CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Cryptanalysis and Modification of Certificateless Blind Signature Scheme using Elliptic Curve Cryptography

by

Fazeela Karim

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

in the

Faculty of Computing
Department of Mathematics

2021

To my parents, teachers and friends for their support and love.

CERTIFICATE OF APPROVAL

Cryptanalysis and Modification of Certificateless Blind Signature Scheme using Elliptic Curve Cryptography

by

Fazeela Karim

(MMT181022)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Nasir Siddiqui	UET Taxila
(b)	Internal Examiner	Dr. Qamar Mahmood	CUST Islamabad
(c)	Supervisor	Dr. Rashid Ali	CUST Islamabad

Dr. Rashid Ali
Thesis Supervisor
April, 2021

Dr. Muhammad Sagheer
Head
Dept. of Mathematics
April, 2021

Dr. Muhammad Abdul Qadir
Dean
Faculty of Computing
April, 2021

Author's Declaration

I, **Fazeela Karim** hereby state that my M.Phil thesis titled "**Cryptanalysis and Modification of Certificateless Blind Signature Scheme using Elliptic Curve Cryptography**" 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.

(Fazeela Karim)
Registration No: MMT181022

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "**Cryptanalysis and Modification of Certificateless Blind Signature Scheme using Elliptic Curve Cryptography**" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly 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.

(Fazeela Karim)
Registration No: MMT181022

Acknowledgement

All praise be to **Almighty ALLAH** who has been bestowed me with his great bounties, gifted me a loving family and excellent teachers and enabled me to complete my dissertation.

I would like to express my special gratitude to my kind supervisor **Dr. Rashid Ali** for his constant motivation. He was always there whenever I found any problem. I really appreciate his efforts and guidance throughout my thesis and proud to be a student of such kind supervisor.

Also, many thanks are due to all teachers of CUST Islamabad, **Dr. Muhammad Sagheer**, **Dr. Abdul Rehman Kashif**, **Dr. Shafqat Hussain**, **Dr. Muhammad Afzal**, **Dr. Dur-e-Shehwar** and **Dr. Samina Batul** for conveying the excellent lectures.

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

I am thankful to all of my family members for becoming a bulwark of patience and support during my research work. However, my deepest gratitude is reserved for my Parents for their earnest prayers, unconditional love and unflinching support in completing my degree program. They supported and encourage me throughout my life.

I would like to show my gratitude to my seniors specially, **Sir Tahir** for guidance. Especially, I would like to thanks **Sir Zia Malik** for their continuous support and patience during my research work. Also, I would like to thanks all of my friends for motivating me during my degree program.

Finally, I am obliged to all people who pray for me, share their knowledge during my degree program and support me.

(Fazeela Karim)

Abstract

Blind signatures provide authenticity, integrity, nonrepudiation, blindness and untraceability of the message contents. In blind signature, the signer signs the message without revealing the contents of a message. Certificateless blind signature scheme is the combination of certificateless scheme and blind signature scheme. In certificateless blind signature scheme there is no need of certificates to verify the authenticity of the public key of user. In this research, the "Certificatrless blind signature scheme using ECC" of Nayak et al is analyzed. The analysis shows that the proposed certificateless blind signature scheme is not secure against the known cryptanalysis attacks. In fact, an attacker can easily reveal the secret key of the signer by mounting a forgery attack and can become a fake signer. After a successful cryptanalysis, a modified version of the scheme is presented. In the modified version of the scheme, the elliptic curve encryption and decryption is introduced in the signing phase making it secure against the forgery attack. The analysis of the modified scheme shows that the attack that is implemented in the original scheme cannot work in the modified scheme *i.e.* forgery attack. At the end, the application of the modified scheme is presented that shows how modified scheme works.

Contents

List of Figures

List of Tables

Abbreviations

AES	Advanced Encryption Standard
CA	Certificate Authority
CLB	Certificateless Blind Signature Scheme
CLB-ECC	Certificateless Blind Signature Scheme using ECC
CL-PKC	Certificateless Public Key Cryptography
DES	Data Encryption Standard
2DES	Double Data Encryption Standard
DLP	Discrete Logarithm Problem
DSA	Digital Signature Algorithm
ECC	Elliptic Curve Cryptography
ECDLP	Elliptic Curve Discrete Logarithm Problem
GCD	Greatest Common Divisor
IFP	Integer Factorization Problem
PKG	Public Key Generator
RSA	Rivest Shamir Adleman

Symbols

M	Plaintext Space
C	Ciphertext space
E	Encryption Algorithm
D	Decryption Algorithm
K	Secret key
$\mathbb{G}$	Group
$\mathbb{Z}$	Set of integers
$\mathbb{Q}$	Set of rational numbers
$\mathbb{R}$	Set of real numbers
$\mathbb{F}$	Field
H	Hash Function
$\mathbb{Z}_p$	Set of integer modulo p
$\mathbb{GF}\ (p)$	Finite field of order p

Chapter 1

Introduction

From ancient time to today, the secure communication of data over the public network is a big issue. In history there are lots of examples where an individual wants to maintain a data secure from enemies. Many cryptographic techniques were used to stop the enemy from getting important military information during the communication of kings and generals with their soldiers. Because of the rapid growth in science and technology, the requirement for advanced methods for secure data transmission has increased. The requirement for information and electronic services is also increasing day by day. In our life the security of information and electronic systems is very important. The methods required for security of information belong to cryptography [1].

In 1900 BC, Egyptian first time used the concept of cryptography [2]. It was used in various forms and techniques in Egyptian civilization. Egyptian scribes used hieroglyphic symbols to conceal the secret messages from those who did not know it. Hieroglyphic symbols are the picture symbols which are written in a very complicated way. Some of the symbols represent sounds, objects, actions or ideas. These are the most famous Egyptian scripts used for securing government and military information. Later on Roman emperor Julius Caesar introduced the cipher which is known as the Caesar cipher [3] due to his name. After that for transferring codes or secret messages many cryptographic ciphers were developed.

These include, polyalphabetical cipher [4], play-fair cipher [5], mono alphabetical cipher [5], hill ciphers of many orders, etc.

1.1 Cryptography

Cryptography is a branch of cryptology (a science which deals with hidden and encrypted communication) that protect information from unauthorized access. It is used to change the original message into a form which is unreadable in the presence of third party known as an adversary. In cryptography, the original message is known as plaintext and the coded message (meaningless) is called ciphertext. Encryption is a method in which plaintext message is converted into ciphertext whereas the decryption is the process which converts the ciphertext into the corresponding plaintext.

In cryptography, the main focus is on the creation of a strong cryptosystem, so that no one can interfere and change the message between the two parties. Cryptosytem consists of five tuples named as plaintext space M, ciphertext space C, encryption algorithm E, decryption algorithm D and key K. Both sender and receiver uses a secret information for encryption and decryption algorithm and this secret information is known as key [6].

Cryptography is further divided into two main branches [7] the symmetric key cryptography and asymmetric key cryptography. When both the sender and receiver use a single key for encryption and decryption algorithm, then it is called symmetric key cryptography [8]. Only the sender and receiver have access to this single key. The DES (Data Encryption Standard) [9] and AES (Advanced Encryption Standard) [10] are examples of symmetric key cryptography. Key distribution is the main drawback of symmetric key cryptography [11]. In 1976, Diffie and Hellman [12] proposed a Public key cryptography or Asymmetric key cryptography to resolve this problem.

In public key cryptography, both the sender and receiver have two different communication keys, one is known as public key that is known to everybody and the other is known as private key that is kept secret. Examples of Asymmetric key

cryptography are RSA cryptosystem [13], ElGamal cryptosystem [14] and Elliptic curve cryptosystem (ECC) [15].

In cryptography, we not only encrypt and decrypt, but we also solve the real life problems in which the security of information is needed such as confidentiality, data integrity, authentication and non-repudiation (see section 2.1 for details).

1.2 Cryptanalysis

Cryptanalysis is another branch of cryptology [16]. Breaking a cryptosystem is an art known as cryptanalysis. When there is some weakness in the cryptosystem then cryptanalyst can perform cryptanalysis. Cryptanalyst is a person who does this job [17]. Many scientists put their own contribution to the field of cryptanalysis. Many cryptographic attacks were developed that are:

- **Brute force attack** [18], in this attack model attacker tries all possible keys to retrieve original data from the ciphertext.
- **Ciphertext only attack** in this attack, the attacker has only the knowledge of ciphertext and the encryption algorithm. He uses ciphertext to obtain plaintext and the secret key.
- **Chosen ciphertext attack** [19], attacker tries to unveil the secret key with random ciphertext.
- **Chosen plaintext attack** [20], like chosen ciphertext attack in this attack model attacker chooses random plaintext and gets it related ciphertext.
- **Known plaintext attack** [21], secret key is retrieved with the information of plaintext and corresponding ciphertext.
- **Algebraic attack** [22], in this attack, attacker uses his information in algebraic expression and break the scheme to reveal the secret key.
- **Man in the middle attack** [23], in this attack model when two parties are communicating with each other attacker insert himself in the communication and change whole the communication.

1.3 Digital Signature

Digital signature is an electronic signature that can be used to authenticate the message sender's identity. Its basic purpose is to ensure that the content of the original message remains the same. The idea of the digital signature was first presented by Diffie and Hellman in 1976 in their historical paper "New Direction in Cryptography" [12]. In their proposed scheme every single entity has their own public key and a private key. The digital signature is generated by using the private key of the sender and its verification is done by using sender's public key.

On the basis of digital signatures the blind signatures were developed which allows a user to obtain a valid signature without disclosing the content of the message to the signer [24]. For the privacy of the sender, initially a blind signature was presented by David Chaum [25] in 1983. He also proposed a first blind signature scheme at that time that depends on the RSA algorithm. After that many blind signatures scheme are proposed in literature (see [26, 27] for more details). In blind signature the sender of a message and signing authorities are two different parties. Blind signature schemes have two main properties blindness and untraceability. So that these schemes are widely applied where the privacy of the sender is needed. Many blind signature schemes are presented in literature that depends on Integer Factorization Problem (IFP), Discrete logarithm Problem (DLP) and Elliptic Curve Discrete Logarithm Problem (ECDLP) (see Section 2.3 for more details).

To resolve the certificate management problem (*i.e.* storing and managing expensive certificates for a user) of Traditional public key cryptosystem and key escrow problem (*i.e.* is public key generator knows the user's private key) of identity based public key cryptosystem (ID-PKC) a new cryptosystem "Certificateless Public Key Cryptosystem" was firstly presented by Alriyami and Paterson [28] in 2003. In this cryptosystem there is no need of certificates to guarantee the authenticity of the user's public key and it avoids the key escrow problem.

On the basis of a blind signature scheme and certificateless signature scheme a new cryptographic scheme "certificateless blind signature scheme" was proposed

by Nayak et al. [29]. In this scheme, the requester will obtain a certificateless signatures without disclosing the whole message to the signer.

The significance of blind signatures without certificates is growing day by day. Message protection is becoming a big problem when transmitting over a public network. To protect messages from hackers certificateless blind signature is one of the best technique. Due to this reason, certificateless blind signature schemes are widely used in electronic cash system, electronic voting system and in electronic shopping.

1.4 Literature Survey

From the time of creation Certificateless Public Key Cryptography (CL-PKC) and blind signatures find success. Because of the various uses of the certificateless public key cryptography several articles in the field of cryptography have been suggested.

The main purpose of these articles is to improve the security level and overcome the need of certificates to authenticate the users.

Alriyami and Paterson [28] first time introduced the idea of certificateless public key cryptography in 2003. It avoids the inherent key escrow problem of identity based cryptography. In CL-PKC there is no need of certificates for the verification of the authenticity of public keys. They also presented a first certificateless public key encryption scheme. This scheme depends on bilinear maps. The security of this scheme depends upon the hardness of generalized bilinear Diffie-Hellman problem(GBDHP).

Because of the many benefits of the certificateless public key cryptography, a new efficient certificateless pairing-based signature scheme was presented by Zhang and Zhang [24] in 2008. The main advantage of this scheme is that it reduces the computational cost [24]. They also presented a certificateless blind signature scheme which depends upon the certificateless pairing based scheme. The security of these schemes is due to the difficulty of computational Diffie-Hellman (CDH) problem.

Since from the development of CL-PKC and blind signatures, both have found

great success. Zhang and Gao [30] presented an efficient provable certificateless blind signature scheme which is based on pairings. Its security depends upon the difficulty of solving the computational Diffie-Hellman problem and the bilinear pairing inversion problem (BPI). The scheme is more efficient because in the process of signing it does not need pairing based operations. This scheme is more effective in terms of computational cost and the size of signatures. In this scheme, no cost is associated with the signature based on pairing. As compared to other proposed blind signature schemes this CLBS scheme is more efficient.

Jose et al. [31] presented a new, efficient provably secure certificateless blind signature scheme. Its security depends on solving computational Diffie-Hellman (CDH) and chosen-target CDH problem. It is the first certificateless blind signature scheme which is strongly unforgeable and satisfies blindness property.

In 2015, Kumar [32] presented a certificateless blind signature scheme. The proposed scheme fulfils overall security criteria of certificateless signatures and blind signatures. It provides the security attributes including blindness and unforgeability. When we compare this scheme with other existing schemes [30], this scheme has less computational cost [32] as shown in Table 1.1. Its security depends upon the difficulty of solving the discrete logarithm problem (DLP).

Schemes	Signing Cost	Verifying Cost	Total Cost
Zhang and Zhang	$102(ms)$	$53(ms)$	$155(ms)$
Zhang and Gao	$66(ms)$	$34(ms)$	$100(ms)$
Kumar	$79(ms)$	$0.18(ms)$	$79.18(ms)$

TABLE 1.1: Comparison of certificateless blind signature scheme based on DLP with other schemes

1.5 Current Research

In this thesis, the article "Certificateless Blind Signature Scheme using elliptic curve cryptography(CLB:ECC)" proposed by Nayak et al. [29] is reviewed. They proposed the scheme by using elliptic curve cryptography over a finite field. They

claim that due to smaller key size this scheme is suitable for the wireless communication environment. But our cryptanalysis shows that this Certificateless Blind Signature scheme has security flaws. The current research shows that the main drawback of the proposed scheme is that anyone can become a fake signer after getting a secret key of the signer from the public parameters. To get the secret key we apply the forgery attack on the proposed scheme. To counter the attack, a modified form of the scheme is proposed in this thesis. The security and cost analysis of the modified scheme are also discussed.

