2.6.3. (Unique Factorization)**:

“Every monic polynomial $f(x)$ is either irreducible or can be factorized into a product of monic polynomial factors. Further if a factor is not irreducible, it can be factored further. Since factor degrees are decreasing but bounded below by 1, we must eventually arrive at a product of monic irreducible(prime) polynomials [28]”.

**Example 2.6.1.** “Following are examples of the irreducible polynomials

i. $x^2-2$ is irreducible polynomial over $\mathbb{Q}$. It has no simpler factors with rational coefficients.

ii. $x^2+1$ is irreducible over $\mathbb{R}$. It has no factors with real coefficient.”

**Definition 2.6.4. (Discrete Logarithm Problem)**

Given $x, y \in Z_p$ such that

$x^n = y \bmod p$

then finding $n$ is known as discrete logarithm problem [7].

**Definition 2.6.5. (Integer Factorization Problem)**

Let $n$ be a given number, the problem of decomposition of $n$ to the product of prime $p_\alpha$ and $q_\alpha$ such that $n = p_\alpha q_\alpha$ is called integer factorization problem [7, 8].

**Definition 2.6.6. (Symmetrical Decomposition Problem)**

“Given $a, b \in \mathbb{G}$ and $m, n \in \mathbb{Z}$, find $x \in \mathbb{G}$ such that

$$b = x^m.a.x^n$$

then finding $x$ is known as symmetrical decomposition problem [29]”.

**Definition 2.6.7. (Conjugacy Search Problem)**

“Let G be a group and $x, y \in \mathbb{G}$, whether or not they represent conjugate element of $\mathbb{G}$. That is, the problem is to determine whether there exist an element $z$ of $\mathbb{G}$ such that $y = zxz^{-1}$ is known as Conjugacy Search Problem[30]”.

### 2.6.1 Hash Function

“A Hash function is any function, that maps data of random size into a fixed length hash value as shown in the figure 2.3. The hash value is representative of the orignal string of charecter, but is smaller than the orignal [31, 32]. Secure Hash Algorithm (SHA) is commonly used hash function. National institute of standard and technology (NIST) devalpoed SHA in 1993.

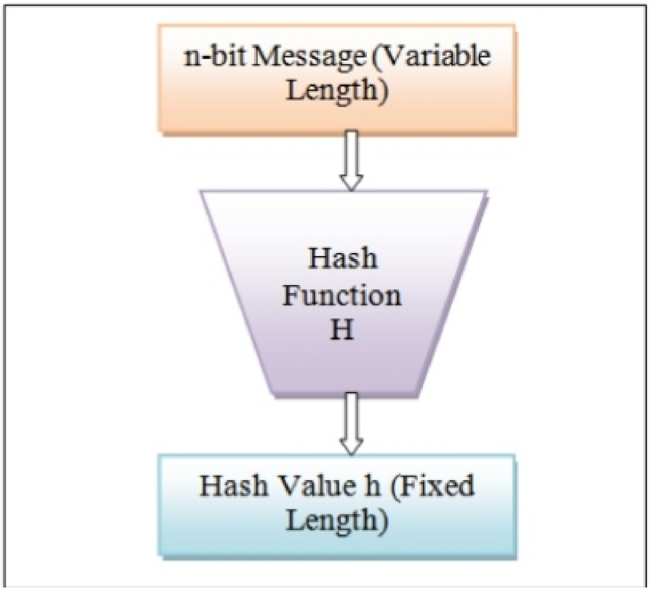

FIGURE 2.3: Hash function

Some known cryptographic hash function are (SHA-1 [33] , which produces a hash value of 160 bits), SHA-256, SHA-512 [34] and MD-6 [35].
There are saveral tools to calculate cryptographic hash function like hash tool 1.2, Crypto-precision and DNS [36].

Followigs are the properties of Hash function

i. **Performance:** It is easy to calculate $H(P)$ where $P$ is plaintext.

ii. **One way Function:** If $H(P)$ is given it is difficult to find $P$.

iii. **Weak Collision Resistance:** If $P$ and $H(P)$ are given it is very hard to find $P'$ such that $H(P) = H(P')$

iv. **Strong Collision Resistance:** It is hard to find $P$, $P'$ such that $H(P) = H(P')$."

# Chapter 3

# Noncommutative Cryptography Based on Groups

In this chapter we will review the research paper “Noncommutaitve Cryptographic Scheme Using Extra Special Group” presented by Saini and Kumar. In this paper Diffie-Helman like key exchange protocol is used. Also ElGamal like encryption/decryption algorithm is used. In the last section we will explain that the encryption/decryption scheme is vulnerable against known plaintext attack.

## 3.1 Extra Speacial Group

In this section we are going to define some basic definition related to extra special group

**Definition 3.1.1. (Cyclic Group)**
“A group $\mathbb{G}$ is called cyclic [37] if it is generated by single element. For example $\{1, \omega, \omega^2\}$ is a cyclic group with $\omega^3 = 1$.”

**Definition 3.1.2. (Center of Group):**
“If $\mathbb{G}$ is group then center of $\mathbb{G}$ [38] is denoted by $Z(\mathbb{G})$ and define as
$Z(\mathbb{G})$={for x $\in \mathbb{G}$ such that $xg = gx \ \forall \ g \in \mathbb{G}$}”

**Definition 3.1.3. (P-Group)**

"A finite group, whose order is power of some prime $p$ is called P-Group [39]. For example $D_4$ is P-Group of order $2^3$."

**Definition 3.1.4. (Heisenberg Group)**

It is a group of square, upper triangular matrices [40] of order 3 by 3. It is a group under multiplication, also it is a nonabelian group. For instance

$$\begin{pmatrix} 1 & p & q \\ 0 & 1 & r \\ 0 & 0 & 1 \end{pmatrix}$$

where element of $p, q, r$ belongs to any commutative ring. In Heisenberg matrices(three-dimension case) the multiplication is given as:

$$\begin{pmatrix} 1 & a & b \\ 0 & 1 & c \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & d & e \\ 0 & 1 & f \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & a+d & e+af+b \\ 0 & 1 & f+c \\ 0 & 0 & 1 \end{pmatrix}.$$

Inverse can be computed by the given general form

$$\begin{pmatrix} 1 & p & q \\ 0 & 1 & r \\ 0 & 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & -p & pr-q \\ 0 & 1 & -r \\ 0 & 0 & 1 \end{pmatrix}$$

**Definition 3.1.5. (Extra Speacial Group)**

A P-Group $\mathbb{G}$, whose center $Z(\mathbb{G})$ is cyclic and is of order prime $P$ , then it is called Extra Special Group. The order of Extra Special Group is always $p^{1+2n}$ where $n$ is some positive integer and it is denoted by $p^{1+2n}$.

### 3.1.1 Why Heisenberg Groups

The use of Heisenberg group in cryptography provides us an advantages not only in the computational cost but also resistance against cryptographic attacks. Note that, the poly-cyclic behavior $G_1 \triangleright G_2 \triangleright G_3 \triangleright ..... \triangleright G_{n+1}$ =1 of Heisenberg group makes the cost-effective implementation on software and hardware.

