CAPITAL UNIVERSITY OF SCIENCE AND
TECHNOLOGY, ISLAMABAD

Generalized Signcryption Based on Elliptic Curve

by

Sohaib Hasan

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

in the

Faculty of Computing
Department of Mathematics

2021

First of all, I dedicate this research project to Allah Almighty, The most merciful and beneficent, creator and Sustainer of the earth

And

Dedicated to Prophet Muhammad (peace be upon him) whom, the world where we live and breathe owes its existence to his blessings

And

Dedicated to my parents, who pray for me and always pave the way to success for me

And

Dedicated to my teachers, who are a persistent source of inspiration and encouragement for me

CERTIFICATE OF APPROVAL

Generalized Signcryption Based on Elliptic Curve

by

Sohaib Hasan

(MMT183012)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Munaza Naz	FJWU, Rawalpindi
(b)	Internal Examiner	Dr. Qamar Mahmood	CUST, Islamabad
(c)	Supervisor	Dr. Rashid Ali	CUST, Islamabad

Supervisor Name
Thesis Supervisor
December, 2021

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

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

Author's Declaration

I, **Sohaib Hasan** hereby state that my MS thesis titled "**Generalized Signcryption Based on Elliptic Curve**" 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 MS Degree.

(Sohaib Hasan)
Registration No: MMT183012

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "**Generalized Signcryption Based on Elliptic Curve**" 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 MS Degree, the University reserves the right to withdraw/revoke my MS 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.

(Sohaib Hasan)
Registration No: MMT183012

Acknowledgement

First and foremost I would like to pay my cordial gratitude to the almighty **Allah** who created us as a human being with the great boon of intellect. I would like to pay my humble gratitude to the Allah almighty, for blessing us with the holy prophet **Hazrat Mohammad (Sallallahu Alaihy Wa'alihi wassalam)** for whom the whole universe is being created. He (Sallallahu Alaihy Wa'alaihi wassalam) removed the ignorance from the society and brought us out of darkness. Thanks again to that Monorealistic power for granting me with a strength and courage whereby I dedicatedly completed my MPhil thesis with positive and significant result. I owe honour, reverence and indebtedness to my accomplished supervisor and mentor **Dr. Rashid Ali** whose affectionate guidance, authentic supervision, keen interest and ingenuity was a source of inspiration for commencement, advancement and completion of the present study.

I would also like to thank especially to the PhD scholars Mr. Malik Zia and Mr. Omar Rabbani for his valuable guidance, suggestions and comments in the completion of my thesis. My heartiest and sincere salutations to my Parents, who put their unmatchable efforts in making me a good human being.

May Almighty Allah shower His choicest blessings and prosperity on all those who helped me in any way during the completion of my thesis.

Sohaib Hasan.

Abstract

A cryptographic technique called signcryption combines the role of digital signature and encryption in a single logical step. Signcryption provides message confidentiality and authenticity at same time. A generalized signcryption scheme provides the extra features, it works in signcryption mode when both confidentiality and authenticity are required and it works in signature mode or encryption mode when one of them is required. In this thesis, we extend the Gupta and Kumar's signcryption scheme to a generalized signcryption scheme. The proposed scheme provides extra features, it works in three different modes such as signcryption mode, signature only mode, encryption only mode. The security of scheme depends on Elliptic Curve Discrete Logarithm Problem (ECDLP) which is currently secure. The analysis of the scheme shows that it has resistance against many known cryptographic attacks. The proposed scheme has the security features of non-repudiation, unforgeability, message confidentiality, forward secrecy, integrity, authentication and unforgeability. The correctness and cost analysis of the proposed scheme is presented which prove the security and efficiency of the scheme.