1.6 Thesis Layout

Rest of the thesis is composed as follows.

In **Chapter 2**, a detailed explanation of some basic concepts to understand the proposed scheme, as described in Chapter 3 is provided. Furthermore a cryptology and some basic definitions related to blind signatures are described. At the end of this chapter, a comprehensive explanation of elliptic curve cryptography and certificateless blind signature is given.

In **Chapter 3**, the review of "CLB-ECC: Certificateless Blind Signature using ECC" by Nayak et al. [29] is presented. For that purpose various known certificateless blind signature schemes are also discussed. Furthermore, the concept of the certificateless blind signature scheme using ECC with the help of an example is described.

In **Chapter 4**, a cryptanalysis of the "CLB-ECC: Certificateless Blind Signature using ECC" scheme is presented.

In **Chapter 5**, the modified version of the scheme is presented. The modified version of the scheme is more secure because it involves the ECC encryption and decryption in the signing phase. The security analysis of the modified certificateless blind signature scheme is also presented. The chapter is closed by presenting an application of the modified scheme.

Chapter 2

Preliminaries

In this chapter, some basic definitions from cryptography are presented. Furthermore, some basic definitions, notations and results from algebra and cryptography are presented for the reader's convenience. The mathematical background and some hard problems in cryptography are also described in this chapter.

2.1 Cryptology

The kryptos and logos are two Greek words from which the word cryptology is originated. Kryptos means concealed and logos mean words. In 1645, James Howell invented the term cryptology [17]. Cryptology is the science that deals with hidden and encrypted communications. It has two main branches (1). Cryptography and (2). Cryptanalysis shown in Figure 2.1.

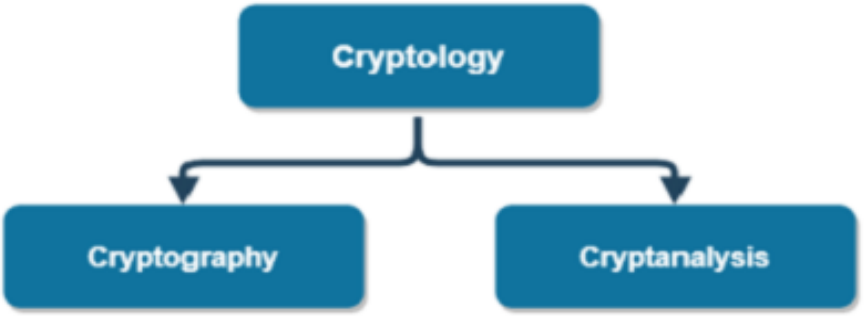

FIGURE 2.1: Types of Cryptology

2.1.1 Cryptography

Thomas Browne a British physician and writer invented the term cryptography. Cryptography originates from two Greek words, kryptos and graphein. Krypto means hidden, and graphein means to write [17].

Cryptography is a technique used to encrypt information in such a way that it can be read or accessed only by authorized participants. The original message is called plaintext, while the coded message is known as ciphertext. In encryption, we change plaintext into ciphertext while in decryption, we change the ciphertext into plaintext. With the help of secret key K the process of encryption and decryption is done as shown in Figure 2.2. A system in which we convert data or message into secret code using encryption algorithm and convert secret codes back into the message using the decryption algorithm is known as cryptosystem. A cryptosystem has five basic elements [6].

- Plaintext Space M
- Ciphertext Space C
- Key K
- Encryption Algorithm E
- Decryption Algorithm D

FIGURE 2.2: Cryptography

Purpose of Cryptography: In cryptography, two parties sender (Alice) and receiver (Bob) have to communicate with each other. The sender transmits the coded message to Bob through the public network. After receiving the encoded

data Bob decodes the encoded message by the decryption algorithm to obtain the original message and vice versa. The security of the whole process is based on key which is kept secret.

The modern cryptography deals with the following security attributes [15].

1. **Confidentiality:** The only authenticated persons know the meaning of original information and no one can understand the transmitted information. If anyone knows the secret information he is able to get the original data.

2. **Data integrity:** The method of data integrity guarantees that the data transmitted from the sender to the recipient is not altered by the intruder. The correctness of the data transmitted is given by this procedure, as this service protects the transmitted data from modification during transfer [33].

3. **Authentication**: It ensures that only authorized person have sent the message [5].

4. **Non-Repudiation**: The sender cannot deny at any stage about the transportation of any information. In any cryptosystem, this property helps the receiver to trust on the sender [34].

2.1.2 Types of Cryptography

Cryptography has two main branches.

- Symmetric Key Cryptography
- Asymmetric Key Cryptography

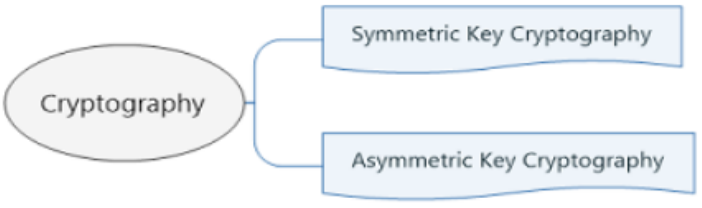

FIGURE 2.3: Types of Cryptography

2.1.2.1 Symmetric Key Cryptography

In symmetric key cryptography (which is also known as private key cryptography), both the sender and receiver while transferring a message use the same key for encryption and decryption. This single key which is kept secret is termed as private key or secret key. It is very fast and simpler [8]. Data Encryption Standard (DES), Double Data Encryption Standard (2DES) [7], Triple Data Encryption Standard (3DES) [7], Advanced Encryption Standard (AES) [10] and Blowfish [35] are the examples of a private key cryptography. The main drawback of symmetric key cryptography is the key sharing, which means that the private (secret) key should transmit to each party involved in communication.

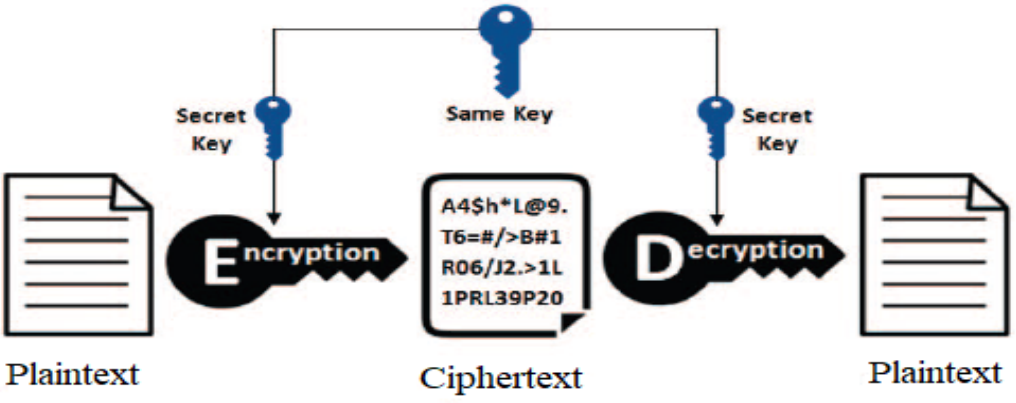

FIGURE 2.4: Symmetric key Cryptography

2.1.2.2 Asymmetric Key Cryptography

To resolve the key exchange issue, Diffie-Hellman in 1976 introduced an asymmetric key cryptography (which is also known as public key cryptography). For encrypting and decrypting data in asymmetric key cryptography, two keys are used. One is known to everybody called public key and the other is kept secret which is known as a private key. The sender of the message used public key of the receiver for the encryption, while receiver used his private key for decryption. A derivation of the private key from the public key is infeasible.
Asymmetric key cryptography is very effective in authentication. Examples include RSA [13], Elgamal [14], DSA [36] , ECC [37].

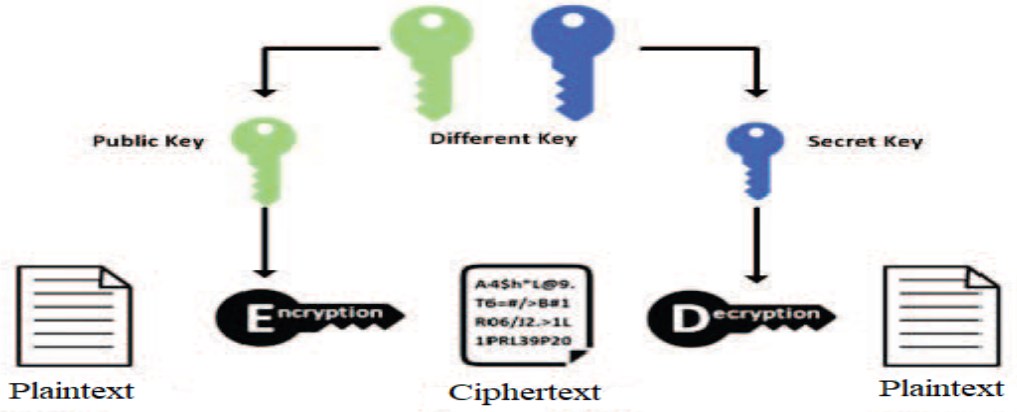

FIGURE 2.5: Asymmetric key Cryptography

2.2 Cryptanalysis

Cryptanalysis is an art of decoding the encrypted message without having the knowledge of secret key used for encryption [5]. Cryptanalyst is a person who does this job. The operation which is used to break down a system or communication is called an attack on that system. The main aim of the attacks is to find secret key K or an original plaintext. These attacks are successful when there is some security weakness in the cryptosystem [38]. A cryptosystem is said to be vulnerable to an attack if any one property from four properties (confidentiality, information integrity, message validation and non-repudiation) are found to be weak. There are two types of attack models exist.

- Active attack
- Passive attack

1. Active attacks:

In this type of attacks, a cryptanalyst interrupts the connection and change the communication. This attack has harmful effects [7] because the content of the message is modify. The integrity feature is compromised by this attack.

2. Passive attacks:

In this type of attacks, the attacker is always in struggle to get or to use information from the system, without affecting the system resources [7]. Known plaintext attack is the example of passive attack.

Cryptanalysis has different types of attacks depending upon above mentioned models. The main difference in these attacks depends on the availability of information which attacker has. Some of these attacks are described below.

2.2.1 Ciphertext only attack

In this attack, the attacker has only the knowledge of ciphertext and the encryption algorithm. He uses ciphertext to obtain plaintext and the secret key. Mostly the attacker has no information about the original data, but he continuously tries to unveil the original data by using ciphertext attack. Frequency analysis is very helpful in ciphertext attack [39].

2.2.2 Known Plaintext Attack

In this type of attack, the attacker has an information regarding plaintext and the corresponding ciphertext. He wants to find the secret key to decode any further information with the help of previous data.

2.2.3 Chosen Plaintext Attack

For attacking a cryptosystem there are many structures, chosen plaintext attack is one of them. The attacker chooses plaintext of his own choice and then obtain its related ciphertext. To obtain more information of the cryptosystem is the main purpose of this attack.

2.2.4 Chosen Ciphertext Attack

In this attack, the cryptanalyst arbitrarily chooses the ciphertext of his own choice and gets a plaintext of it. He struggles to retrieve the secret key from the information. It is also known as "midnight attack" or "lunch break attack" [19]. From all other attacks, it is considered to be a stronger attack on any cryptosystem.

2.2.5 Man-in-the-Middle-Attack

In this attack model, when two parties are communicating with each other, the attacker insert himself in the communication and control the communication between sender and receiver. The attacker first selects two fake keys to execute this attack and then begins the communication with the first participant using his one key. The attacker creates a successful connection with the first participant. In the same way another successful connection is formed with the second participant [23]. Now the attacker sends message of his own choice to both participants. The both participants believe that they are communicating with each other.

2.2.6 Key Only Attack

In this attack, an attacker only has the knowledge of the signer's public key. Using public key he tries to create the signatures and tries to find out the secret key of the signer. For instance, in the case of RSA the hard underlying problem is the integer factorization problem. If the size of the parameters is less than the recommended size, then the attacker may be able to find out the corresponding decryption key. Because if we take key size smaller then attacker can easily find out the totient and through totient and Extended Euclidean Algorithm he can get a decryption key that is if we select two prime integers $p = 5$ and $q = 3$ then $n = p \times q = 5 \times 3 = 15$ and $\phi(n) = 4 \times 2 = 8$. After this he selects the encryption key $e = 7$ such that $gcd(e, \phi(n)) = 1$. Now the attacker can easily find out the decryption key $d < 8$ and $de = 1 \bmod 8$, using these conditions only valid decryption key is $d = 7$ because $d \times e = 7 \times 7 = 49 = 1 \bmod 8$. Hence when decryption key is retrieved, then signatures can be easily created.

2.2.7 Forgery Attack

In this attack an adversary Eve attempt to forge a blind signature for the message without knowing the corresponding secret signing key of the signer [40]. The word

forgery generally refer to a message related attack against a cryptographic digital signature (for more details see 4.1).

2.3 Mathematical Background

The purpose of this section is to recall some basic definitions which will be used throughout the thesis.

Definition 2.3.1 (Algorithm)**.**
Algorithm is a finite sequence of well-defined set of instructions arranged to complete specific task. A problem can be solved by following the step by step instructions of an algorithm.

Definition 2.3.2 (Group)**.**
"A group [41] $(\mathbb{G}, *)$ is a set $\mathbb{G}$, closed under a binary operation $*$, such that the following axioms are satisfied:

1. Associativity:
   For all a, b, $c \in \mathbb{G}$, we have

$$(a * b) * c = a * (b * c) \tag{2.1}$$

2. Identity element:
   There is an element e in $\mathbb{G}$ such that for all $x \in \mathbb{G}$,

$$e * x = x * e = x \tag{2.2}$$

   where e is called the identity of $\mathbb{G}$.

3. Existence of inverse:
   Corresponding to each $a \in \mathbb{G}$, there is an element a' in $\mathbb{G}$ such that

$$a * a' = a' * a = e \tag{2.3}$$

where e is the identity element".

Moreover when $f_1 * f_2 = f_2 * f_1 \; \forall \; f_1, \; f_2 \in \mathbb{G}$ then $\mathbb{G}$ is called commutative group.

Example 2.3.3. Some examples of group are described below:

- Set of real numbers $\mathbb{R}$, rational numbers $\mathbb{Q}$, complex numbers $\mathbb{C}$, and integers $\mathbb{Z}$ all becomes group under binary operation of addition.
- The set of all $n \times n$ matrices , with real coordinates denoted by $M_2(\mathbb{R})$ become a group under matrix addition.
- Set of integers $\mathbb{Z}$ under multiplication is not a group.