## 3.2 Noncommutative Cryptography

"Noncommutative cryptography on groups and rings [13], is the mathematical rationalization over matrix group or ring is exemplified on $M(\mathbb{Z}_{\mathbb{N}})$, based on $N = p.q$, where $p$ and $q$ are two secure primes. This is intractable, in view of the fact that $A = \begin{pmatrix} a & 0 \\ 0 & 0 \end{pmatrix} \in M(\mathbb{Z}_{\mathbb{N}})$, $a \in \mathbb{Z}_{\mathbb{N}}$, from $a^2$= $\begin{pmatrix} a^2 & 0 \\ 0 & 0 \end{pmatrix} \in M(\mathbb{Z}_{\mathbb{N}})$ with no significant factors of $N$.

The above mentioned ring can be enhanced with respect to security by using special or sparse matrices."

## 3.3 Key Exchange Protocol on Noncommutative Cryprography

The "noncommutative key exchange cryptography [13] works similar to Diffie-Hellman key exchange [3] like a commutative case", but the main difference is the listed actions on selecting of public parameters, formation of secret key, generation rules for common secret keys, and encryption-decryption. The efficaciousness of the protocol is dependent upon the impossibility of calculating the DLP [7]. The security of the protocol lies in the prime factorization [7] on two enormous primes. Random secret polynomial selected first by seder Y and then by receiver Z. A thorough explanation by the numerical example is elaborated in this section.

“**Key Exchange protocol over Noncommutaitve Ring**

Public Parameters

$$r, s \in \mathbb{Z}^+$$
$$\alpha, \beta\text{: ring elements}$$

Key Generation by user Y

i. User Y picks an arbitrary polynomial : $f_1(x)$.

ii. if $f_1(\alpha) \neq 0$, then $f_1(\alpha)$ is considered to be secret key.

iii. Public key generation $X_Y = f_1(\alpha)^r.\beta.f_1(\alpha)^s$.

Key generation by user Z

i. User Z picks an arbitrary polynomial : $g_1(x)$.

ii. if $g_1(\alpha) \neq 0$, then $g_1(\alpha)$ is considered to be secret key.

iii. Public key generation $X_Z = g_1(\alpha)^r.\beta.g_1(\alpha)^s$.”

Common session key generation by user Y

$$K_Y = f_1(\alpha)^r.X_Z.f_1(\alpha)^s.$$

Common session key generation by user Z

$$K_Z = g_1(\alpha)^r.X_Y.g_1(\alpha)^s.$$

This protocol can be explained by the following example

**Example 3.3.1.** Public parameters are

$$r = 3,\ s = 5,$$

$$\alpha = \begin{pmatrix} 17 & 5 \\ 7 & 4 \end{pmatrix}, \beta = \begin{pmatrix} 1 & 9 \\ 3 & 2 \end{pmatrix}$$

$$N = p.q = 77$$

User Y chooses an arbitrary polynomial $f_1(x) = 3x^3 + 4x^2 + 5x + 6$ and computes the polynomial on $f_1(\alpha)$, and if $f_1(\alpha) \neq 0$ then computed value will be secret key for user Y. Then user Y's secret key is given as:

$$f_1(\alpha) = 3 \begin{pmatrix} 17 & 5 \\ 7 & 4 \end{pmatrix}^3 + 4 \begin{pmatrix} 17 & 5 \\ 7 & 4 \end{pmatrix}^2 + 5 \begin{pmatrix} 17 & 5 \\ 7 & 4 \end{pmatrix} + 6I$$

$$= \begin{pmatrix} 19 & 20 \\ 28 & 44 \end{pmatrix} \bmod 77$$

Now, the public key generation $X_Y$ by user Y is given as:

$X_Y = f_1(\alpha)^r.\beta.f(\alpha)^s$

$$= \begin{pmatrix} 19 & 20 \\ 28 & 44 \end{pmatrix}^3 . \begin{pmatrix} 1 & 9 \\ 3 & 2 \end{pmatrix} . \begin{pmatrix} 19 & 20 \\ 28 & 44 \end{pmatrix}^5$$

$$= \begin{pmatrix} 3 & 56 \\ 9 & 2 \end{pmatrix} \bmod 77$$

At the other end user Z picks an arbitrary polynomial $g_1(x) = x^5 + 5x + 1$. Computes the polynomial $g_1(\alpha)$ and if $g_1(\alpha) \neq 0$ then the computed value will be secret key for user Z.

$$g_1(\alpha) = \begin{pmatrix} 17 & 5 \\ 7 & 4 \end{pmatrix}^5 + 5 \begin{pmatrix} 17 & 5 \\ 7 & 4 \end{pmatrix}^2 + 5 \begin{pmatrix} 17 & 5 \\ 7 & 4 \end{pmatrix} + 1.I$$

$$= \begin{pmatrix} 70 & 52 \\ 42 & 58 \end{pmatrix} \bmod 77$$

and the Public key generation $X_Z$ by user Z is given as:

$X_Z = g_1(\alpha)^r.\beta.g_1(\alpha)^s$

$$= \begin{pmatrix} 70 & 52 \\ 42 & 58 \end{pmatrix}^3 . \begin{pmatrix} 1 & 9 \\ 3 & 2 \end{pmatrix} . \begin{pmatrix} 70 & 52 \\ 42 & 58 \end{pmatrix}^5$$

$$= \begin{pmatrix} 0 & 39 \\ 35 & 68 \end{pmatrix} \text{ mod } 77$$

Lastly, common session key computed by the user Y as $K_Y$:

$K_Y = f_1(\alpha)^r.X_Z.f_1(\alpha)^s$

$$= \begin{pmatrix} 19 & 20 \\ 28 & 44 \end{pmatrix}^3 . \begin{pmatrix} 0 & 39 \\ 35 & 68 \end{pmatrix} . \begin{pmatrix} 19 & 20 \\ 28 & 44 \end{pmatrix}^5$$

$$= \begin{pmatrix} 21 & 37 \\ 49 & 69 \end{pmatrix} \text{ mod } 77$$

and the common session key computed by user Z as $K_Z$:

$K_Z = g_1(\alpha)^r.X_Y.g_1(\alpha)^s$

$$= \begin{pmatrix} 70 & 52 \\ 42 & 48 \end{pmatrix}^3 . \begin{pmatrix} 3 & 56 \\ 9 & 2 \end{pmatrix} . \begin{pmatrix} 70 & 52 \\ 42 & 48 \end{pmatrix}^5$$

$$= \begin{pmatrix} 21 & 37 \\ 49 & 69 \end{pmatrix} \text{ mod } 77$$

### 3.3.1 Key Exchange Protocol over Heisenberg Group

The protocol that are used in Section 3.3 is applicable on Heisenberg group. It is demonstrated on the public parameters, with following supposition

$$r, s \in \mathbb{Z}^+$$
$$r = 3,\ s = 5$$

$$\alpha, \beta\text{: Heisenberg Group element over } M(\mathbb{Z}_{\mathbb{N}})$$

$$\alpha = \begin{pmatrix} 1 & 5 & 7 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix},\ \beta = \begin{pmatrix} 1 & 6 & 9 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix}$$

$$N = p.q = 77$$
$$N = p \cdot q \text{ where } p \text{ and } q \text{ are two secure prime numbers}$$