Contents

List of Figures

List of Tables

Abbreviations

AES	Advanced Encryption Standard
DES	Data Encryption Standard
DH	Diffie-Hellman Key Exchange Protocol
DLP	Discrete Logarithm Problem
DSA	Digital Signature Algorithm
ECC	Elliptic Curve Cryptography
ECDHP	Elliptic Curve Deffie Helmen Problem
ECDLP	Elliptic Curve Discrete Logarithm Problem
ECDSA	Elliptic Curve Digital Signature Algorithm
GCD	Greatest Common Divisor
GSC	Generalized Signcryption
PKC	Public Key Cryptography
RSA	Rivest-Shamir-Adleman

Symbols

C	Ciphertext
D	Decryption algorithm
D_k	Decryption key
E	Encryption algorithm
E_k	Encryption key
k	Key
$\mathcal{O}$	Point at infinity
P	Plaintext or message
$\mathbb{R}$	Set of real numbers
$\mathbb{Z}$	Set of integers
$\mathbb{Z}_p$	Finite field of order prime p

Chapter 1

Introduction

Cryptography [1] is the study of the transmission of a message in such a way that it cannot be read by any unauthorized party. It is the technique that uses mathematical functions for protecting the data or information from adversaries. Using an encryption algorithm [2], the initial message known as plaintext is translated into a ciphertext for transmission to the public network. The ciphertext is then converted back to plaintext through the decryption algorithm by the receiver or an authorised individual. For encryption and decryption, both the sender and recipient use secret information (known only to the sender and receiver). This secret information is known as a key. The whole structure is called a cryptosystem. The security of cryptosystem's depends on the secret key.

The cryptographic scheme is classified into two primary groups, namely symmetric key cryptography and asymmetric key cryptography. In symmetric key encryption only one key is used which is only known to the sender and reciever. Examples of symmetric key encryption [3] are DES [4] and AES [5]. In this strategy, the main issue is key delivery between the sender and the receiver. When we have thousands of users to connect with each other, it becomes a significant challenge to allocate the key among all the participants. Diffie and Helman [6] introduced the concept of asymmetric key cryptography to address this problem, which is also known as Public Key Cryptography (PKC). In PKC, participants has two forms of

encryption keys, one is a public key that is made public and the other is a private key. RSA [7] and ElGamal [8] are the examples of asymmetric key cryptography.

An authentication mechanism that allows the sender of a message to attach a code that acts as a signature is called a digital signature which is an electronic equivalent of a person's physical signature [9]. Digital signatures consist of three algorithms: key generation, generation of signatures, and verification of the signatures.
For decades, it has been a tradition for the originator of the message to write his/her signature on it and then seal it in an envelope before handing it over to a deliverer in order to prevent forgery and maintain secrecy of the contents of a letter. The way people perform safe and authenticated communications has been revolutionized by public key cryptography, discovered almost two decades ago. It is now possible for individuals who have never interacted before to connect with each other through both open and inaccessible networks such as the internet in a safe and authenticated manner. In doing so, the same two-step technique has been implemented. The sender will sign it with a digital signature scheme, and then encrypt the message using a private key encryption algorithm under a randomly selected message encryption key, before a message is sent out. The key for random message encryption will then be encrypted using the public key of the receiver. This two-step technique is called signature-then-encryption. The weaknesses in signature then encryption are: amount of bits, computational cost.

1.1 Introduction to Domain

- The equation of the form $y^2 = x^3 + ax + b \mod p$ is called Weierstrass equation, where $p > 3$ be any prime and a, b are Weierstrass coefficients and they are selected from the finite field $\mathbb{F}_p$. The curve is said to be smooth if the discriminant $4a^3 - 27b^2 \neq 0$ and this curve is called elliptic curve.

- **Confidentiality**: It should be infeasible for an adaptive attacker to access any secret information from the encrypted text without the knowledge of the private key of the sender or designated receiver.

- **Unforgeability**: For an adaptive attacker to disguise as a sincere sender in generating an accurate encrypted text that can be recognized by the decryption algorithm, it should be computationally-infeasible.

- **Non-repudiation**: The receiver should have the potential to show to a third party that the signcrypted text sending by the authentic sender. This means that his previously encrypted messages will not be rejected by the sender.

- **Integrity**: It should be possible for the receiver to check that the message received is the same that was sent by the sender.