Definition 2.3.4 (Field)**.**
"Let $\mathbb{F}$ [42] be a set with two binary operations $+$ and $.$ with respective identity elements 0 and 1, where $0 \neq 1$. Then $\mathbb{F}$ is called a field if.

1. the set of all elements of $\mathbb{F}$ is an abelian group under $+$;
2. the set of all nonzero elements of $\mathbb{F}$ is an abelian group under $.$;
3. $a(b + c) = ab + ac$ and $(a + b)c = ac + bc$ for all $a, b, c \in \mathbb{F}$"

Example 2.3.5. Some examples of field are as follows.

- For any prime number p, $\mathbb{Z}_p$ is a field. Where $\mathbb{Z}_p = \{1, 2, 3, ..., p - 1\}$
- Set of complex numbers $\mathbb{C}$ and real numbers $\mathbb{R}$ are fields under usual addition and multiplication.
- Set of integers $\mathbb{Z}$ is not a field under binary operation of multiplication because there are no multiplicative inverses in $\mathbb{Z}$.

Definition 2.3.6 (Finite field)**.**
"A field that contains a finite number of elements is called a finite field [43]".

Example 2.3.7. Some examples of finite field are as follows.

- Galois field is an example of finite field.
- $\mathbb{Z}_3 = \{ 0, 1, 2 \}$ is an example of finite field.

Explanation:
For x,$y \in \mathbb{Z}_3$, $x + y$ will be equal to the remainder which is left after dividing the usual sum of x and y by 3. It means that $1 + 2 = 3$ will be equal to 0 in $\mathbb{Z}_3$.

+	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

TABLE 2.1: Addition in $\mathbb{Z}_3$

And for x,$y \in \mathbb{Z}_3$, $x \times y$ will be equal to the remainder left after dividing the usual multiplication of x and y by 3. That is $2 \times 2 = 4$ will be equal to 1 in $\mathbb{Z}_3$.

×	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

TABLE 2.2: Multiplication in $\mathbb{Z}_3$

Every non-zero element of finite field $\mathbb{Z}_p$ has a multiplicative inverse. To find inverses in $\mathbb{Z}_p$ we use extended Euclidean algorithm.

Definition 2.3.8 (Modular Inverses)**.**
For any two given integers a and b to find an integer c such that $a.c \equiv 1 \bmod b$ and $a^{-1} \equiv c \bmod b$, where $1 \leq c \leq b - 1$. The multiplicative inverse of a in mod b is c if a is relatively co-prime to b that is $gcd(a, b) = 1$.

Definition 2.3.9 (Extended Euclidean algorithm)**.**
To find inverses in $\mathbb{Z}_p$ we use extended Euclidean algorithm [7].
Following is the process for finding inverse of a mod b.
Input: Two integers a and b
Output: a^{-1} mod b

1. Initialize six integers U_i and W_i for $i = 1, 2, 3$
   $(U_1, U_2, U_3) = (1, 0, b)$
   $(W_1, W_2, W_3) = (0, 1, a)$
2. If $W_3 = 0$, return $U_3 = gcd(a, b)$, no inverse of a exist in mod b.
3. If $W_3 = 1$, return $W_3 = gcd(a, b)$ and $W_2 = a^{-1} \bmod b$
4. Now divide U_3 by W_3 and find the quotient Q
5. $(T_1, T_2, T_3) = ((U_1 - QW_1), (U_2 - QW_2), (U_3 - QW_3))$
6. $(U_1, U_2, U_3) = (W_1, W_2, W_3)$
7. $(W_1, W_2, W_3) = (T_1, T_2, T_3)$
8. Go to step 2

Definition 2.3.10 (Galois Field)**.**
"For every prime p and positive integer n, there is exactly one finite field of order p^n. This field $GF(p^n)$ usually referred as Galois field of order p^n." Evaiste Galois is the french Mathematician who invent Galois field in 1830 [44].

Definition 2.3.11 (Integer Factorization Problem)**.**
Let n be a number, the problem of decomposition of n to the product of primes s_1 and s_2 such that $n = s_1\ s_2$ is called integer factorization problem.
The security of well known RSA cryptosystem [13] relies on the difficulty of IFP.

Definition 2.3.12 (Discrete Logarithm Problem)**.**
The process of finding an unknown n, when A, k and p are known is called a discrete logarithm problem [33] from the following equation.

$$A = k^n \bmod p \tag{2.4}$$

2.4 Elliptic Curve Cryptography(ECC)

Elliptic curve cryptography (ECC) is created on the basis of elliptic curves. ECC was first introduced by Miller and Koblitz in 1985. It is a public key cryptosystem

that derived basically from the algebraic construction of elliptic curves over finite fields [33]. A finite elliptic curve E can be described through the cubic equation of the form,

$$y^2 = (x^3 + ux + v) \bmod p \tag{2.5}$$

Where u and v are constants and are the elements of a finite field $\mathbb{F}$. This kind of equation is also called Weierstrass equation. From a cryptographic point of view the elliptic curve must require the non singularity and it must satisfy the following equation.

$$(4u^3 + 27v^2) \bmod p \neq 0 \tag{2.6}$$

In the above equation, u and v are constants and elements of Finite field $\mathbb{F}$. Consider an elliptic curve

$$y^2 = (x^3 + x + 6) \bmod 11 \tag{2.7}$$

In above equation, $u = 1$, $v = 6$ and $p = 11$. We first verify that:

$$\begin{aligned} 4u^3 + 27v^2 \bmod p &= 4(1)^3 + 27(6)^2 \bmod 11 \\ &= 4 + 27 \times 36 \bmod 11 \\ &= 976 \bmod 11 \\ &= 8 \bmod 11 \neq 0 \end{aligned}$$

Now we will find the quadratic residues Q_{11} from the reduced set of residues $\mathbb{Z}_{11}$ $= \{0, 1, 2, ..., 10\}$

$x^2 \bmod p$	$(p-1)^2 \bmod p$	$=$
$1^2 \bmod 11$	$10^2 \bmod 11$	1
$2^2 \bmod 11$	$9^2 \bmod 11$	4
$3^2 \bmod 11$	$8^2 \bmod 11$	9
$4^2 \bmod 11$	$7^2 \bmod 11$	5
$5^2 \bmod 11$	$6^2 \bmod 11$	3

TABLE 2.3: Quadratic residues in $\mathbb{Q}_{11}$

Set of quadratic residues are $\dfrac{p-1}{2} = 5$

$\Rightarrow Q_{11} = \{1, 3, 4, 5, 9\}$

To find the points on elliptic curve, we will compute $y^2 = x^3 + x + 6 \bmod 11$ for $0 \leq x < 11$ and check whether the y^2 lies in the set of quadratic residues.

x	0	1	2	3	4	5	6	7	8	9	10
y^2	1	3	11	8	0	16	16	6	15	3	22
$y^2 \in Q_{11}$?	No	No	Yes	Yes	No	Yes	No	Yes	Yes	No	Yes
y_1			4	5		2		2	3		2
y_2			7	6		9		9	8		9

The elliptic group $E_{11}(1,6)$ includes the following points.

$$E_{11}(1,6) = \{(0,0),\ (2,4),\ (2,7),\ (3,5),\ (3,6),\ (5,2),\ (5,9)$$
$$(7,2),\ (7,9),\ (8,3),\ (8,8),\ (10,2),\ (10,9)\}$$

ECC is a public key cryptography. The keys that are developed by ECC are smaller in size as compared to RSA and Elgamal. In comparison to other cryptosystems, ECC obtains the same security level with smaller key size. It uses 160 bit key size, as compared to 1024 bit RSA [45]. That is why, ECC makes the cryptosystem secure with less computational cost. ECC is faster than RSA due to shorter key size and take less storage [46]. Over this characteristic, ECC is widely used in encryption and decryption method contrast to other cryptographic approaches. It is observed that the ECDLP is exponentially very tough to solve.
The set of all points satisfying equation (2.5) form a finite group $E_p(u,v)$ known as Elliptic Curve. The main operations performed on elliptic curves are as follows.

- Point addition
- Point multiplication

2.4.1 Point Addition

In point addition, the new point is obtained on a curve when we add two points that lie on an elliptic curve. On an elliptic curve, for the addition of two points P and Q we have to follow the following defined rules.

1. $P_1 + \mathcal{O} = \mathcal{O} + P_1 = P_1$, where $\mathcal{O}$ is the point at infinity.
2. If $t_2 = t_1$ and $z_2 = -z_1$, that is $P_1 = (t_1, z_1)$ and $P_2 = (t_2, z_2) = (t_1, -z_1) = -P_1$ then $P_1 + P_2 = \mathcal{O}$.

3. If $P_2 \neq -P_1$:

   Then $P_1 + P_2 = (t_3, z_3)$ is given by:

   $t_3 = \lambda^2 - t_1 - t_2 \bmod p$

   $z_3 = \lambda(t_1 - t_3) - z_1 \bmod p$

where

$$\lambda = \begin{cases} (z_2 - z_1)(t_2 - t_1)^{-1} \bmod p & \text{if } P_1 \neq P_2 \\ (3t_1^2 + a)(2z_1)^{-1} \bmod p & \text{if } P_1 = P_2. \end{cases}$$

Example 2.4.1. Sum of two points $P_1 = (5, 2)$ and $P_2 = (7, 9)$ is $(3, 5)$ on an elliptic curve $y^2 = x^3 + x + 6 \bmod 11$.

Explanation:

To add two different points on elliptic curve first we will find λ.

$$\begin{aligned} \lambda &= (z_2 - z_1)(t_2 - t_1)^{-1} \bmod p \\ &= (9 - 2)(7 - 5)^{-1} \bmod 11 \\ &= (7)(2)^{-1} \bmod 11 \\ &= (7)(6) \bmod 11 \\ &= 42 \bmod 11 \\ &= 9 \bmod 11 \end{aligned}$$

Now we find sum of P and Q by using:

$$\begin{aligned} x_3 &= \lambda^2 - t_1 - t_2 \bmod p \\ &= 9^2 - 5 - 7 \bmod 11 \\ &= 81 - 5 - 7 \bmod 11 \\ &= 69 \bmod 11 \\ &= 3 \bmod 11 \end{aligned}$$

and

$$\begin{aligned} y_3 &= \lambda(t_1 - t_3) - z_1 \bmod p \\ &= 9(5 - 3) - 2 \bmod 11 \end{aligned}$$

$$\begin{aligned} y_3 &= 9(5-3) - 2 \bmod 11 \\ &= 9(2) - 2 \bmod 11 \\ &= 18 - 2 \bmod 11 \\ &= 16 \bmod 11 \\ &= 5 \bmod 11 \end{aligned}$$

Hence sum of P_1 and P_2 is $(3, 5) \in E_{11}(1, 6)$.

Steps for adding two points graphically:

The idea of point addition graphically is came from real elliptic curve. Let us consider two points P and Q on an elliptic curve. To add two points graphically on an elliptic curve, the following steps are used to add two distinct points.

1. By using the points P and Q, draw a straight line.
2. The intersection of a straight line and elliptic curve, we get a point R.
3. Take reflection (negative) of R.
4. This reflection of R is our desired addition of two distinct points.

Example 2.4.2. Let us consider the two points $P = (2, 4)$ and $Q = (-1, 2)$ on the real elliptic curve $y^2 = x^3 + x + 6$. Addition of P and Q is done graphically by first drawing the straight line through P and Q that intersected the curve at point R as shown in figure. Then take reflection of R, denoted by R' that is the additive inverse of the point R and it is obtained by translating the y- coordinate.

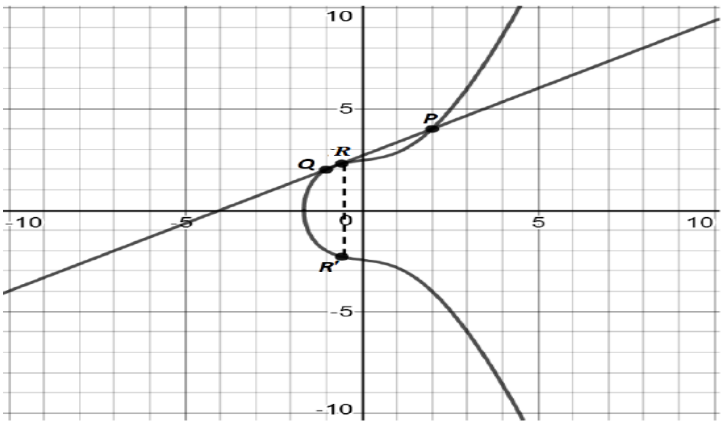

FIGURE 2.6: Point Addition

2.4.2 Point Multiplication

The multiplication of a point with a scalar n on an elliptic curve is defined as

$$nP = \overbrace{P + P + ... + P}^{(n \quad \text{times})}; \text{ where } n = 1, 2, 3, ... \tag{2.8}$$

Example 2.4.3. For $P_1 = (5, 2)$ then $2P_1$ can be computed by using 2.8 as follows.

Explanation:

To multiply point on elliptic curve first we will find λ.

$$\begin{aligned}
\lambda &= (3t_1^2 + a)(2z_1)^{-1} \bmod p \\
&= (3 \times 5^2 + 1)(2 \times 2)^{-1} \bmod 11 \\
&= (3 \times 25 + 1)(4)^{-1} \bmod 11 \\
&= (75 + 1) \times 3 \bmod 11 \\
&= 76 \times 3 \bmod 11 \\
&= 228 \bmod 11 \\
&= 8 \bmod 11
\end{aligned}$$

Now we find $2P = (x_3, y_3)$ by using

$$\begin{aligned}
x_3 &= \lambda^2 - t_1 - t_2 \bmod p \\
&= 8^2 - 5 - 5 \bmod 11 \\
&= 64 - 10 \bmod 11 \\
&= 54 \bmod 11 \\
&= 10 \bmod 11
\end{aligned}$$

and

$$\begin{aligned}
y_3 &= \lambda(t_1 - t_3) - z_1 \bmod p \\
&= 8(5 - 10) - 2 \bmod 11 \\
&= 8(-5) - 2 \bmod 11 \\
&= -40 - 2 \bmod 11
\end{aligned}$$

$$\begin{aligned} y_3 &= -40 - 2 \bmod 11 \\ &= -42 \bmod 11 \\ &= 2 \bmod 11 \end{aligned}$$

Hence $2P_1$ is $(10, 2) \in E_{11}(1, 6)$.