User Y picks an arbitrary polynomial $f_1(x) = 3x^3 + 4x^2 + 5x + 6$. Compute the polynomial $f_1(\alpha)$, and if $f_1(\alpha) \neq 0$ then computed value will be secret key for user Y. Then Y's secret key is given as:

$$f_1(\alpha) = 3\begin{pmatrix} 1 & 5 & 7 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix}^3 + 4\begin{pmatrix} 1 & 5 & 7 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix}^2 + 5\begin{pmatrix} 1 & 5 & 7 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix} + 6I$$

$$= \begin{pmatrix} 18 & 33 & 69 \\ 0 & 18 & 11 \\ 0 & 0 & 18 \end{pmatrix} \bmod 77$$

Now, the public key generation $X_Y$ by user Y is given as:

$X_Y = f_1(\alpha)^r.\beta.f_1(\alpha)^s$

$$= \begin{pmatrix} 18 & 33 & 29 \\ 0 & 18 & 11 \\ 0 & 0 & 1 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & 6 & 9 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 18 & 33 & 29 \\ 0 & 18 & 11 \\ 0 & 0 & 18 \end{pmatrix}^5$$

$$= \begin{pmatrix} 18 & 33 & 29 \\ 0 & 18 & 11 \\ 0 & 0 & 18 \end{pmatrix} \bmod 77$$

At the other end user Z picks an aribtrary polynomial $g_1(x) = x^5 + 5x + 1$. He computes the polynomial $g_1(\alpha)$, and if $g_1(\alpha) \neq 0$ then computed value value will be secret key for user Z:

$$g_1(\alpha) = 5 \begin{pmatrix} 1 & 5 & 7 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 5 & 7 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix}^5 +1$$

$$= \begin{pmatrix} 7 & 50 & 39 \\ 0 & 7 & 40 \\ 0 & 0 & 7 \end{pmatrix} \bmod 77$$

and the public key generation $X_Z$ is given as:

$$X_B = g_1(\alpha)^r.\beta.g_1(\alpha)^s$$

$$= \begin{pmatrix} 7 & 50 & 39 \\ 0 & 7 & 40 \\ 0 & 0 & 7 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & 6 & 9 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 7 & 50 & 39 \\ 0 & 7 & 40 \\ 0 & 0 & 7 \end{pmatrix}^5$$

$$= \begin{pmatrix} 42 & 56 & 35 \\ 0 & 42 & 0 \\ 0 & 0 & 42 \end{pmatrix} \bmod 77$$

Lastly, the common session key computed by user Y as $K_Y$ is given as:

$K_Y = f_1(\alpha)^r.X_Z.f_1(\alpha)^s$

$$= \begin{pmatrix} 18 & 33 & 29 \\ 0 & 18 & 11 \\ 0 & 0 & 1 \end{pmatrix}^3 \cdot \begin{pmatrix} 42 & 56 & 35 \\ 0 & 42 & 0 \\ 0 & 0 & 42 \end{pmatrix} \cdot \begin{pmatrix} 18 & 33 & 29 \\ 0 & 18 & 11 \\ 0 & 0 & 18 \end{pmatrix}^5$$

$$= \begin{pmatrix} 70 & 42 & 28 \\ 0 & 70 & 0 \\ 0 & 0 & 70 \end{pmatrix} \bmod 77.$$

The common session key computed by user Z as $K_Z$ is given as:

$K_Z = g_1(\alpha)^r.X_Y.g_1(\alpha)^s$

$$= \begin{pmatrix} 7 & 50 & 39 \\ 0 & 7 & 40 \\ 0 & 0 & 7 \end{pmatrix}^3 \cdot \begin{pmatrix} 9 & 32 & 10 \\ 0 & 9 & 71 \\ 0 & 0 & 9 \end{pmatrix} \cdot \begin{pmatrix} 7 & 50 & 39 \\ 0 & 7 & 40 \\ 0 & 0 & 7 \end{pmatrix}^5$$

$$= \begin{pmatrix} 70 & 42 & 28 \\ 0 & 70 & 0 \\ 0 & 0 & 70 \end{pmatrix} \bmod 77$$

## 3.4 Cryptosystem Based on Heisenberg Group

First of all, we are going to define $M$ as plain-text space and $H$ as hash function [41]. For simplicity, we suppose that $M = M_3(\mathbb{Z}_{\mathbb{N}})$, while hash function is defined as

$H : M_3(\mathbb{Z}_{\mathbb{N}}) \mapsto P = M_3(\mathbb{Z}_{\mathbb{N}}), m_{ij} \mapsto 2^{m_{ij}} \bmod N$

### 3.4.1 Encryption-Decryption Algorithm

“Encryption-Decryption Algorithm on Heisenberg group [13] is as follows
Public Parameters

$r, s \in \mathbb{Z}^+$
$\alpha, \beta$ are the elements of Heisenberg Group
$M$: Plaintext
$H(P)$: Hashed text

Key Generation by user Y

i. User Y picks an arbitrary polynomial : $f_1(x)$.

ii. If $f_1(\alpha) \neq 0$, then $f_1(\alpha)$ is considered to be secret key.

iii. Public key generation $X_Y = f_1(\alpha)^r.\beta.f_1(\alpha)^s$.

Key generation by user Z

i. User Z picks an arbitrary polynomial : $g_1(x)$.

ii. If $g_1(\alpha) \neq 0$, then $g_1(\alpha)$ is considered to be secret key.

iii. Public key generation $X_Z = g_1(\alpha)^r.\beta.g_1(\alpha)^s$.”

Common session key generation by user Y

$$K_Y = f_1(\alpha)^r.X_Z.f_1(\alpha)^s.$$

Common session key generation by user Z

$$K_Z = g_1(\alpha)^r.X_Y.g_1(\alpha)^s.$$

Encryption by user Z

C: cipher-text

D: decyption key

$$C = g_1(\alpha)^r.\beta.g_1(\alpha)^s \ , \ D = H(g_1(\alpha)^r.X_Y.g_1(\alpha)^s) \oplus P$$

Decryption by user Y

$$P = H(f_1(\alpha)^r.X_Z.f_1(\alpha)^s) \oplus D$$

**Example 3.4.1.** The public parameters

$r, s \in \mathbb{Z}^+$ such that

$r = 3, s = 5$

$\alpha, \beta$ are the elements of Heisenberg Group over $M(\mathbb{Z}_\mathbb{N})$

$$\alpha = \begin{pmatrix} 1 & 5 & 9 \\ 0 & 1 & 9 \\ 0 & 0 & 1 \end{pmatrix} , \beta = \begin{pmatrix} 1 & 9 & 5 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix}$$

$N = p.q = 77$ where $p$ and $q$ are prime numbers

$$M = \begin{pmatrix} 27 & 19 & 25 \\ 34 & 8 & 7 \\ 45 & 5 & 9 \end{pmatrix}$$

User Y picks an arbitrary polynomial $f_1(x) = 3x^3 + 4x^2 + 5x + 6$. Compute the polynomial $f_1(\alpha)$, and if $f_1(\alpha) \neq 0$ then computed value will be secret key for user Y. Hence Y's secret key is given as:

$$f_1(\alpha) = 3\begin{pmatrix} 1 & 5 & 9 \\ 0 & 1 & 9 \\ 0 & 0 & 1 \end{pmatrix}^3 + 4\begin{pmatrix} 1 & 5 & 9 \\ 0 & 1 & 9 \\ 0 & 0 & 1 \end{pmatrix}^2 + 5\begin{pmatrix} 1 & 5 & 9 \\ 0 & 1 & 9 \\ 0 & 0 & 1 \end{pmatrix} + 6\mathrm{I}$$

$$= \begin{pmatrix} 18 & 33 & 13 \\ 0 & 18 & 44 \\ 0 & 0 & 18 \end{pmatrix} \bmod 77.$$