- **Public verifiability**: Any third party (judge) without the need for a recipient's private key can verify that the signcrypted text is valid or not.

- **Forward secrecy**: If the sender's secret key is stolen, no one should be able to retrieve plaintext from previously encrypted messages. Without forward secrecy, if the secret key is stolen, all previous released messages will no longer be trustworthy in a encryption scheme. As the risk of key leakage is becoming more severe as cryptographic computations are more commonly conducted on poorly secured devices such as cell phones. In such schemes, forward confidentiality appears to be an important feature.

1.2 Literature Review

In 1997, Zheng [10] introduced a new cryptographic technique called 'signcryption' which fulfills both the functions of digital signature and encryption in a logically single step. Its cost is significantly lower than that needed by signature-then-encryption technique. Not all messages require the features of confidentiality as well as authenticity. If only one of the two features is needed then the signcryption scheme is not effective. According to Zheng, signcryption can be replaced by signature-then-encryption algorithm in this scenario. Thus, we must use three cryptographic algorithms to solve the problem. Encryption, signature and signcryption as required and they termed as generalized signcryption.

Many variants of signcryption schemes have been proposed since 1997. Elliptic curve cryptography was suggested by Koblitz [11] and Miller [12] in 1985. In 1999, Imai and Zheng [13] used the elliptic curve cryptography (ECC) in signcryption and suggested a signcryption scheme whose security depends on the eliptic curve discrete logarithm problem (ECDLP) [14]. They showed that signcryption depends on the ECC has nearly 58% computational cost and 40% of communication cost expenses are lower than the cost required by signature then encryption scheme [15]. Later on, Bao and Deng [16] pointed out that the judge could not check the sender's authenticity without the secret key of the receiver, so they have extended Zheng's signcryption scheme in such a way that judge can verify the data without the secret key of the receiver. Gamage [17] proposed a signcryption scheme that allows anyone to verify sender's authentication, but only firewalls have been fixed in the application field. Jung [18] figure out that scheme of Zheng does not provide forward secrecy when the sender's secret key is illuminated. In 2005, Ren-Junn Hwang [19] proposed an eliptic curve discrete logarithm problem (ECDLP) and eliptic curve Diffie Helman problem (ECDHP) [20] based signcryption scheme with additional forward secrecy and public verification.

A definition of digital signature algorithm (DSA) was used by Shin [21] and suggested a DSA verifiable signcryption scheme, but there is no forward secrecy in the scheme. Raylin Tso [22] suggested a signature scheme depends on the toughness of the elliptic curve digital signature algorithm (ECDSA). The different ECC based signature schemes proposed in the literature are given in a recent overview [23]. In particular, confidentiality, integrity, authentication, unforgeability, non-repudiation, forward secrecy, and public verification were compared in terms of security attributes. For more signcryption schemes we refer to [24–29].

In 2006, Han and Yang [30] proposed a new idea of signcryption system, that can be used separately as an encryption system, and as a signature system when required. They termed the new primitive as generalized signcryption.
There schemes is based on elliptic curves. Wang [31] enhanced the scheme [30] and provided generalized signcryption scheme security concepts. In the year 2010, Yu et al [32] introduced an 'identity-based GSC' system and a security model. Kushwah and Lal [33] simplified the system's security model [32] and recommended a

more powerful GSC identity-based system in 2011. In 2014, Zhou et al [34] suggested a certificateless GSC scheme that could survive a KGC malicious-butpassive attack [35]. In the traditional model, Wei et al [36] proposed an identity-based GSC scheme and extended it in 2015 to big data protection. Zhou [37] identified and strengthened the assault on the scheme [38] in the same year. Han and Lu [39] consequently recommended, in the traditional model, an attribute-based GSC scheme and extended it to online social networks. In 2016, Zhou et al [40], [41] expanded GSC, added two new definitions 'generalized proxy signcryption and generalized signcryption' and suggested a concrete scheme. Zhang et al [42] suggested a certificateless lightweight certification.