Steps for point multiplication on graph:

The multiplication of a point with a scalar on an elliptic curve is obtained by adding point to itself, it is also known as point doubling. It is done graphically by using following steps.

Step 1: Draw a tangent line through the point P.

Step 2: At a point R, this tangent line intersects on elliptic curve.

Step 3: Find vertically opposite point of R, called reflection of R.

Step 4: This reflection of R is our desired point and denote it by R'.

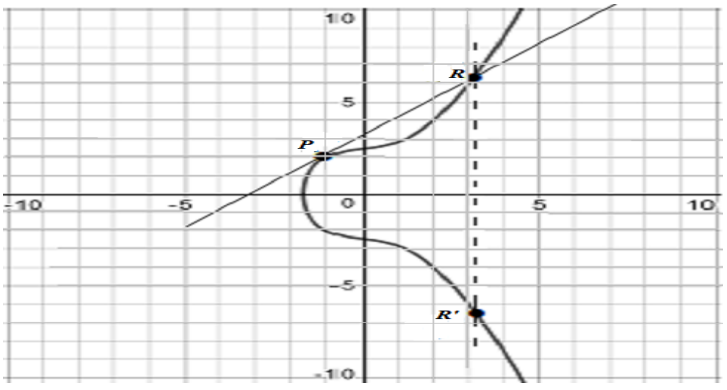

FIGURE 2.7: Point Multiplication

Definition 2.4.4 (Elliptic Curve Discrete Logarithm Problem)**.**

Elliptic curve discrete logarithm problem [29] is the problem of finding an integer q when P, $R \in E(F_p)$ are given in the following equation.

$$P = qR \tag{2.9}$$

Security of ECC relies on the ECDLP. ECC (Elliptic Curve Cryptography) can be implemented in different methods, it is more complex than RSA [46]. As compared

to IFP, ECDLP is harder to break. The researchers have attempted to break ECC, but in the present computational resources it is infeasible. At present, ECC is more secure than other public key cryptosystems [45].

2.5 ECC Encryption and Decryption

ECC is an asymmetric key cryptosystem. This means that for communication using ECC, each participant must have two keys, one is known as public key and the other is called private key [47]. For every participant which involve in the communication requires a set of parameters to be known.

2.5.1 Global Setting:

The domain parameters which are necessary to known by all entities participating in the communication are as follows:

- The generator point G of elliptic curve such that $nG = \mathcal{O}$.
- The constant a and b
- A prime integer modulo p.

2.5.2 Key Generation:

For conversation using ECC between two parties must require keys which are generated by Alice and Bob as follows:

- Alice selects a random number $j_A < p$ as private key and finds public key by multiplying private key with a generator point G that is:

$$J_A = j_A \times G$$

- Bob also selects a random number $j_B < p$ as private key and finds public key by multiplying private key with a generator point G that is:

$$J_B = j_B \times G$$

2.5.3 Encryption:

Alice wants to send a message m to Bob by using elliptic curve encryption. For this he first converts m into a point Q_m on an elliptic curve. After this he selects a random positive integer r and encrypt it as follows:

$$R_m = \{r \times G, Q_m + r \times J_B\}$$

2.5.4 Decryption:

After receiving the encrypted message (ciphertext) Bob decrypts it by multiplying the first point of the ciphertext with private key j_B using point multiplication and then subtract the result from the second point of the ciphertext that is:

$$\begin{aligned} Q_m + r \times J_B - j_B(r \times G) &= Q_m + r(j_B \times G) - j_B(r \times G) \\ &= Q_m \end{aligned}$$

Example 2.5.1. Let us consider an elliptic curve $y^2 = (x^3 + x + 1) \bmod 23$ and generator point $G = (3, 10)$. Alice sends the message m to Bob by using ECC.
Explanation:
To send any message from one entity to another by using elliptic curve encryption. First of all both entities selects their private key and computes public key. Alice chooses $j_A = 2 < 23$ as a private key and creates public key as:

$$\begin{aligned} J_A &= 2(3, 10) \\ &= (7, 12) \end{aligned}$$

Bob selects $j_B = 5 < 23$ as a private key and calculates public key as:

$$\begin{aligned} J_B &= 5(3,10) \\ &= (9,16) \end{aligned}$$

Let message m map on the elliptic curve point $Q_m = (11,3)$. To encrypt this Alice selects a random number $r = 3$ and produces a ciphertext R_m as follows:

$$\begin{aligned} R_m &= \{r \times G, Q_m + r \times J_B\} \\ &= \{3 \times (3,10), (11,3) + 3 \times (9,16)\} \\ &= \{(19,5), (11,3) + (1,16)\} \\ &= \{(19,5), (12,19)\} \end{aligned}$$

and sends $R_m = \{(19,5), (12,19)\}$ to Bob.
After receiving R_m Bob decrypts it to get ciphertext Q_m as follows:

$$\begin{aligned} Q_m &= Q_m + r \times J_B - j_B(r \times G) \\ &= (11,3) + 3(5 \times (3,10)) - 5(3 \times (3,10)) \\ &= (11,3) + 3(9,16) - 5(19,5) \\ &= (11,3) + (1,16) - (1,16) \\ &= (11,3) \end{aligned}$$

2.6 Digital Signature

It is an electronic signature which is used to authenticate the sender's identity on a message [34]. It is a process that ensures that information has not been changed. It is equivalent to a person's written signature. It is a technique that allows the sender of a message to connect a code that act as a signature for authentication. On the basis of digital document and hash function, digital signatures are formed. The idea of digital signature was initially presented in 1976 by Diffie and Hellman

[12]. They only explain the idea of a digital signature scheme, but they did not propose any algorithm. They only proposed that such kind of scheme can be constructed.

In digital signature schemes every entity has their own public and private key in the proposed digital signature scheme. The digital signature is generated with the help of the sender's private key and it is verified by using sender's public keys [48]. Digital signature comprises of two algorithms, namely signature generation and signature verification. A secure digital signature provides a recipient reason to trust that a known sender (authentication) has created the message, that the sender cannot refuse having sent the message (non-repudiation), and that the message was not changed in transferring it. Due to digital signatures, following properties of security are obtained.

1. **Correctness**: The signatures cannot be validated without using the signer's public key.
2. **Authenticity**: This confirms that the message was signed by a right person.
3. **Unforgeability**: This means that for each message, a valid signer can only produce one unique valid signature.
4. **Non-repudiation**: The signer cannot deny having signed the message.
5. **Integrity**: It can tell us that the message did not alter during transmission.
6. **Non-reusability**: The signature that has been used in one message cannot used for signing other messages.

2.6.1 ElGamal digital signature scheme

In 1984, Taher Elgamal [49] invented the Elgamal algorithm that is a digital signature scheme. This scheme is based on asymmetric key cryptosystem. This scheme depends on the hardness of solving discrete logarithm problem (DLP). In this scheme the two participants are involved, the signer and the verifier. Also, this

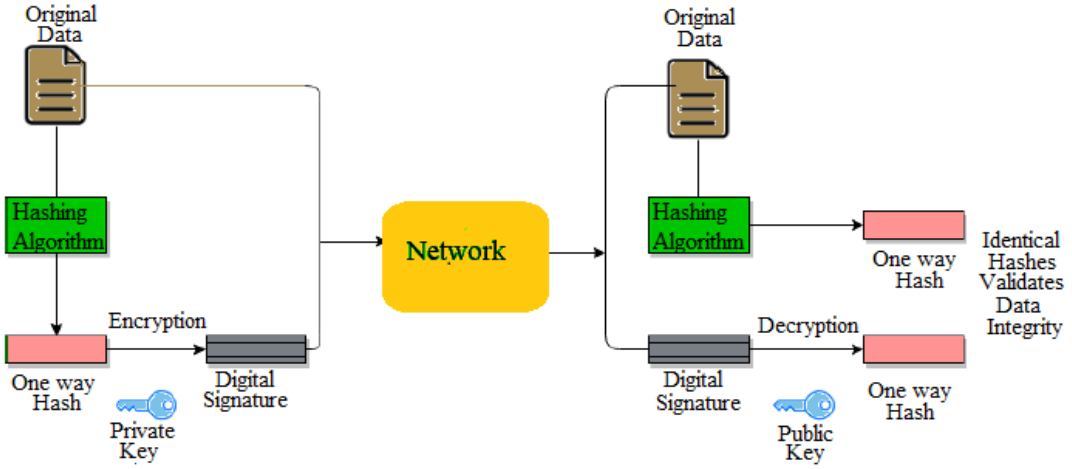

FIGURE 2.8: Digital Signature

scheme comprises of three phases (key generation, signing and verification). To sign a document the signer utilizes his secret key. After signing the document the signer sends this signature to the verifier. After getting the signature, the verifier verify the signature's validity by using public key [50].

Key generation:

1. Signer chooses large prime numbers p and q, where $p \in \mathbb{Z}_p$ and q is the primitive root of p. $\mathbb{Z}_p$ is a Galois field.
2. Signer also chooses secret key $u \in \mathbb{Z}_p$.
3. Calculates the public key as $v = q^u \bmod p$.
4. Publish p, q, and v in public but keep u as secret.

Signature generation:

Signer performs the following steps to generate the signature.

1. Signer randomly selects an integer r. Where $(r < p) \in \mathbb{Z}_p$ and $\gcd(r, p-1) = 1$.
2. Calculates $s = q^r \bmod p$ and $t = r^{-1}(m - us) \bmod (p-1)$. Where m is the message and (s, t) is the signature.
3. Signer sends signature (s, t), m to the verifier.

Verification:

1. After receiving the signatue from signer, requester performs the following step to check the validity of the signature (s, t)

2. Verifier check the validity of received (s, t) by $q^m \equiv v^s s^t \mod p$ with the help of p, q and v.

If it verifies then the signature is valid and the message is authentic otherwise discard it.

2.7 Blind Signature

Blind signature is an electronic signature in which signer singed the message without knowing the message content. David Chaum [25] first time gave the idea of blind signature in 1983. Blind signature enables a person to get a message signed by another party without disclosing any information related to the message to the other party [51]. It is mostly applied where two different parties the signer and the message author are involved. For example, electronic election systems and digital cash systems.

In blind digital signature, three parties are involved, namely a requester (sender), a signer and a verifier [50]. The requester is a person who wants that the signer sign his message. The signer is a person who signs the message received from the requester. Verifier is the one who verifies the signature.

In blind signature scheme all the properties of digital signatures are satisfied with two additional properties of blindness and untraceability [24].

Blindness: It is a signature protocol that allows the sender to transmit a message to the signer and the signer cannot be able to read the content of the original message.

Untraceability: This property confirms that the signer cannot link back any pair of message and signature even if the signature is made public.

Blind signature scheme contains five phases [50]. These phases are as follows.

1. **Initialization:** This phase initializes the system parameters for the signer and the requester.

2. **Blinding:** The requester chooses a blind factor to blind the message and then sends this blinded message to the signer.

3. **Signing:** After getting the blinded message, the signer uses his secret key to sign it and then return the blind signature to the requester.

4. **Unblinding:** When the requester gets the blinded signature, he uses his blind factor to retrieve the digital signature of the signer and sends it to the verifier.

5. **Verifying.** The Verifier verifies the authenticity of the signature by using the public key of the signer.

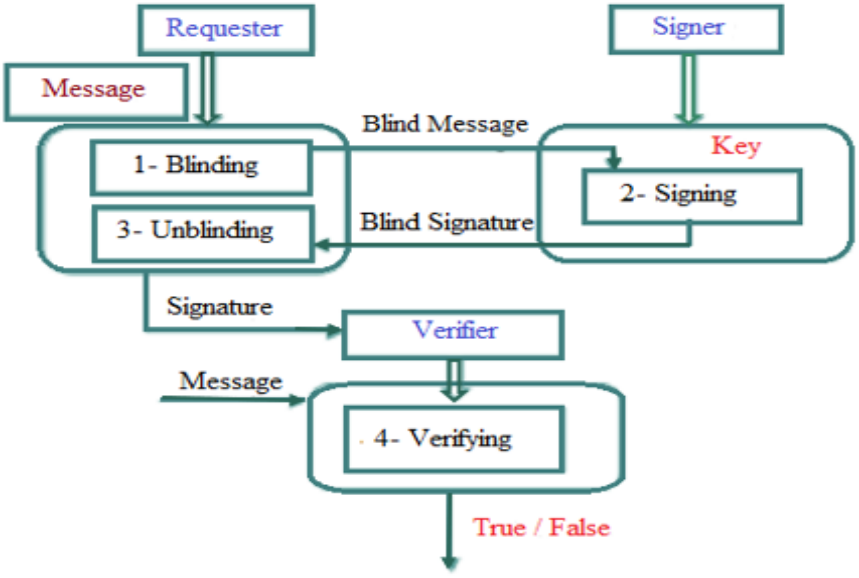

FIGURE 2.9: Blind Signature

2.7.1 Chaum's Blind Signature Scheme

Firstly, David Chaum [25] presented the blind signature scheme in the history. This scheme makes use of RSA cryptosystem. The reliability of this scheme based on the strength of RSA algorithm. This scheme merges RSA with blinding and unblinding characteristics.

The three functions make up the blind signature algorithm.

1. **Signing function:** The signer knows only the signing function $c^{'}$ and c is its publically known inverse, such that $c(c^{'}(y)) = y$ and c does not provide any information about $c^{'}$.

2. **Commuting function:** The requester only knows r, a commuting function and its inverse $r^{'}$, such that $r^{'}(c^{'}(r(y))) = c^{'}(y)$.

3. **Redundancy checking:** A redundancy checking generates t, which checks adequate redundancy to create a search for valid signatures impractical.

David Chaum [25] blind signature scheme is as follows.

1. Requester randomly selects y such that $t(y)$, forms $r(y)$, and sends $r(y)$ to the signer.

2. The signer sign the received $r(y)$ by using $c^{'}$ and sends back the $c^{'}(r(y))$ to the requester.

3. The requester unblind the received message by applying $r^{'}$, such that $r^{'}(c^{'}(r(y))) = c^{'}(y)$.

4. By using the public key c of the signer anyone can verify that the unblinded message $c^{'}(y)$ was created by the signer, and checking the $t(c(c^{'}(y)))$.

For further details see [25, 26, 52].