The public key generation $X_Y$ by user Y is given as:

$$X_Y = f_1(\alpha)^r.\beta.f_1(\alpha)^s$$

$$= \begin{pmatrix} 18 & 33 & 13 \\ 0 & 18 & 44 \\ 0 & 0 & 18 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & 9 & 5 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 18 & 33 & 13 \\ 0 & 18 & 44 \\ 0 & 0 & 18 \end{pmatrix}^5$$

$$= \begin{pmatrix} 9 & 59 & 42 \\ 0 & 9 & 49 \\ 0 & 0 & 9 \end{pmatrix} \bmod 77$$

Moving onwards, user Z picks an arbitarary polynomial $g_1(x) = x^5 + 5x + 1$. Computes the polynomial $g_1(\alpha)$, and if $g_1(\alpha) \neq 0$ then computed value will be secret key for user B:

$$g_1(\alpha) = 5\begin{pmatrix} 1 & 5 & 9 \\ 0 & 1 & 9 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 5 & 9 \\ 0 & 1 & 9 \\ 0 & 0 & 1 \end{pmatrix}^5 + 1I$$

$$= \begin{pmatrix} 7 & 50 & 1 \\ 0 & 7 & 13 \\ 0 & 0 & 7 \end{pmatrix} \bmod 77$$

and the public key generation $X_Z$ by user Z is given as:

$$X_Z = g_1(\alpha)^r \cdot \beta \cdot g_1(\alpha)^s$$

$$= \begin{pmatrix} 7 & 50 & 1 \\ 0 & 7 & 13 \\ 0 & 0 & 7 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & 9 & 5 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 7 & 50 & 1 \\ 0 & 7 & 13 \\ 0 & 0 & 7 \end{pmatrix}^5$$

$$= \begin{pmatrix} 42 & 28 & 35 \\ 0 & 42 & 35 \\ 0 & 0 & 42 \end{pmatrix} \bmod 77$$

The public key of user Z is considered as ciphertext (in our case user Z is sender and user Y is receiver):

$$C = g_1(\alpha)^r.\beta.g_1(\alpha)^s = \begin{pmatrix} 42 & 28 & 35 \\ 0 & 42 & 35 \\ 0 & 0 & 42 \end{pmatrix},$$

$$D = H(g_1(\alpha)^r.X_Y.g_1(\alpha)^s) \oplus M$$

$$= H\left(\begin{pmatrix} 7 & 50 & 1 \\ 0 & 7 & 13 \\ 0 & 0 & 7 \end{pmatrix}^3 \cdot \begin{pmatrix} 9 & 59 & 42 \\ 0 & 9 & 49 \\ 0 & 0 & 9 \end{pmatrix} \cdot \begin{pmatrix} 9 & 59 & 42 \\ 0 & 9 & 49 \\ 0 & 0 & 9 \end{pmatrix}^5\right) \oplus \begin{pmatrix} 27 & 19 & 25 \\ 34 & 8 & 7 \\ 45 & 5 & 9 \end{pmatrix}$$

$$= H\left(\begin{pmatrix} 70 & 21 & 25 \\ 0 & 70 & 7 \\ 0 & 0 & 70 \end{pmatrix}\right) \oplus \begin{pmatrix} 27 & 19 & 25 \\ 34 & 8 & 7 \\ 45 & 5 & 9 \end{pmatrix}$$

$$= \begin{pmatrix} 2^{70} & 2^{21} & 2^{25} \\ 2^{0} & 2^{70} & 2^{7} \\ 2^{0} & 2^{0} & 2^{70} \end{pmatrix} \oplus \begin{pmatrix} 27 & 19 & 25 \\ 34 & 8 & 7 \\ 45 & 5 & 9 \end{pmatrix}$$

$$D = \begin{pmatrix} 12 & 42 & 57 \\ 35 & 31 & 52 \\ 44 & 4 & 30 \end{pmatrix} \bmod 77$$

Plaintext is given as follows:

$M = H(f_1(\alpha)^r.C.f_1(\alpha)^s) \oplus D$

$$= H\left(\begin{pmatrix} 18 & 33 & 13 \\ 0 & 18 & 44 \\ 0 & 0 & 18 \end{pmatrix}^3 \cdot \begin{pmatrix} 42 & 28 & 35 \\ 0 & 42 & 35 \\ 0 & 0 & 42 \end{pmatrix} \cdot \begin{pmatrix} 18 & 33 & 13 \\ 0 & 18 & 44 \\ 0 & 0 & 18 \end{pmatrix}^5\right) \oplus \begin{pmatrix} 12 & 42 & 57 \\ 35 & 31 & 52 \\ 44 & 4 & 30 \end{pmatrix}$$

$$= H\left(\begin{pmatrix} 70 & 21 & 25 \\ 0 & 70 & 7 \\ 0 & 0 & 70 \end{pmatrix}\right) \oplus \begin{pmatrix} 12 & 42 & 57 \\ 35 & 31 & 52 \\ 44 & 4 & 30 \end{pmatrix}$$

$$= \begin{pmatrix} 2^{70} & 2^{21} & 2^{25} \\ 2^0 & 2^{70} & 2^7 \\ 2^0 & 2^0 & 2^{70} \end{pmatrix} \oplus \begin{pmatrix} 12 & 42 & 57 \\ 35 & 31 & 52 \\ 44 & 4 & 30 \end{pmatrix}$$

$$M = \begin{pmatrix} 27 & 19 & 25 \\ 34 & 8 & 7 \\ 45 & 5 & 9 \end{pmatrix} \mod 77$$

Hence we get plaintext M.

## 3.5 Cryptanalysis

In this section we have shown that the encryption/decryption scheme of Saini and Kumar [13], is vulnerable to known plaintext attack.

Recall that in a known plaintext attack we assume that an attacker has the knowledge of plaintext and ciphertext pairs like $(m_1, c_1), (m_2, c_2), ..., (m_n, c_n)$. Note that the ciphertext $D$ for the plaintext $M$ is

$$D = H(g_1(\alpha)^r.X_Y.g_1(\alpha)^s) \oplus M \tag{3.1}$$

and the decryption is done as

$$M = H(f_1(\alpha)^r.X_Z.f_1(\alpha)^s) \oplus D \tag{3.2}$$

with the property that

$$H(g_1(\alpha)^r.X_Y.g_1(\alpha)^s) = H(f_1(\alpha)^r.X_Z.f_1(\alpha)^s)$$

If an attacker has the knowledge of both $M$ and $D$ then from equation (3.1)

$$H(g_1(\alpha)^r.X_Y.g_1(\alpha)^s) = D \oplus M \tag{3.3}$$

Therefore the scheme will not be secure for any subsequent encryption. We will illustrate the attack by applying it on the example given in [13].