1.3 Thesis Contribution

In this research, the Kumar and Gupta's signcryption scheme [43] has been extended to a generalized signcryption scheme. This scheme [43] uses elliptic curves for secure and authenticated message delivery, which performs all the functions of digital signature and encryption with a cost less than that required by the current standard signature than encryption method. The security of this scheme depends on the ECDLP and ECDHP, which are currently secure. The scheme provides integrity, message confidentiality, forward secrecy, unforgeability, verification, and non-repudiation security attributes. The computational time of this scheme is little bit higher than the Zheng and Imai scheme [13] but it is more secure. The proposed scheme performs double functions when both confidentiality and authenticity are required and it performs a single function without any additional calculations when confidentiality and authenticity required separately. The proposed scheme fulfils all the security attributes and it is more resistant against various known attacks.

1.4 Organization of Thesis

The rest of the thesis is organised as follows:

In Chapter 1, the introduction of cryptography, digital signature and basic terms

related to thesis will be discussed.

In Chapter 2, some basic concepts and material related to basic algebra and elliptic curve cryptography which facilitates the reader's best understanding of the terms are presented.

In Chapter 3, the basic terminology and concepts related to signcryption are presented.

In Chapter 4, the proposed generalized signcryption scheme which is based on elliptic curve will be discussed.

In Chapter 5, the analysis of proposed generalized signcryption scheme is presented.

Chapter 2

Premilinaries

The objective of this chapter is to present some basic definitions, cryptographic backgrounds from algebra and number theory that are needed for a good understanding of the work done in this thesis. The terms related to elliptic curve cryptography will also be discussed in the later sections.

2.1 Mathematical Background

Definition 2.1.1.
"A **group** G, sometimes denoted by $\{G, \cdot\}$ is a set of elements with a binary operation, denoted by $\cdot$, that associates to each ordered pair (a, b) of elements in G an element $(a \cdot b)$ in G, such that the following axioms are obeyed:

1. **Closure:** If a and b belong to G, then $a \cdot b$ is also in G.
2. **Associative:** $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ for all a, b, c in G.
3. **Identity element:** There is an element e in G such that $a \cdot e = e \cdot a = a$ for all a in G.
4. **Inverse element:** For each a in G there is an element $a^{'}$ in G such that $a \cdot a^{'} = a^{'} \cdot a = e$

A group G is said to be **abelian** if it satisfies the following additional condition $a \cdot b = b \cdot a$ for all a, b in G.

A group G is **cyclic** if every element of G is a power a^k (k is an integer) of a fixed element $a \in G$. The element a is said to generate the group G, or to be a generator of G " [1]

Example 2.1.2.

The following are the examples of a group and cyclic group.

1. The set of real numbers $\mathbb{R}$, complex numbers $\mathbb{C}$, integers $\mathbb{Z}$ all are groups under $+$.
2. Set of integers $\mathbb{Z}$ is not a group under multiplication.
3. All the characteristic of being a group are retained in the set of integers $\mathbb{Z}$. In addition all the elements of $\mathbb{Z}$ can be generated by 1 and -1 with respect to addition which are called generators.

Definition 2.1.3.

"A nonempty set $(F, +, \cdot)$ together with binary operations '$+$' and '$\cdot$' is called **field** F, if the following properties hold:

1. F is abelian under addition.
2. F forms an abelian group under multiplication (only nonzero elements).
3. Multiplication is distributed over addition in F." [44]

Example 2.1.4.

Some examples of field are given below:

1. Set of real numbers $\mathbb{R}$ and complex numbers $\mathbb{C}$ are fields.
2. Set of $\mathbb{Z}$ is not a field because the multiplicative inverse does not hold in $\mathbb{Z}$.

Definition 2.1.5.