2.8 Hash Function

Hash function takes input data of arbitrary length and returns fixed-size output, called the hash value [53] (see Figure2.10). Normally a hash value is smaller than the actual value. Hashed message is called a message digest. Hash function ensures that if there is a small change in the input information then completely different output will be generated. It is developed by the National Institute of Standard and Technology (NIST) in 1993. The hash function must be collision resistance. Its examples include Secure Hash Algorithm (SHA) [54], MD5 [5], SHA-1 [55], SHA-2 [33] and SHA-3 [33].

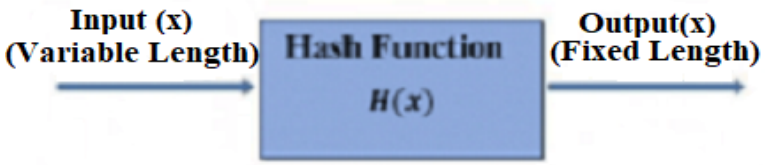

FIGURE 2.10: Hash Function

Properties of hash function Hash function has four properties which are described below.

1. **Performance:** It is easy to find $H(m)$, where m denotes the message.

2. **Weak Collision Resistance:** It is very difficult to retrieve m' if m and $H(m)$ are given such that

$$H(m) = H(m') \tag{2.10}$$

3. **Strong Collision Resistance:** It is not possible to compute two different inputs m' and m^* with the same hash values. That is, $m \neq m^*$ but

$$H(m') = H(m^*) \tag{2.11}$$

4. **One Way Function:** It is impossible to reverse the process that is, if $H(m)$ is given, it is not possible to find m.

2.9 Certificate Authority

A certificate authority (CA) is a trust able third party which assigns and authorized the certificates and public keys. For secure communication in a public network, we use these certificates and public keys. CA is an authority which

enhances trust between users. Through CA, users can easily confirm the authentication of each other's identities (such as email addresses or individual persons). See for further details [56].

2.10 Public Key Generator

Public Key Generator (PKG) is a trusted third party which generate the public key of the user directly from the user's identity. In identity based public key cryptography the public keys of the users are generated with the help of PKG.

2.11 Key Escrow Problem

Key escrow problem is the problem in which Public Key Generator (PKG) has the knowledge of private key of the user [56]. This problem usually occurs in the ID-based cryptography.

2.12 Certificate Based Cryptography

In 2003, Gentry [57] first presented the concept of certificate based public key cryptography. Certificate based public key schemes uses public key infrastructure that need less information for validation and distribution of certificates. The certificates are used in certificate based encryption scheme only for decryption. So to encrypt a message there is no need for digital certificates or authentication. The user's identity is needed for the encryption.

2.13 Certificateless Cryptography

Certificateless cryptography was first introduced by Alriyami and Paterson in 2003. The certificateless cryptography is a public key cryptography, which does not need

the certificates to verify the authenticity of public keys. CL cryptography does not depend on the trusted third party (TTP) also known as Certificate Authority (CA) [58].

CL-PKC is related to identity based cryptography, but it is not affected by the key escrow problem. In ID based cryptography, user's identity such as email, IP addresses are used as public key in place of digital certificates and secret key is created by the trusted third party. On the other hand, in CL-PKC, the secret key is created by both the user and the PKG collaboration [59].

2.14 Certificateless Blind Signature

After the Chaums Blind signature scheme many blind signature schemes were presented for different applications. The combination of cetificateless signature scheme and the blind signature scheme is known as certificateless blind signature. A CLB scheme was first proposed by Zhang and Zhang [24] in 2008. A typical certificateless blind signature scheme comprises of the following algorithms.

1. Setup
2. Partial private key extract
3. Set secret value
4. Set private key
5. Set public key
6. Sign
7. Verify

Due to less computational cost and storage CLB are widely used in many applications such as e-cash, e- voting and online shopping [32].

Chapter 3

The Certificateless Blind Signature Scheme using an Elliptic Curve Cryptography

In this chapter the article entitled "Certificateless Blind Signature Scheme" by Nayak et al. [29] is reviewed. Here their proposed CLB scheme is presented. At the end, its application in electronic cash is presented.

3.1 Introduction

Due to large applications in privacy related mechanisms, the digital signature is very important for offering authentication of the documents. Public key cryptography is a basis for the electronic signatures. Digital signatures provide accuracy, non-repudiation and integrity to the message over unsecured networks. In the process of generating public key the participant's identity is not considered in the traditional public key cryptosystem. Generally certificating authority (CA) is a trusted third party that issue and maintain the public key associated with a particular user [56]. The CA authenticates the participant's public key with its owner. CA faces many disadvantages of storing and managing user's certificates.

Moreover, a lot of computations are needed to execute these systems and it is mandatory for receiver to authenticate the sender's public key before using it. For solving this problem, the concept of an identity based public key cryptosystem (ID-PKC) was developed by Shamir [60]. In the proposed scheme the identities of particular user are used to create his public key. All the process is done by a certificate authority called PKG [61]. ID-PKC enjoys the benefits of storing and managing the public keys but the disadvantage is of key escrow problem. Alriyami and Paterson have presented a certificateless public key cryptosystem (CL-PKC) to manage identity based public key cryptography problem. CL-PKC gives all the properties of the ID-PKC without the key escrow problem [28]. The authenticity of the user's public key does not need certificates for guarantee in CL-PKC. The user creates his private key with the help of PKG.

User's authentication is important in number of applications such as e-commerce and e-voting systems. Many protocols of blind signature have been designed which are based upon mathematical hard problems, like the DLP and IFP. Various schemes are proven to be secure against existing attacks, and some are vulnerable to some cryptographic attacks [62]. The following properties must be satisfied by any blind signature scheme [63, 64].

1. Blindness:
2. Correctness:
3. Authenticity:
4. Unforgeability:
5. Non-repudiation:
6. Integrity:
7. Non-reuseability:
8. Untraceability:

An effective public key cryptosystem presented by Miller [65] and Koblitz [66] in which the group of points on an elliptic curve defined over a finite field is used. The use of ECC with smaller parameters [67] as compared to ElGamal and RSA achieves an equivalent level of protection. It was observed that a 256-bit ECC key can achieve a similar level of security as compared to 3,072-bit key in RSA [68]. Due to the smaller key size ECC has numerous benefits of faster processing, reduces processing power and the storage requirement. Such advantages make ECC,

suitable for a resource-restricted environment such as mobile phones, smart cards, etc. The first ECC-based blind signature scheme reveals that the storage space saved by it is 34% in contrast to the blind signature scheme depends on DLP [69]. The combination of blind signature scheme and certificateless blind signature scheme is known as certificateless blind signature scheme. This means that without disclosing the message content to the Signer, the Requester will get a certificateless signature. Certificateless blind signature (CLB) scheme was first introduced by Zhang and Zhang [24] in 2008. They present the concept of blind signature as a certificateless public key cryptography. Their scheme depends on the pairing based cryptography. Pairing based cryptography is the cryptography in which the use of pairing between elements of an additive cyclic group to the other multiplicative cyclic group. Actually pairing is a map between elements of one cyclic group to the other cyclic group that is $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ which satisfies the properties of bilinearity, non-degeneracy and computability [30]. In 2009 a provably secure certificateless blind signature scheme was present by Yang et al. [70]. This scheme is proven to be secure against two different types of adversaries. In contrast to earlier known ID-based blind signature schemes, this scheme is very efficient as it uses one pairing operation that is if $P, Q \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_P^*$, then $e(aP, bQ) = e(P, Q)^{ab}$. For further certificateless blind signature schemes we refer to [30, 71].

In literature, most of the existing certficateless blind signature schemes are based upon the difficulty of solving IFP, ElGamal, DLP, RSA and ECDLP. Before discussing the security aspects of this scheme, the next section 3.2 is devoted to the detailed description of scheme.

3.2 Certificateless Blind Signature Scheme of Nayak et al.

Recently, in 2017, Nayak et al. [29] proposed a CLB scheme using ECC. In this proposed scheme, three parties are involved, Public Key Generator (PKG), a Signer and a Requester. The seven phases used in proposed scheme are listed below:

1. Setup
2. Partial private key extraction
3. Secret value setup
4. Private key setup
5. Public key setup
6. Certificateless blind signing
7. Verification

In this scheme the lower case letters represent elements of $\mathbb{Z}_P^*$ and upper case letters denotes the points in $E_P(u, v)$.

Global parameters:

The scheme consists of the following global parameters.

G : A generator point of an elliptic curve $E_P(u, v)$ with larger order such that $nG = \mathcal{O}$

n : Order of base point G

$\mathcal{O}$: Point at infinity

$H(.)$: A collision-free hash function

Notations:

The following symbols are used in the scheme.

p_A : The secret key of PKG

s_B : The secret key choosen by the Signer

γ, ϕ : A random parameters choosen by the Requester

$[T]_x$: The x-coordinate of the elliptic point T.

$[T]_y$: The y-coordinate of the elliptic point T.

m : Message

In the proposed CLB-ECC scheme the operations in different phases are described bellow.

1. **Setup:** It contains two steps which are given below.

   Step 1: PKG chooses point G from $E_P(u, v)$ known as base point in an elliptic curve $E_P(u, v)$.

   Step 2: Then PKG selects $p_A \in \mathbb{Z}_P^*$ randomly and computes the public key as:

$$P_A = p_A G \tag{3.1}$$

2. **Partial Private Key Extraction:** PKG arbitrary selects $q \in \mathbb{Z}_P^*$ for a Signer and computes T, c and V.

$$T = qG \bmod p \tag{3.2}$$

$$c = (q + p_A[T]_x) \bmod p \tag{3.3}$$

$$V = cG \bmod p \tag{3.4}$$

Then PKG sends (c, T) with the identity ID_B to the Signer. The Signer can verify the authenticity of the received (c, T) by verifying the following conditions:

$$cG = T + [T]_x P_A \tag{3.5}$$

The system parameters published by PKG are $\langle E, G, V, P_A \rangle$.

3. **Secret Value Setup:** The Signer selects $r \in \mathbb{Z}_P^*$ as his/her secret information with identity ID_B.

4. **Private Key Setup:** The Signer first fixes his secret/private key $s_B \in \mathbb{Z}_P^*$ and using the above secret value r, computes the elliptic curve point W as:

$$W = rG \tag{3.6}$$

5. **Public Key Setup:** The Signer computes and publishes his public key S_B as:

$$S_B = s_B G \tag{3.7}$$

6. **Certificateless Blind Signing:** For a message m the following steps are involved in the creation of a blind signature.

Step 1: The Signer calculates:

$$x_1 = s_B r^{-1} \bmod p \tag{3.8}$$

$$x_2 = c s_B^{-1} \bmod p \tag{3.9}$$

and sends (W, x_1, x_2, c) to the Requester.

Step 2: The requester selects random integers $\gamma, \phi \in \mathbb{Z}_P^*$ and computes x'

and x as:

$$x' = H(m||x_1W||x_2S_B + W - \gamma G - c\phi G) \tag{3.10}$$

$$x = (x' + \phi) \bmod p \tag{3.11}$$

and sends x to the Signer. Where H is the hash function, fixed in the global setting.

Step 3: The partial signature $c^{'}$ is calculated by the Signer as:

$$c^{'} = (r - xc) \bmod p \tag{3.12}$$

The Signer then sends $c^{'}$ to the Requester.

Step 4: After getting $c^{'}$, the Requester calculates $c^{''}$ as:

$$c^{''} = (c^{'} - \gamma) \bmod p \tag{3.13}$$

So $\Omega = (c^{''}, x^{'})$ is the CLBS on the message m.

7. **Verification:** The Verifier performs the following steps to verify the authenticity of received signature Ω on the message m.

   Step 1: Calculates $x^{''}$ as:

$$x'' = H(m||S_B||c''G + (1 + x')V) \tag{3.14}$$

   Step 2: When $x^{''} = x^{'}$ then the Requester accepts the signature as valid one (see Figure 3.1).

   Proof of correctness: The proposed scheme of Nayak et al. [29] is correctly verifiable. The blind signature of the scheme are same on the Requester's and Verifier's ends. The subsequent steps in the proofs given below, are formed using the above equations.

$$\begin{aligned} x' &= H(m||x_1W||x_2S_B + W - \gamma G - c\phi G \\ &= H(m||s_Br^{-1}.rG||cs_B^{-1}.s_B.G + rG - \gamma G - c\phi G) \\ &= H(m||s_BG||cG + rG - \gamma G - c\phi G) \\ &= H(m||s_BG||cG + (c^{'} + xc)G - \gamma G - c\phi G) \end{aligned}$$

$$
\begin{aligned}
x' &= H(m||s_{_B}G||cG + (c^{'} + xc)G - \gamma G - c\phi G) \\
&= H(m||s_{_B}G||cG + c^{'}G + xcG - \gamma G - c\phi G) \\
&= H(m||s_{_B}G||cG + c^{'}G + (x^{'} + \phi)cG - \gamma G - c\phi G) \\
&= H(m||s_{_B}G||cG + c^{'}G + x^{'}cG + \phi cG - \gamma G - c\phi G) \\
&= H(m||s_{_B}G||cG + c^{'}G + x^{'}cG - \gamma G) \\
&= H(m||s_{_B}G||c^{'}G - \gamma G + cG + x^{'}cG) \\
&= H(m||s_{_B}G||(c^{'} - \gamma)G + cG + x^{'}cG) \\
&= H(m||s_{_B}G||(c^{''}G + c(1 + x^{'})G) \\
&= H(m||s_{_B}G||(c^{''}G + (1 + x^{'})V) \\
&= x''
\end{aligned}
$$