**Example 3.5.1.** Consider the message $M$, ciphertext $D$ of Example 3.4.1

$$M = \begin{pmatrix} 27 & 19 & 25 \\ 34 & 8 & 7 \\ 45 & 5 & 9 \end{pmatrix}, D = \begin{pmatrix} 12 & 42 & 57 \\ 35 & 31 & 52 \\ 44 & 4 & 30 \end{pmatrix} \text{ and mod=77}$$

Since

$$M = H(f_1(\alpha)^r.X_Z.f_1(\alpha)^s) \oplus D$$

therefore

$$H(f_1(\alpha)^r.X_Z.f_1(\alpha)^s) = M \oplus D$$

$$H(f_1(\alpha)^r.X_Z.f_1(\alpha)^s) = \begin{pmatrix} 27 & 19 & 25 \\ 34 & 8 & 7 \\ 45 & 5 & 9 \end{pmatrix} \oplus \begin{pmatrix} 12 & 42 & 57 \\ 35 & 31 & 52 \\ 44 & 4 & 30 \end{pmatrix}$$

$$H\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} 27 \oplus 12 & 19 \oplus 42 & 25 \oplus 57 \\ 34 \oplus 35 & 8 \oplus 31 & 7 \oplus 52 \\ 45 \oplus 44 & 5 \oplus 4 & 9 \oplus 30 \end{pmatrix}$$

$$= \begin{pmatrix} 23 & 57 & 32 \\ 1 & 23 & 51 \\ 1 & 1 & 23 \end{pmatrix} \bmod 77$$

Therefore the encryption/decryption scheme of Saini and Kumar is vulnerable to known plaintext attack. We have found a hash value of share secret key, which is used for both encryption-decryption.

# Chapter 4

# Noncommutative Key Exchange using Heisenberg Group Over Galois Field

In this chapter, we will represent and discuss a modified form of the key exchange on noncommutative cryptographic scheme proposed by Saini and Kumar[13]. For this purpose we aim to use extra special group of matrices over a Galois field ($GF(p^n)$). The key exchange is part of the proposed noncommutative cryptosystem is like Diffie-Hellman [3] key exchange in the abelian case, but the main difference is the listed operations on choosing the public parameters, generating of secret keys, generation rules for common secret keys and encryption-decryption. Security of the scheme relies on the impossibility of calculating the symmetrical decomposition and conjugacy search problem [7]. Particularly, an attacker has to factorize public polynomial which is the product of the arbitrary secret irreducible polynomial over $\mathbb{Z}_p$. In fact, the attacker has to solve exponential equations, that is $X = f_1(\alpha)^r \cdot \beta \cdot (f(\alpha)^{-1})^s$ and it is hard to find $f_1(\alpha)$ from the knowledge of public parameters. Illustrative examples is given to explain the working of the proposed scheme.

## 4.1 The Proposed Key Exchange Protocol

In this section, we will explain a modified form of the key exchange which was explained in Chapter 3.

**Algorithm 4.1.1 (Key Exchange protocol over Galois Field)**

Public Parameters:

$r, s \in \mathbb{Z}^+$

$\alpha, \beta$: Heisenberg Group element over $GF(p^n)$

$N(x) = P(x) \cdot Q(x)$ both $P(x), Q(x)$ are irreducible polynomials over $GF(p^n)$

Key Generation by user Y

i. User Y picks an arbitrary modulo prime based polynomial : $f_1(x)$

ii. If $f_1(\alpha) \neq 0$ and $f_1(\alpha) \in GF(p^n)$ , then $f_1(\alpha)$ is supposed to be secret key

iii. Public key formation $X_Y = f_1(\alpha)^r.\beta.(f_1(\alpha)^{-1})^s \bmod N(x)$

Key generation by user Z

i. User Z picks an arbitrary modulo prime based polynomial : $g_1(x)$

ii. If $g_1(\alpha) \neq 0$ and $g_1(\alpha) \in GF(p^n)$ , then $g_1(\alpha)$ is supposed to be secret key

iii. Public key formation $X_Z = g_1(\alpha)^r.\beta.(g_1(\alpha)^{-1})^s \bmod N(x)$

Common session key generation by user Y

$$K_Y = f_1(\alpha)^r.X_Z.(f_1(\alpha)^{-1})^s$$

Common session key generation by user Z

$$K_Z = g_1(\alpha)^r.X_Y.(g_1(\alpha)^{-1})^s$$

**Correctness**

User Y calculates

$$K_Y = f_1(\alpha)^r.X_Z.(f_1(\alpha)^{-1})^s \tag{4.1}$$

but $X_Z = g_1(\alpha)^r.\beta.(g_1(\alpha)^{-1})^s$

therefore

$$K_Y = f_1(\alpha)^r.g_1(\alpha)^r.\beta.(g_1(\alpha)^{-1})^s.(f_1(\alpha)^{-1})^s \tag{4.2}$$

since polynomial multiplication is commutative therefore

$$f_1(\alpha)g_1(\alpha) = g_1(\alpha)f_1(\alpha) \text{ and } f_1(\alpha)^{-1}.g_1(\alpha)^{-1} = g_1(\alpha)^{-1}.f_1(\alpha)^{-1}$$

let

$$f_1(\alpha)g_1(\alpha) = g_1(\alpha)f_1(\alpha) = P \tag{4.3}$$

and

$$f_1(\alpha)^{-1}.g_1(\alpha)^{-1} = g_1(\alpha)^{-1}.f_1(\alpha)^{-1} = Q \tag{4.4}$$

Equation (4.2) becomes

$$K_Y = P^r.\beta.Q^s \tag{4.5}$$

On the other hand user B computes

$$K_Z = g_1(\alpha)^r.X_Y.(g_1(\alpha)^{-1})^s \tag{4.6}$$

but $X_Y = f_1(\alpha)^r.\beta.(f_1(\alpha)^{-1})^s$

therefore

$$K_Z = g_1(\alpha)^r.f_1(\alpha)^r.\beta.(f_1(\alpha)^{-1})^s.(g_1(\alpha)^{-1})^s \tag{4.7}$$

since polynomial multiplication is commutative therefore

$$f_1(\alpha)g_1(\alpha) = g_1(\alpha)f_1(\alpha),\ f_1(\alpha)^{-1}.g_1(\alpha)^{-1} = g_1(\alpha)^{-1}.f_1(\alpha)^{-1}$$

let

$$f_1(\alpha)g_1(\alpha) = g_1(\alpha)f_1(\alpha) = P \tag{4.8}$$

and

$$f_1(\alpha)^{-1}.g_1(\alpha)^{-1} = g_1(\alpha)^{-1}.f_1(\alpha)^{-1} = Q \tag{4.9}$$

Equation (4.7) becomes

$$K_Z = P^r.\beta.Q^s \tag{4.10}$$

From Equation (4.5) and (4.10)

$$K_Y = K_Z$$

**Example 4.1.1.** (Key Exchange on Galois Field $GF(5^2)$)

The Public parameters are

$$r, s \in \mathbb{Z}^+ \text{ such that}$$
$$r = 3,\ s = 2$$
$$\alpha, \beta\text{: Heisenberg Group element over } GF(5^2)$$

$$\alpha = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{pmatrix}, \beta = \begin{pmatrix} 1 & 2x+3 & 4 \\ 0 & 1 & 4x \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{aligned} N(x) &= p(x) \cdot q(x) \\ &= 3x^2 + 3 \end{aligned}$$