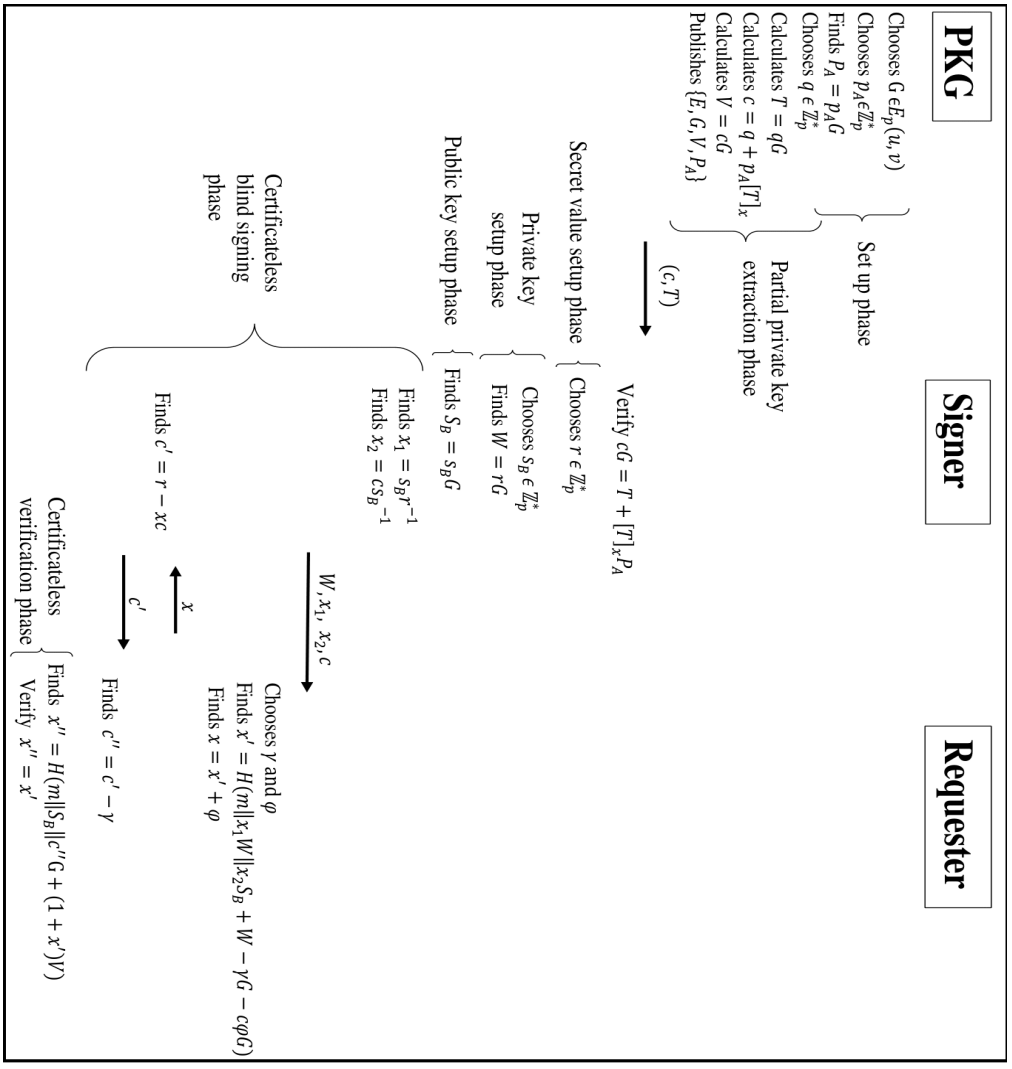

FIGURE 3.1: CLB scheme of Nayak et al.

For the detailed security analysis of the scheme we refer to [29].

3.3 Application in E-cash System

In 1982, D.Chaum introduced a first electronic cash payment on the basis of blind signature protocol. After that, many e-cash schemes are introduced [72]. In 2009, Ashrafi et al. [73] proposed a blind signature scheme using ECC. On the basis of this scheme they also introduced an offline electronic cash payment system. Nayak et al. [29] used electronic cash framework of Ashrafi et al. [73] to proposed an e-cash framework based on CLB-ECC.
The proposed e-cash payment system contains three parties: Bank, Customer and Merchant. The five phases involved in e-cash system are as follows.

1. Setup
2. Initialization Request
3. Initialization Response
4. Payment Request
5. Payment Processing

Global Parameters:
Global parameters are same as stated in Section 3.2 for the original scheme.

1. **Setup:**
   In this phase Bank chooses point $G \in E_{P}(u, v)$, $p_{A} \in \mathbb{Z}_{P}^{*}$ and computes the public key $P_{A} = p_{A}G$.
   The Merchant selects r as a secret information and $s_{B} \in \mathbb{Z}_{P}^{*}$, computes $S_{B} = s_{B}G$.
2. **Initialization Request:**
   Customer makes request for the Merchant's public key, Bank's Public Keys

and system parameters of the Bank and the Merchant when he/she wants to buy goods/services from a Merchant.

3. **Initialization Response:**

   Bank randomly chooses $q \in \mathbb{Z}_P^*$ and computes $T = qG$, $c = (q + p_A[T]_x) \bmod p$ and $V = c\,G$.

   Then sends (c, T) to the Merchant.

   Also Bank sends $\langle E, G, V, P_A \rangle$ to the Customer.

   The Merchant calculates $W = r\,G$, $x_1 = s_B r^{-1} \bmod p$ and $x_2 = c s_B^{-1} \bmod p$ using his/her own secret parameter. Then Merchant sends (W, x_1, x_2, c) to the Customer.

4. **Payment Request:**

   The Customer forms the e-coin m by adding the card information, validity period and the cost.

   He also selects γ and $\phi \in \mathbb{Z}_P^*$ as his/her transaction ID and calculates $x' = H(m||x_1 W||x_2 S_B + W - \gamma G - c\phi G)$, $x = (x' + \phi) \bmod p$ and sends x to the Merchant.

5. **Payment Processing:** The Merchant communicates to the Bank by transmitting his/her identity ID_B after getting the payment response from the Customer and can verify the authenticity of the obtained (c, T) by verifying the $cG = T + [T]_x P_A$.

   If above equation is verified, the Merchant calculates $c' = (r - xc) \bmod p$ and sends c' to the Customer. After getting c', Requester calculates $c'' = (c' - \gamma) \bmod p$. $\Omega = (c'', x')$ is the certificateless blind signature on e-cash(m).

   After this Customer sends (c'', x') (signature) and electronic coin to the Bank. After receiving the electronic coin and its signature the Bank checks the validity of the coin by calculating $x'' = H(m||S_B||c''G + (1 + x')V)$. Bank authenticates whether $x'' = x'$.

   If it verifies, Bank accepts the coin and withdraw the needed amount from the Customer's account and send it in to the account of Merchant (see Figure 3.2).

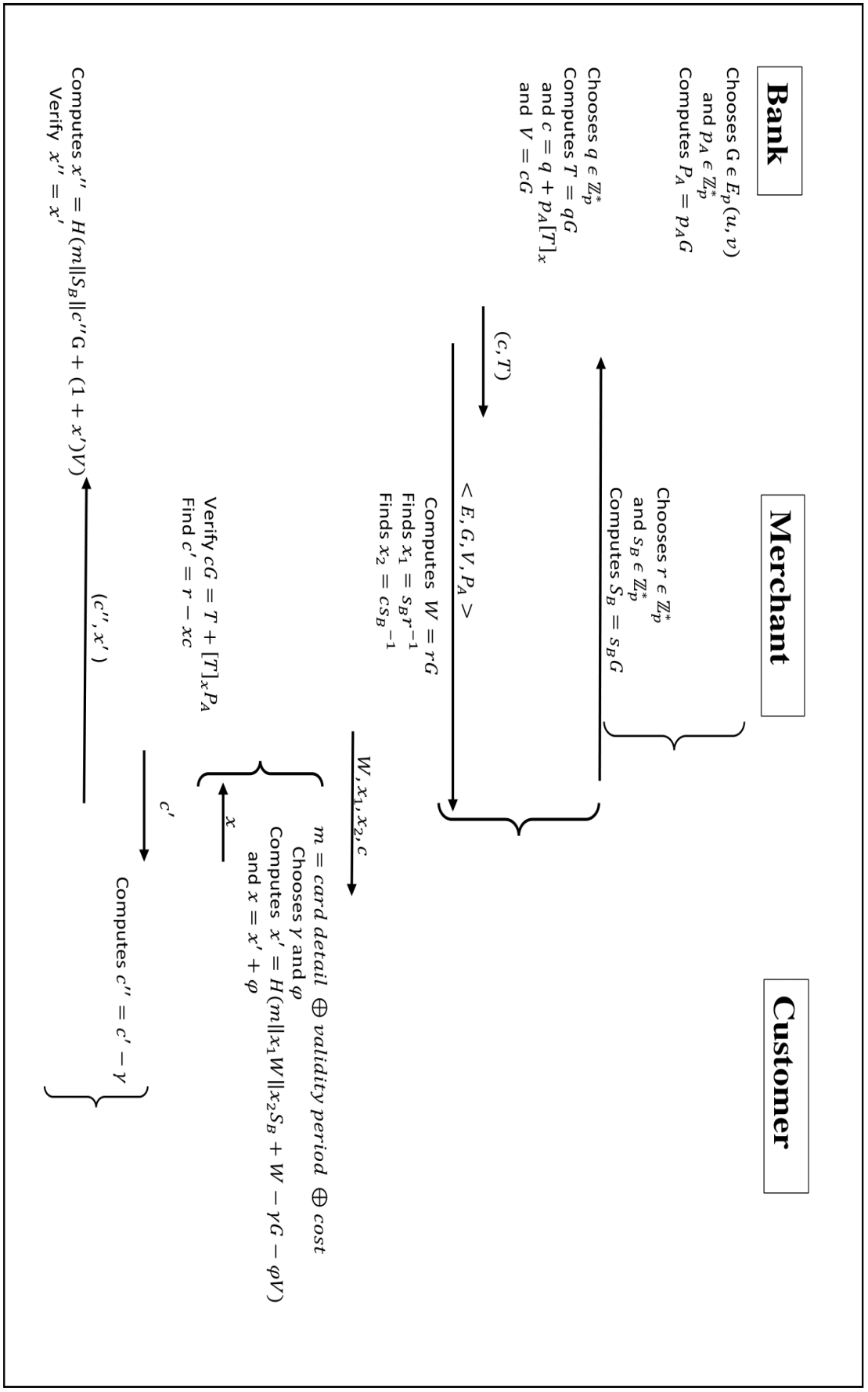

FIGURE 3.2: Proposed e-cash scheme of Nayak et al.

Chapter 4

Cryptanalysis

In this chapter, the certificateless blind signature scheme of Nayak et al. [29] is analyzed that is based on ECC and successfully apply the forgery attack. The analysis shows that the proposed scheme is not secure against the known cryptographic attack that is needed to be fixed.

4.1 The Forgery Attack

In this attack, an adversary Eve attempt to forge a blind signature for the message without knowing the related secret signing key of the Signer [40]. By using the previous signature, the Attacker tries to find another valid signature. That is, the Signer long term secret key is revealed by using the public parameters of the Signer.

Before describing the method of the attack, note that an Attacker has the following information.

- The point G of an Elliptic curve.
- Public keys of PKG and Signer.
- (c, T) sent by PKG to Signer.
- The Attacker also has information of W, x_1 and x_2.

First the Attacker will generate the secret key of the Signer and then uses this secret key to play the role of a fake Signer. The cryptanalysis is described in the following phases.

Phase-1

PKG performs the following steps.

1. Chooses $G \in E_{P}(u, v)$
2. Chooses random number $p_{A} \in \mathbb{Z}_{P}^{*}$
3. Computes $P_{A} = p_{A}G$
4. Chooses $q \in \mathbb{Z}_{P}^{*}$
5. Finds $T = qG$
6. Finds $c = (q + p_{A}[T]_{x}) \bmod p$
7. Finds $V = cG$

Then PKG sends (c, T) to the Signer. The system parameters published by PKG are $\langle E, G, V, P_{A} \rangle$.

Phase-2

After receiving (c, T), Signer executes the following steps.

1. Verifies $cG = T + [T]_{x}P_{A}$
2. Chooses $r \in \mathbb{Z}_{P}^{*}$
3. Chooses $s_{B} \in \mathbb{Z}_{P}^{*}$
4. Finds $W = rG$
5. Finds $S_{B} = s_{B}G$
6. Finds $x_{1} = s_{B}r^{-1} \bmod p$
7. Finds $x_{2} = cs_{B}^{-1} \bmod p$

and sends (W, x_{1}, x_{2}, c) to the Requester.

Phase-3

Attacker gets the secret key s_{B} by using the public parameters and the Extended Euclidean algorithm.

$$s_{_B} = x_{_2}^{-1}.c \bmod p$$

After finding the secret key of the Signer, attacker plays a role of a fake Signer. He then executes the following steps:

1. Selects $r^* \in \mathbb{Z}_{_P}^*$.
2. Calculates $W^* = r^*G$
3. Computes $x_{_1}^* = s_{_B}.(r^*)^{-1} \bmod p$

Sends $(W^*, x_{_1}^*, x_{_2}, c)$ to the Requester.

Phase-4

After receiving $(W^*, x_{_1}^*, x_{_2}, c)$, the Requester performs the following steps.

1. Chooses γ, $\phi \in \mathbb{Z}_P^*$
2. Computes $x'^* = H(m||x_{_1}^* W^* || x_{_2} S_{_B} + W^* - \gamma G - c\phi G)$
3. Computes $x^* = (x'^* + \phi) \bmod p$

and sends x^* to the Signer.

Phase-5

The Attacker intercepts the communication and sends c'^* to the Requester after computing it as follows:

$$c'^* = (r^* - x^*.c) \bmod p$$

Phase-6

The Requester will react as follows:

1. Computes $c''^* = (c'^* - \gamma) \bmod p$
2. Computes $x''^* = H(m||S_{_B}||c''^* G + (1 + x'^*)V)$
3. Verifies $x''^* = x'^*$

As $x''^* = x'^*$, so the Requester trusts that the authenticate Signer has signed the message. But actually there was a fake signer who signed the message.

Proof of correctness:

The proposed scheme of Nayak et al. [29] is successfully cryptanalyzed. The verification is given below. The subsequent steps in the proofs given below, are formed using the above equations.

(1). $x_1^* = s_B.(r^*)^{-1} \bmod p$

(2). $W^* = r^*G$

(3). $c'^* = (r^* - x^*.c) \bmod p$

(4). $x^* = (x'^* + \phi) \bmod p$

(5). $c''^* = (c'^* - \gamma) \bmod p$

$$\begin{aligned}
x'^* &= H(m||x_1^*W^*||x_2S_B + W^* - \gamma G - c\phi G) \\
&= H(m||s_Br^*.(r^*)^{-1}G||c.s_B^{-1}s_BG + r^*G - \gamma G - c\phi G) \\
&= H(m||s_BG||cG + r^*G - \gamma G - c\phi G) \\
&= H(m||S_B||cG + (c'^* + x^*.c).G - \gamma G - c\phi G) \\
&= H(m||S_B||cG + c'^*G + x^*.c.G - \gamma G - c\phi G) \\
&= H(m||S_B||cG + c'^*G + (x'^* + \phi).c.G - \gamma G - c\phi G) \\
&= H(m||S_B||cG + c'^*G + x'^*cG + \phi.cG - \gamma G - c\phi G) \\
&= H(m||S_B||cG + c'^*G + x'^*cG - \gamma G) \\
&= H(m||S_B||c'^*G - \gamma G + cG + x'^*cG) \\
&= H(m||S_B||(c'^* - \gamma)G + (1 + x'^*)cG) \\
&= H(m||S_B||c''^*G + (1 + x'^*)V) \\
&= x''^*
\end{aligned}$$

Hence $x'^* = x''^*$.

Note that in Phase 2, the Signer sends (W, x_1, x_2, c) to the Requester, but Attacker changes the role of Signer and becomes a fake Signer. He sends (W^*, x_1^*, x_2, c) instead of (W, x_1, x_2, c) to the requester. After this, all the communication is carried out between the Attacker and the Requester, but Requester thinks that he is communicating with authenticate Signer. In this way the Attacker successfully applies the signatures of his own choice on a message.

FIGURE 4.1: Cryptanalysis of proposed scheme

Chapter 5

Modified Certificateless Blind Signature scheme

As the analysis in Chapter 4 shows that the scheme is not secure against the forgery attack. Attacker can easily generate a signature of his choice and verifies it. The scheme can be modified to counter the attack described in Section 4.1. In this Chapter, a modified CLB scheme is presented to counter the forgery attack described in Chapter 4.

5.1 The modified CLB scheme

In this section, a modified version of the certificateless blind signature scheme is presented. In the modified scheme the lower case letters represent elements of $\mathbb{Z}_P^*$ and upper case letters denotes the points in $E_P(u, v)$.

Global parameters:

The modified scheme consists of the following global parameters.

G : A generator point of an elliptic curve $E_P(u, v)$ with larger order such that $nG = \mathcal{O}$

n : Order of base point G

$\mathcal{O}$: Point at infinity

$H(.)$: A collision-free hash function

Notations:

The following symbols are used in the scheme.

q : A random number choosen by the PKG

$p_{_A}$: A secret key choosen by the Requester

r : A random parameter choosen by the Signer

$s_{_B}$: A secret key choosen by the Requester

$E_{S_{_R}}$: Encryption algorithm with key $S_{_R}$

$D_{s_{_R}}$: Decryption algorithm with key $s_{_R}$

$[T]_{_x}$: A x-coordinate of the elliptic point T

$[T]_{_y}$: A y-coordinate of the elliptic point T

γ, ϕ : A random parameters choosen by the Requester

$s_{_R}$: A secret key choosen by the Requester

$S_{_R}$: A public key computed by the Requester as $S_{_R} = s_{_R}G$

m : Message

In the modified CLB scheme the operations performed in different phases are described bellow.

Phase-1

PKG performs the following steps.

1. Chooses $G \in E_{_P}(u, v)$
2. Chooses random number $p_{_A} \in \mathbb{Z}^*_{_P}$
3. Computes $P_{_A} = p_{_A}G$
4. Chooses $q \in \mathbb{Z}^*_{_P}$
5. Finds $T = qG$
6. Finds $c = (q + p_{_A}[T]_{_x}) \bmod p$
7. Finds $V = cG$

Then PKG sends (c, T) to the Signer. The system parameters published by PKG are $\langle E, G, V, P_{_A}, S_{_R} \rangle$.

Phase-2

After receiving (c, T), signer executes the following steps.

1. Verifies $cG = T + [T]_x P_A$
2. Chooses $r \in \mathbb{Z}_P^*$
3. Chooses $s_B \in \mathbb{Z}_P^*$
4. Finds $W = rG$
5. Finds $S_B = s_B G$
6. Finds $x_1 = s_B r^{-1} \bmod p$
7. Finds $x_2 = c s_B^{-1} \bmod p$
8. Finds $x_e = E_{S_R}(x_2) \bmod p$ by using elliptic curve encryption with the public key of the Requester.

The Signer publishes S_B and sends (W, x_1, x_e, c) to the Requester.

Phase-3

After receiving (W, x_1, x_e, c), the Requester first decrypts x_e by using his secret key s_R as follows:

$$x_2 = D_{s_R}(x_e) \bmod p$$

After getting x_2, Requester performs the following steps.

1. Chooses γ and $\phi \in \mathbb{Z}_P^*$
2. Computes $x' = H(m||S_B||x_2 S_B + W - \gamma G - c\phi G)$
3. Computes $x = (x' + \phi) \bmod p$

and sends x to the Signer.

Phase-4

The Signer computes c' as:

$$c' = (r - x.c) \bmod p$$

and sends c' to the Requester.

Phase-5

After getting $c^{'}$, the Requester performs the following steps.

1. Computes $c^{''} = (c^{'} - \gamma) \bmod p$
2. Computes $x'' = H(m||S_{_B}||c''G + (1 + x')V)$
3. Verifies $x^{''} = x^{'}$

So $\Omega = (c^{''}, x^{'})$ is the CLBS on the message m (see figure 5.1).

Verification:

The modified scheme of is correctly verifiable. The blind signature of the scheme is same on the Requester's and Verifier's ends. The subsequent steps in the proofs given below, are formed using the above equations

(1). $c^{'} = (r - x.c) \bmod p$

(2). $x = (x' + \phi) \bmod p$

(3). $c^{''} = (c^{'} - \gamma) \bmod p$

$$
\begin{aligned}
x' &= H(m||S_{_B}||x_{_2}S_{_B} + W - \gamma G - c\phi G \\
&= H(m||S_{_B}||cs_B^{-1}.s_{_B}.G + rG - \gamma G - c\phi G) \\
&= H(m||S_{_B}||cG + rG - \gamma G - c\phi G) \\
&= H(m||S_{_B}||cG + (c^{'} + xc)G - \gamma G - c\phi G \\
&= H(m||S_{_B}||cG + c^{'}G + xcG - \gamma G - c\phi G) \\
&= H(m||S_{_B}||cG + c^{'}G + (x^{'} + \phi)cG - \gamma G - c\phi G) \\
&= H(m||S_{_B}||cG + c^{'}G + x^{'}cG + \phi cG - \gamma G - c\phi G) \\
&= H(m||S_{_B}||cG + c^{'}G + x^{'}cG - \gamma G) \\
&= H(m||S_{_B}||cG + x^{'}cG + c^{'}G - \gamma G) \\
&= H(m||S_{_B}||(1 + x^{'})cG + (c^{'} - \gamma)G) \\
&= H(m||S_{_B}G||(c(1 + x^{'})G + c^{''}G \\
&= H(m||S_{_B}G||(c^{''}G + (1 + x^{'})V) \\
&= x''
\end{aligned}
$$

Hence $x' = x''$.

FIGURE 5.1: Modified scheme

5.2 Security Analysis

In this section, the security analysis of the modified scheme is presented. Like the original scheme, the security of the modified certificateless blind signature scheme depends on the difficulty of solving the elliptic curve discrete logarithm problem. It is also shown that the attack that is implemented on CLB does not work for the modified scheme, that is the forgery attack and the key only attack. The modified CLB scheme satisfies the blindness property and that it can withstand forgery attack and the key only attack because it is infeasible to find out the secret key of the signer due to hardness of solving ECDLP.

5.2.1 Blindness Property:

The modified scheme also provides the security property of blindness. The message's Signer is unable to see the contents of the message in the blind signature phase. Requester uses secret random integers γ and ϕ for blinding the original message in Phase 3 of the modified scheme. The Signer cannot find these secret numbers and therefore he is unable to create link between the valid blind signatures and previously stored blind signatures.

5.2.2 Forgery Attack:

In the modified scheme, the hash function SHA is used. The hash function has property that it is impossible to retrieve the message from the message digest. In Step 5 of Phase 2, from S_B and G it is impossible to compute s_B (that is a secret key of the Signer) because solving ECDLP is very difficult. In Step 4 of Phase 2 also W depends on r, from W it is impossible to generate r because ECDLP is difficult to solve. The forgery attack described in Section 4.1 will not work for the modified scheme.

Note that in Phase 3 of the cryptanalysis, the Attacker computes s_B by computing inverse of x_2 modulo p. But in the modified scheme an Attacker will not be able

to extract x_2 from x_e with solving ECDLP. Therefore the Requester cannot able to verify this signature. That is the fake signature will be rejected by the system. Hence from the valid signatures $(c^{''}, x^{'})$, it is impossible to generate another valid signature (c''^{*}, x'^{*}).

5.2.3 Key only Attack:

To apply key only attack successfully on the modified scheme, Attacker has to create valid signature pair. Consider that an adversary is able to create the valid signature pair. But due to the requirement of secret parameters r, s_B and γ, he will not be able to unblind the signature pair. Because these parameters are infeasible to find due to ECDLP.

5.3 Cost Analysis

In this section, we compare the modified scheme with other certificatelss blind signature scheme and the proposed certificatelss blind signature scheme [29] interms of the operations involved for working in the finite elliptic curve group in $E_P(u, v)$. The modified scheme slightly increases the cost because it involves encryption and decryption operations in the signing phase.

Let P_m represents the scalar multiplication, P_{ex} represents the exponentiation operation and P_e represents the pairing operation. As the encryption operation, and the decryption operation is done by using the point multiplication. So, P_m represents the encryption and decryption operation. The result of the comparison is shown in Table 5.1.

Phase	Zhang and Gao	Zhang and Zhang	CLB:ECC	Modified Scheme
Signing	$P_e + P_{ex} + 3P_m$	$P_e + P_{ex}$	$6P_m$	$7P_m$
Verifying	$P_e + P_{ex} + P_m$	$3P_e + 3P_{ex}$	$2P_m$	$2P_m$
Total	$2P_e + 2P_{ex} + 4P_m$	$4P_e + 4P_{ex}$	$8P_m$	$9P_m$

TABLE 5.1: Comparison of modified certificateless blind signature scheme

5.4 Application of the Modified Certificateless Blind Signature Scheme

The modification of the proposed CLB scheme can be applied in e-voting, e-cash, etc. The application of modified certificateless blind signature scheme in e-cash consists of the following steps. The application of the modified CLB scheme contains three parties: Customer, Merchant and Bank. It consists of five phases.

Global Parameters:

Global parameters are same as stated in Section 5.1 for the modified scheme.

Phase-1

Bank executes the following steps.

1. Chooses $G \in E_{P}(u, v)$
2. Chooses random number $p_{A} \in \mathbb{Z}_{P}^{*}$
3. Computes $P_{A} = p_{A}G$

Phase-2

Merchant chooses a secret value $r \in \mathbb{Z}_{P}^{*}$ and secret key $s_{B} \in \mathbb{Z}_{P}^{*}$, then computes S_{B} as:

$$S_{B} = s_{B}G$$

Phase-3

When Customer wants to buy goods from a Merchant. He sends request for the Merchant's and Bank's public keys, and system parameters of the Merchant and Bank. Then Bank performs the following steps.

1. Chooses $q \in \mathbb{Z}_{P}^{*}$
2. Computes $T = qG$
3. Computes $c = (q + p_{A}[T]_{x}) \bmod p$
4. Computes $V = cG$

Then Bank sends (c, T) to the Merchant.

The Bank sends as $\langle E, G, V, P_{A} \rangle$ to the Customer.

Phase-4

After receiving (c, T) from the Bank, Merchant performs the following steps.

1. Computes $W = rG$, where r is the transaction ID.
2. Computes $x_1 = s_B r^{-1} \bmod p$
3. Finds $x_2 = c s_B^{-1} \bmod p$
4. Finds $x_e = E_{S_R}(x_2) \bmod p$. He uses elliptic curve encryption with the public key of the Customer.

The Merchant sends (W, x_1, x_e, c) to the Customer.

Phase-5

Customer performs following steps after receiving (W, x_1, x_e, c) from the Merchant.

1. Decrypts x_e by using his secret key s_R as follows:

$$x_2 = D_{s_R}(x_e) \bmod p$$

2. Constitutes e-coin m = card detail + validity period + cost
3. Chooses $\gamma, \phi \in \mathbb{Z}_P^*$
4. Computes $x' = H(m||S_B||x_2 S_B + W - \gamma G - c\phi G)$
5. Computes $x = (x' + \phi) \bmod p$

and sends x to the Merchant. After receiving it, Merchant performs following steps.

1. Verifies $cG = T + [T]_x P_A$
2. Computes $c' = (r - xc) \bmod p$

Sends c' to the Customer.

Phase-6

After getting c', the Customer computes c'' by using his/her password γ.

$$c'' = (c' - \gamma) \bmod p$$

Sends (c'', x') (certificateless blind signature) and electronic coin to the Bank.

Phase-7

After receiving certificateless blind signature and electronic coin Bank verifies the signatures by the following steps:

1. Computes $x'' = H(m||S_B||c''G + (1 + x')V)$
2. Verifies $x'' = x'$.

If it verifies, the Bank accepts the coin and withdraw the needed amount from the Customer's account (see Figure 5.2).