**Generation of secret key by user Y:**

User Y, picks an arbitrary polynomial $f_1(x) = x^4 + 3x^2 + x + 4$. Compute the polynomial on $f_1(\alpha)$, and if $f_1(\alpha) \neq 0 \in M(GF(5^2))$ then computed value is considered to be secret key of user Y:

$$f_1(\alpha) = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{pmatrix}^4 + 3 \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{pmatrix}^2 + \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 3 & 2x+2 \\ 0 & 1 & 4x \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 3 & 2 & x+3 \\ 0 & 3 & x \\ 0 & 0 & 3 \end{pmatrix} + \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix}$$

$$= \begin{pmatrix} 4 & 2 & 3x+3 \\ 0 & 4 & x \\ 0 & 0 & 4 \end{pmatrix} \bmod 3x^2+3$$

and computing $f_1(a)^{-1}$ by using our implementation in ApCoCoA, we get

$$(f(\alpha))^{-1} = \begin{pmatrix} 4 & 3 & 2 \\ 0 & 4 & 4x \\ 0 & 0 & 4 \end{pmatrix} \bmod 3x^2+3$$

**The public key generation $X_Y$ by user Y is given as:**

$X_Y = f_1(\alpha)^3.\beta.(f_1(\alpha)^{-1})^2$

$$X_Y = \begin{pmatrix} 4 & 2 & 3x+3 \\ 0 & 4 & x \\ 0 & 0 & 4 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & 2x+3 & 4 \\ 0 & 1 & 4x \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 4 & 3 & 2 \\ 0 & 4 & 4x \\ 0 & 0 & 4 \end{pmatrix}^2$$

$$= \begin{pmatrix} 4 & 1 & 3x+4 \\ 0 & 4 & 3x \\ 0 & 0 & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2x+3 & 4 \\ 0 & 1 & 4x \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 4 & 2x+1 \\ 0 & 1 & 2x \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 4 & 3x+4 & x+3 \\ 0 & 4 & 2x \\ 0 & 0 & 4 \end{pmatrix} \bmod 3x^2+3$$

**Generation of secret key by user Z:**

User Z picks an arbitrary polynomial $g_(x) = 4x^4 + 3x + 2$. Compute the polynomial on $g_1(\alpha)$, if $g_1(\alpha) \neq 0$ and $g_1(\alpha) \in GF(5^2)$ then computed value will be secret key for user Z:

$$g_1(\alpha) = 4\begin{pmatrix}1 & 2 & 3\\0 & 1 & x\\0 & 0 & 1\end{pmatrix}^4 + 3\begin{pmatrix}1 & 2 & 3\\0 & 1 & x\\0 & 0 & 1\end{pmatrix} + \begin{pmatrix}2 & 0 & 0\\0 & 2 & 0\\0 & 0 & 2\end{pmatrix}$$

$$= \begin{pmatrix}4 & 2 & 3x+3\\0 & 4 & x\\0 & 0 & 4\end{pmatrix} + \begin{pmatrix}3 & 1 & 4\\0 & 3 & 3x\\0 & 0 & 3\end{pmatrix} + \begin{pmatrix}2 & 0 & 0\\0 & 2 & 0\\0 & 0 & 2\end{pmatrix}$$

$$g_1(\alpha) = \begin{pmatrix}4 & 3 & 3x+2\\0 & 4 & 4x\\0 & 0 & 4\end{pmatrix} \bmod 3x^2+3$$

and

$$(g_1(\alpha))^{-1}{=}\begin{pmatrix}4 & 2 & 3\\0 & 4 & x\\0 & 0 & 4\end{pmatrix} \bmod 3x^2+3$$

**The public key generation $X_Z$ by user Z is given as:**

$$X_Z = g_1(\alpha)^3.\beta.(g_1(\alpha)^{-1})^2$$

$$X_Z = \begin{pmatrix}4 & 3 & 3x+2\\0 & 4 & 4x\\0 & 0 & 4\end{pmatrix}^3 \cdot \begin{pmatrix}1 & 2x+3 & 4\\0 & 1 & 4x\\0 & 0 & 1\end{pmatrix} \cdot \begin{pmatrix}4 & 2 & 3\\0 & 4 & x\\0 & 0 & 4\end{pmatrix}^2$$

$$= \begin{pmatrix}4 & 4 & 3x+1\\0 & 4 & 2x\\0 & 0 & 4\end{pmatrix} \cdot \begin{pmatrix}1 & 2x+3 & 4\\0 & 1 & 4x\\0 & 0 & 1\end{pmatrix} \cdot \begin{pmatrix}1 & 1 & 2x+4\\0 & 1 & 3x\\0 & 0 & 1\end{pmatrix}$$

$$= \begin{pmatrix}4 & 3x & 4\\0 & 4 & 0\\0 & 0 & 4\end{pmatrix} \bmod 3x^2+3$$

**Lastly, common secret key computed by the user Y as $K_Y$ is:**

$K_Y = f_1(\alpha)^3.X_Z.(f_1(\alpha)^{-1})^2$

$$K_Y = \begin{pmatrix} 4 & 2 & 3x+3 \\ 0 & 4 & x \\ 0 & 0 & 4 \end{pmatrix}^3 \cdot \begin{pmatrix} 4 & 3x & x+3 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix} \cdot \begin{pmatrix} 4 & 3 & 2 \\ 0 & 4 & 4x \\ 0 & 0 & 4 \end{pmatrix}^2$$

$$= \begin{pmatrix} 4 & 1 & 3x+4 \\ 0 & 4 & 3x \\ 0 & 0 & 4 \end{pmatrix} \cdot \begin{pmatrix} 4 & 3x & 4 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & 4 & 2x+1 \\ 0 & 1 & 2x \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 2x+3 & 2x+4 \\ 0 & 1 & 4x \\ 0 & 0 & 1 \end{pmatrix} \bmod 3x^2+3$$

**and the common secret key computed by the user Z as $K_Z$ is:**

$K_Z = g_1(\alpha)^3.X_Y.(g_1(\alpha)^{-1})^2$

$$K_Z = \begin{pmatrix} 4 & 3 & 3x+2 \\ 0 & 4 & 4x \\ 0 & 0 & 4 \end{pmatrix}^3 \cdot \begin{pmatrix} 4 & 3x+4 & x+3 \\ 0 & 4 & 2x \\ 0 & 0 & 4 \end{pmatrix} \cdot \begin{pmatrix} 4 & 2 & 3 \\ 0 & 4 & x \\ 0 & 0 & 4 \end{pmatrix}^2$$

$$= \begin{pmatrix} 4 & 1 & 3x+4 \\ 0 & 4 & 3x \\ 0 & 0 & 4 \end{pmatrix} \cdot \begin{pmatrix} 4 & 3x & 4 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & 4 & 2x+1 \\ 0 & 1 & 2x \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 2x+3 & 2x+4 \\ 0 & 1 & 4x \\ 0 & 0 & 1 \end{pmatrix} \bmod 3x^2+3$$

**Example 4.1.2.** (Key Exchange on $GF(2^n)$)

Public parameters are

$$r, s \in \mathbb{Z}^+$$

$$r = 3,\ s = 2$$

$$\alpha, \beta\text{: Heisenberg Group element over } GF(2^8)$$

$$\alpha = \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}, \beta = \begin{pmatrix} 1 & y^2+1 & y \\ 0 & 1 & y^3 \\ 0 & 0 & 1 \end{pmatrix}$$

$$N(y) = P(y) \cdot Q(y)$$

$$N(y) = y^4 + y^3 + y + 1$$

User Y picks an arbitrary polynomial $f(y) = y^3 + y^2 + 1$. Compute the polynomial $f_1(\alpha)$, and if $f_1(\alpha) \neq 0 \in M(GF(2^8))$ then computed value will be secret key for user Y:

$$f_1(\alpha) = \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^3 + \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^2 + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & y & y^3+y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & y^3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} \text{mod } y^4 + y^3 + y + 1$$

and computing $(f_1(\alpha))^{-1}$ by using our implementation in ApCoCoa, we get

$$(f_1(\alpha))^{-1} = \begin{pmatrix} 1 & y & y^3+y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} \bmod y^4+y^3+y+1$$

Now the public key generation $X_Y$ by user Y is given as:

$X_Y = f_1(\alpha)^3.\beta.(f_1(\alpha)^{-1})^2$

$$X_Y = \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & 2y+3 & 4 \\ 0 & 1 & 4y \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & y & y^3+y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^2$$

$$= \begin{pmatrix} 1 & y & y^3 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & y^2+1 & y \\ 0 & 1 & y^3 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & y^3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & y^2+y+1 & y^3+1 \\ 0 & 1 & y^3+y^2 \\ 0 & 0 & 1 \end{pmatrix} \bmod y^4+y^3+y+1$$

At the other end user Z picks an arbitrary polynomial $g_1(y) = y^5 + y^2 + 1$. Compute the polynomial $g_1(\alpha)$, if $g_1(\alpha) \neq 0 \in GF(2^8)$ then computed value will be secret key for user Z:

$$g_1(\alpha) = \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^5 + \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^2 + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & y^3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$g_1(\alpha) = \begin{pmatrix} 1 & y & y^3+y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} \bmod y^4+y^3+y+1$$

and

$$(g_1(\alpha))^{-1} = \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} \bmod y^4+y^3+y+1$$

The public key Generation $X_Z$ by user Z is given as

$X_Z = g_1(\alpha)^3.\beta.(g_1(\alpha)^{-1})^2$

$$X_Z = \begin{pmatrix} 1 & y & y^3+y+1 \\ 0 & 1 & x^2 \\ 0 & 0 & 1 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & y^2+1 & y \\ 0 & 1 & y^3 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^2$$

$$= \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & y^2+1 & y \\ 0 & 1 & y^3 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & y^3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & y^2+y+1 & y \\ 0 & 1 & y^3+y^2 \\ 0 & 0 & 1 \end{pmatrix} \bmod y^4+y^3+y+1$$

Common session key $K_Y$ computed by the user Y is given as:

$K_Y = f_1(\alpha)^3.X_Z.(f_1(\alpha)^{-1})^2$

$$K_Y = \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^3 \\ 0 & 0 & 1 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & y^2+y+1 & y \\ 0 & 1 & y^3+y^2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & y & y^3+y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^2$$

$$= \begin{pmatrix} 1 & y & y^3 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & y^2+y+1 & y \\ 0 & 1 & y^3+y^2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & y^3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & y^2+1 & 1 \\ 0 & 1 & y^3 \\ 0 & 0 & 1 \end{pmatrix} \bmod y^4+y^3+y+1$$

Common session key $K_Z$ computed by the user Z is given as:

$K_Z = g_1(\alpha)^3.X_Y.(g_1(\alpha)^{-1})^2$

$$K_Z = \begin{pmatrix} 1 & y & y^3+y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^3 \cdot \begin{pmatrix} 1 & y^2+y+1 & y^3+1 \\ 0 & 1 & y^3+y^2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix}^2$$

$$= \begin{pmatrix} 1 & y & y+1 \\ 0 & 1 & y^2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & y^2+y+1 & y^3+1 \\ 0 & 1 & y^3+y^2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & y^3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & y^2+1 & 1 \\ 0 & 1 & y^3 \\ 0 & 0 & 1 \end{pmatrix} \bmod y^4+y^3+y+1$$

We heve suggest that for our proposed key exchange protocol one can use any secure encryption/decryption algorithm.

# Chapter 5

# Security Analysis and Conclusion

In this chapter we will explain security analysis of the proposed key exchange protocol, also we will discussed conclusion and future work.

## 5.1 Security Analysis of Key Exchange Protocol

Computational difficulty or complexity analysis with its associated strength as security and accomplishment consideration is explained in this section.

### 5.1.1 Irreducible Factors of Polynomial $N(x)$

The working method offered is based on irreducible factorization of polynomial $N(x)$. Irreducible factors of $N(x)$ are hidden in the offered algorithm, but because of obvious clarity it is presented whenever required. The following points indicates the powerful security analysis.

i. As $N(x) = p(x) \cdot q(x)$ is based on two irreducible polynomials and it is quite difficult to find exact factors of $N(x)$, if $N(x)$ is the large degree polynomial.

ii. The time required for irreducible polynomial factors increases exponentially, so if the large degree polynomial is used in algorithm, it is improbable to get its irreducible factorization.

### 5.1.2 Secret Keys:

A private key formation is based on arbitrary selected polynomials $f_1(x)$ or $g_1(x)$. Polynomial which cannot be further reduced is called irreducible polynomial. There are three different types of polynomials such as:

i. Usual polynomials

ii. Polynomials based on modulo prime

iii. Polynomials based on modulo prime defined on other polynomials which have some power $n$, where $n$ is integer.

In class (i) Arithmetic operations (+,- ×) are performed on polynomials using the usual principal of algebra. Division is only workable, if coefficient are taken from field elements.
Class (ii) has a similar number juggling activity as class (i), yet the division can be utilized in remainder and quotient form. This exhibits that polynomial based on modulo prime is useful in cryptography.
And in class (iii) the polynomials of Galois fields $GF(p^n)$ are used.

### 5.1.3 The Public Keys:

Public key of senders and receivers has its basis on the polynomial function $f_1(x)$ or $g_1(x)$ to the power of $r$ and polynomial functions $(f_1(x))^{-1}$ or $(g_1(x))^{-1}$ to the power of $s$ with multiplication in modulo a prime. Attacker try to recover the private key from the public keys which are easily accessible. For the algorithm symmetrical decomposition problem and the conjugacy search problem are used in public keys which are hard to find. From the offered scheme, factors of $N(x)$(standard length of 160 bits) may be enough for preventing from the attacker to get valuable ideas or valid private keys.

### 5.1.4 Brute-Force Attack:

The brute-force attack is to find all possible private keys. There is larger arbitrariness and uncertain behavior on smaller key length on our modified scheme. It is a particular case of ECC, hence the attack is effective on a shorter length keys. A shorter length keys have shorter process time, so the brute force attack works on shorter length keys. Regarding the speed, efficiency and cryptanalysis non-commutative approach is better as compared to ECC [5] and RSA [4] algorithm.