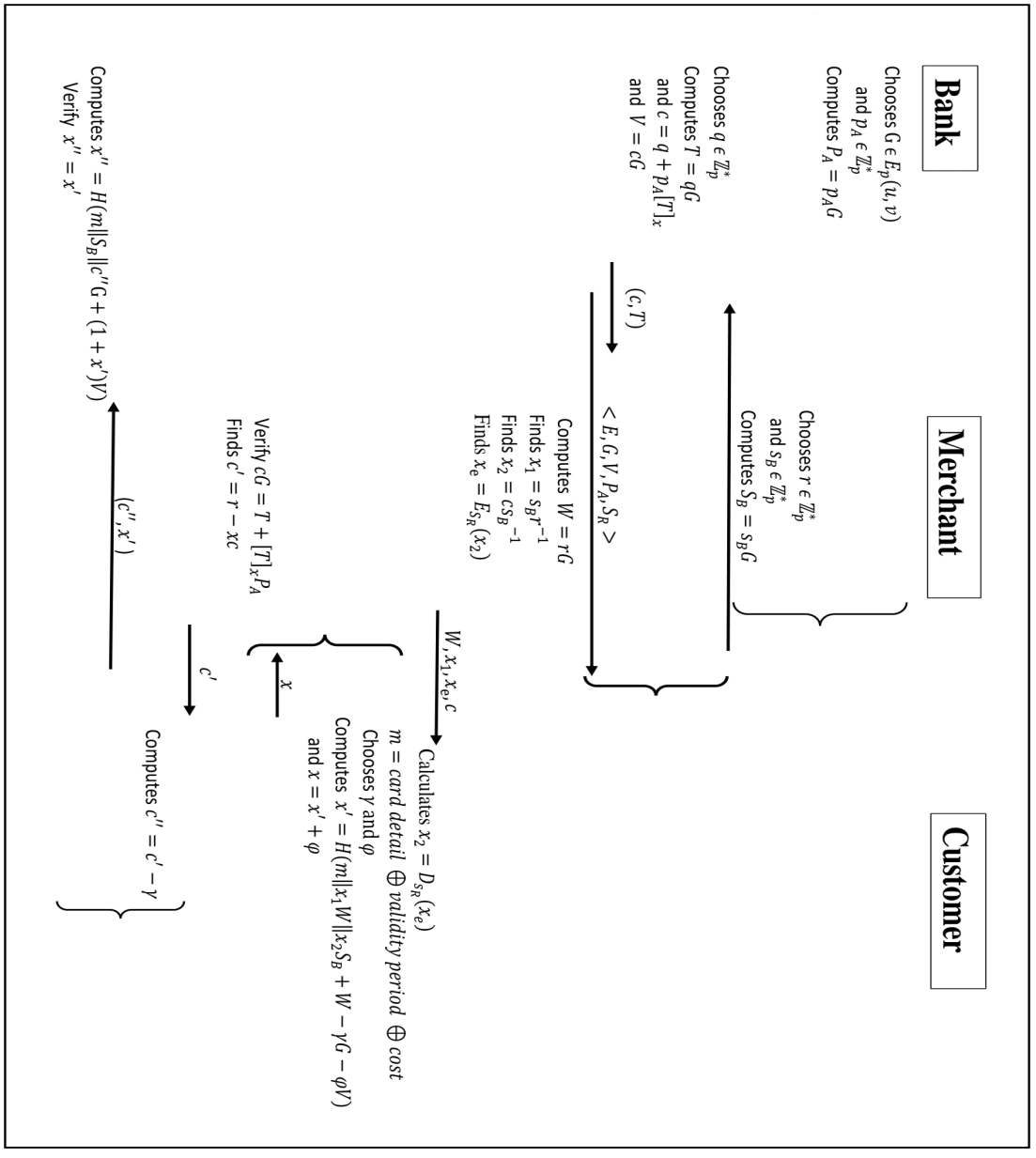

FIGURE 5.2: Application of modified scheme

5.5 Conclusion

In this thesis, the article "Certificateless Blind Signature Scheme using Elliptic Curve Cryptography (CLB:ECC)" proposed by Nayak et al. [29] is reviewed. The analysis of the scheme shows that it has many security flaws. Anyone can become a fake Signer after finding the secret key of the Signer by using public parameters. Because of the successful cryptanalysis, an Attacker can sign a message that will be verified at the Requester's end. The proposed scheme is unable to give the claimed security attributes. To fix the security issues a modified version of the scheme is presented. The modified version of the scheme is more secure than the original scheme because it involves the encryption and decryption process in the signing. The security of the modified scheme depends on the Elliptic Curve Discrete Logarithm Problem (ECDLP). The security analysis of the modified scheme is also given and shown that the same attack cannot implement. The modified scheme satisfies the blindness property and can withstand forgery attack, key only attack. At last the application of the modified scheme in the electronic cash system is presented.

Bibliography

[1] W. Trappe, *"Introduction to cryptography with coding theory"*. Pearson Education India, 2006.

[2] T. M. Damico, "A brief history of cryptography," *Inquiries Journal*, vol. 1, no. 11, 2009.

[3] O. Abraham and G. O. Shefiu, "An improved caesar cipher (icc) algorithm," *International Journal Of Engineering Science & Advanced Technology (IJESAT)*, vol. 2, pp. 1198–1202, 2012.

[4] S. Som, M. Kundu, and S. Ghosh, "A simple algebraic model based polyalphabetic substitution cipher," *International Journal of Computer Applications*, vol. 975, p. 8887, 2012.

[5] V. Pachghare, *Cryptography and information security*. PHI Learning Pvt. Ltd., 2019.

[6] D. R. Stinson and M. Paterson, *"Cryptography: theory and practice"*. CRC press, 2018.

[7] W. Stallings, "Cryptography and network security principles and practices," 2006.

[8] 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.

[9] A. Verma, P. Guha, and S. Mishra, "Comparative study of different cryptographic algorithms," *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, vol. 5, no. 2, pp. 58–63, 2016.

[10] A. Abdullah, "Advanced encryption standard (aes) algorithm to encrypt and decrypt data," *Cryptography and Network Security*, vol. 16, 2017.

[11] M. Agrawal and P. Mishra, "A comparative survey on symmetric key encryption techniques," *International Journal on Computer Science and Engineering*, vol. 4, no. 5, p. 877, 2012.

[12] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[13] M. M. Rahman, T. K. Saha, and M. A.-A. Bhuiyan, "Implementation of rsa algorithm for speech data encryption and decryption," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 12, no. 3, p. 74, 2012.

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

[15] D. Hankerson, A. J. Menezes, and S. Vanstone, *"Guide to elliptic curve cryptography"*. Springer Science & Business Media, 2006.

[16] H. Delfs, H. Knebl, and H. Knebl, *"Introduction to cryptography"*, vol. 2. Springer, 2002.

[17] R. A. Mollin, *"An introduction to cryptography"*. CRC Press, 2000.

[18] N. Kumar, "Investigations in brute force attack on cellular security based on des and aes," *IJCEM International Journal of Computational Engineering & Management*, vol. 14, pp. 50–52, 2011.

[19] M. Naor and M. Yung, "Public-key cryptosystems provably secure against chosen ciphertext attacks," in *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pp. 427–437, 1990.

[20] C. Rackoff and D. R. Simon, "Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack," in *Annual International Cryptology Conference*, pp. 433–444, Springer, 1991.

[21] M. Matsui, "Linear cryptanalysis method for des cipher," in *Workshop on the Theory and Application of of Cryptographic Techniques*, pp. 386–397, Springer, 1993.

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

[23] G. N. Nayak and S. G. Samaddar, "Different flavours of man-in-the-middle attack, consequences and feasible solutions," in *2010 3rd International Conference on Computer Science and Information Technology*, vol. 5, pp. 491–495, IEEE, 2010.

[24] L. Zhang and F. Zhang, "Certificateless signature and blind signature," *Journal of Electronics (China)*, vol. 25, no. 5, pp. 629–635, 2008.

[25] D. Chaum, "Blind signatures for untraceable payments," in *Advances in cryptology*, pp. 199–203, Springer, 1983.

[26] Z. Tan, Z. Liu, and C. Tang, "Digital proxy blind signature schemes based on dlp and ecdlp," *MM Research Preprints*, vol. 21, no. 7, pp. 212–217, 2002.

[27] D. Pointcheval and J. Stern, "Provably secure blind signature schemes," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 252–265, Springer, 1996.

[28] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *International conference on the theory and application of cryptology and information security*, pp. 452–473, Springer, 2003.

[29] S. K. Nayak, S. Mohanty, and B. Majhi, "Clb-ecc: Certificateless blind signature using ecc.," *JIPS*, vol. 13, no. 4, pp. 970–986, 2017.

[30] J. Zhang and S. Gao, "Efficient provable certificateless blind signature scheme," in *2010 International Conference on Networking, Sensing and Control (ICNSC)*, pp. 292–297, IEEE, 2010.

[31] S. Jose, A. Gautam, and C. P. Rangan, "A new certificateless blind signature scheme.," *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, vol. 5, no. 1, pp. 122–141, 2014.

[32] S. Kumar, *Certificateless Blind Signature based on DLP*. PhD thesis, 2015.

[33] S. Bose and P. Vijaykumar, *"Cryptography and Network Security"*. Pearson Education India, 2016.

[34] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *"Handbook of applied cryptography"*. CRC press, 2018.

[35] B. Schneier, "Description of a new variable-length key, 64-bit block cipher (blowfish)," in *International Workshop on Fast Software Encryption*, pp. 191–204, Springer, 1993.

[36] D. W. Kravitz, "Digital signature algorithm," July 27 1993. US Patent 5,231,668.

[37] M. W. Barsagade and S. Meshram, "Overview of history of elliptic curves and its use in cryptography," *International Journal of Scientific & Engineering Research*, vol. 5, no. 4, pp. 467–471, 2014.

[38] T.-C. Wu, C.-L. Lei, V. Rijmen, and D.-T. Lee, *"Information Security: 11th International Conference, ISC 2008, Taipei, Taiwan, September 15-18, 2008, Proceedings"*, vol. 5222. Springer Science & Business Media, 2008.

[39] T. Siegenthaler, "Decrypting a class of stream ciphers using ciphertext only," *IEEE Transactions on computers*, no. 1, pp. 81–85, 1985.

[40] H. C. Van Tilborg and S. Jajodia, *"Encyclopedia of cryptography and security"*. Springer Science & Business Media, 2014.

[41] J. B. Fraleigh, *"A first course in abstract algebra"*. Pearson Education India, 2003.

[42] J. A. Beachy and W. D. Blair, *"Abstract algebra"*. Waveland Press, 2019.

[43] R. Lidl and H. Niederreiter, *"Finite fields"*, vol. 20. Cambridge university press, 1997.

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

[45] F. Mallouli, A. Hellal, N. S. Saeed, and F. A. Alzahrani, "A survey on cryptography: Comparative study between rsa vs ecc algorithms, and rsa vs el-gamal algorithms," in *2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pp. 173–176, IEEE, 2019.

[46] Z. Vahdati, S. Yasin, A. Ghasempour, and M. Salehi, "Comparison of ecc and rsa algorithms in iot devices," *Journal of Theoretical and Applied Information Technology*, vol. 97, no. 16, 2019.

[47] L. Tawalbeh, M. Mowafi, and W. Aljoby, "Use of elliptic curve cryptography for multimedia encryption," *IET Information Security*, vol. 7, no. 2, pp. 67–74, 2013.

[48] S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," *SIAM Journal on computing*, vol. 17, no. 2, pp. 281–308, 1988.

[49] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE transactions on information theory*, vol. 31, no. 4, pp. 469–472, 1985.

[50] M. M. Khater, A. Al-Ahwal, M. M. Selim, and H. H. Zayed, "Blind signature schemes based on elgamal signature for electronic voting: A survey," *International Journal of Computer Applications*, vol. 975, p. 8887.

[51] C. Chandra, "Design of blind signature protocol based upon dlp," 2013.

[52] G. K. Verma, "Blind signature scheme over braid groups.," *IACR Cryptol. ePrint Arch.*, vol. 2008, p. 27, 2008.

[53] J. Menezes Alfred, "Handbook of applied cryptography/alfred j. menezes, paul c. van oorschot, scott a. vanstone," 1997.

[54] A. K. Sharma and S. Mittal, "Cryptography & network security hash function applications, attacks and advances: A review," in *2019 Third International Conference on Inventive Systems and Control (ICISC)*, pp. 177–188, IEEE, 2019.

[55] N. Harini, D. T. Padmanabhan, and C. Shyamala, "Cryptography and security," 2011.

[56] Z. Zhang, D. S. Wong, J. Xu, and D. Feng, "Certificateless public-key signature: security model and efficient construction," in *International Conference on Applied Cryptography and Network Security*, pp. 293–308, Springer, 2006.

[57] C. Gentry, "Certificate-based encryption and the certificate revocation problem," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 272–293, Springer, 2003.

[58] G. K. Verma, B. Singh, and H. Singh, "Provably secure certificate-based proxy blind signature scheme from pairings," *Information Sciences*, vol. 468, pp. 1–13, 2018.

[59] J. Baek, J. Newmarch, R. Safavi-Naini, and W. Susilo, "A survey of identity-based cryptography," in *Proc. of Australian Unix Users Group Annual Conference*, pp. 95–102, 2004.

[60] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the theory and application of cryptographic techniques*, pp. 47–53, Springer, 1984.

[61] D. H. Yum and P. J. Lee, "Generic construction of certificateless signature," in *Australasian Conference on Information Security and Privacy*, pp. 200–211, Springer, 2004.

[62] C.-I. Fan, W.-K. Chen, and Y.-S. Yeh, "Randomization enhanced chaum's blind signature scheme," *Computer Communications*, vol. 23, no. 17, pp. 1677–1680, 2000.

[63] Z. Shao, "Improved user efficient blind signatures," *Electronics Letters*, vol. 36, no. 16, pp. 1372–1374, 2000.

[64] S. K. Nayak, B. Majhi, and S. Mohanty, "An ecdlp based untraceable blind signature scheme," in *2013 International Conference on Circuits, Power and Computing Technologies (ICCPCT)*, pp. 829–834, IEEE, 2013.

[65] V. S. Miller, "Use of elliptic curves in cryptography," in *Conference on the theory and application of cryptographic techniques*, pp. 417–426, Springer, 1985.

[66] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of computation*, vol. 48, no. 177, pp. 203–209, 1987.

[67] J. Lopez and R. Dahab, "An overview of elliptic curve cryptography," 2000.

[68] S. A. Vanstone, "Elliptic curve cryptosystemthe answer to strong, fast public-key cryptography for securing constrained environments," *Information Security Technical Report*, vol. 2, no. 2, pp. 78–87, 1997.

[69] M. Chang, I. Chen, I. Wu, and Y. Yeh, "Schnorr blind signature based on elliptic curves," *Asian Journal of Information Technology, Published by Grace Publication Network*, pp. 130–134, 2003.

[70] X. Yang, Z. Liang, P. Wei, and J. Shen, "A provably secure certificateless blind signature scheme," in *2009 Fifth International Conference on Information Assurance and Security*, vol. 2, pp. 643–646, IEEE, 2009.

[71] S. Sun and Q. Wen, "Novel efficient certificateless blind signature schemes," in *2009 International Symposium on Computer Network and Multimedia Technology*, pp. 1–5, IEEE, 2009.

[72] J. Wang, "Realization of non-track electronic cash," *Procedia Engineering*, vol. 15, pp. 3265–3269, 2011.

[73] M. Z. Ashrafi and S. K. Ng, "Privacy-preserving e-payments using one-time payment details," *Computer Standards & Interfaces*, vol. 31, no. 2, pp. 321–328, 2009.