## 5.2 Conclusion

In this thesis, we have applied a new platform on research paper **"Non commutative Cryptographic Scheme Using Extra Special group"** [13]. Also we have proved that the scheme is vulnerable for known plaintext attack. In order to increase the security of the scheme, we have involved conjugacy search problem together with symmetrical decomposition problem. Also the security of key exchange is improved by taking matrices from Galois field $GF(p^n)$. In fact, the attacker has to solve exponential equations, that is $X = f_1(\alpha)^r \cdot \beta \cdot (f(\alpha)^{-1})^s$ and it is hard to find $f_1(\alpha)$ from the knowledge of public parameters. The Overall security of the scheme is increased by using Galois Field $GF(p^n)$. We have given security analysis of our scheme. One can extend our work by checking the possibility of minus-plus algebra.

# Bibliography

[1] J. W. Ceaser, *Presidential selection: Theory and development.* Princeton University Press, 1979.

[2] R. Churchhouse and R. Churchhouse, *Codes and ciphers: Julius Caesar, the Enigma and the Internet.* Cambridge University Press, 2002.

[3] W.Diffie and M.E.Hellman, "New direction in cryptography," *IEEE Transaction on information theory*, vol. 22, no. 6, pp. 644–654, 1976.

[4] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[5] R. Singh and S. Kumar, "Elgamals algorithm in cryptography," *International Journal of Scientific & Engineering Research*, vol. 3, no. 12, pp. 1–4, 2012.

[6] P. Lee and C. Lim, "Method for exponentiation in a public-key cryptosystem," Dec. 7 1999, uS Patent 5,999,627.

[7] K. S. McCurley, "The discrete logarithm problem, cryptography and computational number theory (c. pomerance, ed.)," in *Proceedings of Symposia in Applied Mathematics*, vol. 42, p. 4974.

[8] N. Koblitz, *Towards a quarter-century of public key cryptography.* Springer, 2000.

[9] P. Dehornoy, "Braid-based cryptography," *Contemp. Math*, vol. 360, pp. 5–33, 2004.

[10] J. J. Climent, P. R. Navarro, and L. Tortosa, "Key exchange protocols over noncommutative rings. the case of," *International Journal of Computer Mathematics*, vol. 89, no. 13-14, pp. 1753–1763, 2012.

[11] B. Tsaban, "Polynomial-time solutions of computational problems in noncommutative-algebraic cryptography," *Journal of Cryptology*, vol. 28, no. 3, pp. 601–622, 2015.

[12] J. Gryak and D. Kahrobaei, "The status of polycyclic group-based cryptography: A survey and open problems," *Groups Complexity Cryptology*, vol. 8, no. 2, pp. 171–186, 2016.

[13] G. Kumar and H. Saini, "Novel noncommutative cryptography scheme using extra special group," *Security and Communication Networks*, vol. 2017, 2017.

[14] M. S. Iqbal, S. Singh, and A. Jaiswal, "Symmetric key cryptography: Technological developments in the field," *International Journal of Computer Applications*, vol. 117, no. 15, 2015.

[15] W. G. Barker, *Introduction to the analysis of the Data Encryption Standard (DES).* Aegean Park Press, 1991.

[16] W. Stallings, *Cryptography and Network Security, 4/E.* Pearson Education India, 2006.

[17] M. A. Musa, E. F. Schaefer, and S. Wedig, "A simplified advance encryption standard algorithm and its linear and differential cryptanalyses," *Cryptologia*, vol. 27, no. 2, pp. 148–177, 2003.

[18] J. J. Rotman, *A first course in abstract algebra.* Pearson College Division, 2000.

[19] P. M. Cohn, *Basic algebra: groups, rings and fields.* Springer Science & Business Media, 2012.

[20] T. Satoh and K. Araki, "On construction of signature scheme over a certain non-commutative ring," *IEICE transactions on fundamentals of electronics, communications and computer sciences*, vol. 80, no. 1, pp. 40–45, 1997.

[21] C. J. Monico, "Semirings and semigroup actions in public-key cryptography," Ph.D. dissertation, University of Notre Dame Notre Dame, 2002.

[22] C. Reutenauer and H. Straubing, "Inversion of matrices over a commutative semiring," *Journal of Algebra*, vol. 88, no. 2, pp. 350–360, 1984.

[23] C. J. Benvenuto, "Galois field in cryptography," *University of Washington*, vol. 1, no. 1, pp. 1–11, 2012.

[24] S. W. Golomb, "Combinatorial proof of fermats little theorem," *The American Mathematical Monthly*, vol. 63, no. 10, p. 718, 1956.

[25] K. Ono, *The Web of Modularity: Arithmetic of the Coefficients of Modular Forms and q-series: Arithmetic of the Coefficients of Modular Forms and Q-series.* American Mathematical Soc., 2004, no. 102.

[26] J. Kerl, "Computation in finite fields," *Arizona State University and Lockheed Martin Corporation*, vol. 1, no. 1, pp. 1–84, 2004.

[27] P. Jovanovic and M. Kreuzer, "Algebraic attacks using sat-solvers," *Groups–Complexity–Cryptology*, vol. 2, no. 2, pp. 247–259, 2010.

[28] Z. Zhang, H. Wang, A. V. Vasilakos, and H. Fang, "Ecg-cryptography and authentication in body area networks," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 6, pp. 1070–1078, 2012.

[29] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in *international conference on the theory and applications of cryptographic techniques.* Springer, 1998, pp. 127–144.

[30] K. H. Ko, S. J. Lee, J. H. Cheon, J. W. Han, J. S. Kang, and C. Park, "New public-key cryptosystem using braid groups," in *annual international cryptology conference.* Springer, 2000, pp. 166–183.

[31] H. Krawczyk, R. Canetti, and M. Bellare, "Hmac: Keyed-hashing for message authentication," 1997.

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

[33] S. Gueron, "Speeding up sha-1, sha-256 and sha-512 on the 2nd generation intel® core processors," in *2012 Ninth International Conference on Information Technology-New Generations.* IEEE, 2012, pp. 824–826.

[34] S. Gueron, S. Johnson, and J. Walker, "Sha-512/256," in *2011 Eighth International Conference on Information Technology: New Generations.* IEEE, 2011, pp. 354–358.

[35] R. L. Rivest, B. Agre, D. V. Bailey, C. Crutchfield, Y. Dodis, K. E. Fleming, A. Khan, J. Krishnamurthy, Y. Lin, L. Reyzin *et al.*, "The md6 hash function–a proposal to nist for sha-3," *Submission to NIST*, vol. 2, no. 3, pp. 1–234, 2008.

[36] R. C. Merkle, "One way hash functions and des," in *Conference on the Theory and Application of Cryptology.* Springer, 1989, pp. 428–446.

[37] S. Altmann and P. Herzig, *Point-group theory tables.* Oxford, 1994.

[38] P. Hegarty, "The absolute centre of a group," *Journal of Algebra*, vol. 169, no. 3, pp. 929–935, 1994.

[39] W. M. Kantor, "Sylow's theorem in polynomial time," *Journal of Computer and System Sciences*, vol. 30, no. 3, pp. 359–394, 1985.

[40] B. C. Hall, "Lie groups, lie algebras, and representations, volume 222 of graduate texts in mathematics," 2003.

[41] Z. Cao, X. Dong, and L. Wang, "New public key cryptosystems using polynomials over non-commutative rings." *IACR Cryptology ePrint Archive*, vol. 2007, p. 9, 2007.