"The elements of **Galois Field** $GF(p^n)$ is defined as,
$GF(p^n) = (0, 1, 2, \ldots p-1) \cup (p, p+1, p+2 \ldots p+p-1) \cup (p^2, p^2+1, p^2+2, \ldots p^2+p-1) \cup \ldots \cup (p^{n-1}, p^{n-1}+1, p^{n-1}+2. \ldots p^{n-1}+p-1)$ where n is any integer and p is prime. The order of Galois field is given by p^n while p is called the characteristics of field." [45]

Example 2.1.6.

$GF(5) = (0, 1, 2, 3, 4)$ which consists of 5 elements where each of them is a polynomial of degree 0.

Definition 2.1.7.

For any two given integers a and b to find an integer c such that $a \cdot c \equiv 1 \bmod b$ 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 prime to b that is $gcd(a, b) = 1$.

Definition 2.1.8.

"Given $x, y \in \mathbb{Z}_{\text{|}}$ such that

$$x^n = y \mod p$$

then finding n is known as **discrete logarithm problem**."[46]

Algorithm 2.1.9 (Euclidean Algorithm)**.**

To find the gcd of the integers P and Q, below mentioned steps are to be followed.

Input: Two integers P and Q

Output: $\gcd(P, Q)$

1. If $P = 0$ then $\gcd(P, Q) = Q$, since $\gcd(0, Q) = Q$ and stop.
2. If $Q = 0$ then $\gcd(P, Q) = P$, since $\gcd(P, 0) = Q$ and stop.
3. Write $P = Q.B + R$ where B is quotient and R is reminder.
4. Find $\gcd(Q, R)$, since $\gcd(P, Q)$=$\gcd(Q, R)$

Algorithm 2.1.10 (Extended Euclid Algorithm)**.**

To find the inverse of a under modulo m, below mentioned steps are to be followed:

Input: a and n

Output: $a^{-1} \mod n$

1. Set $(X, Y, Z) = (1, 0, n)$ and $(P, Q, R) = (0, 1, a)$
2. **If** $R = 0$, return that the inverse does not exist and Z is the gcd of (a, n).
3. If $R = 1$, return that the inverse is P and Q is the gcd of (a, n)

4. Store $T = \lfloor Z/R \rfloor$ where $\lfloor . \rfloor$ represents the floor value.

5. $(L, M, N) = (X - TP, Y - TQ, Z - TR)$

6. $(X, Y, Z) = (P, Q, R)$

7. $(X, Y, Z) = (L, M, N)$

Go back to step no.2

Table 2.1 and Table 2.2 shows the addition and multiplication of field GF_{13}.

+	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
0	0	1	2	3	4	5	6	7	8	9	10	11	12
1	1	2	3	4	5	6	7	8	9	10	11	12	0
2	3	4	5	6	7	8	9	10	11	12	0	1	2
3	4	5	6	7	8	9	10	11	12	0	1	2	3
4	5	6	7	8	9	10	11	12	0	1	2	3	4
5	6	7	8	9	10	11	12	0	1	2	3	4	5
6	7	8	9	10	11	12	0	1	2	3	4	5	6
7	8	9	10	11	12	0	1	2	3	4	5	6	7
8	9	10	11	12	0	1	2	3	4	5	6	7	8
9	10	11	12	0	1	2	3	4	5	6	7	8	9
10	11	12	0	1	2	3	4	5	6	7	8	9	10
11	12	0	1	2	3	4	5	6	7	8	9	10	11
12	0	1	2	3	4	5	6	7	8	9	10	11	12

TABLE 2.1: Addition in $GF(13)$

In Table 2.1 the two elements whose sum is 0 are additive inverses of each other and in Table 2.2 the two elements whose multiplication is 1 are multiplicative inverses of each other.

×	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	10	11	12
2	0	2	4	6	8	10	12	1	3	5	7	9	11
3	0	3	6	9	12	2	5	8	11	1	4	7	10
4	0	4	8	12	3	7	11	2	6	10	1	5	9
5	0	5	10	2	7	12	4	9	1	6	11	3	8
6	0	6	12	5	11	4	10	3	9	2	8	1	7
7	0	7	1	8	2	9	3	10	4	11	3	10	4
8	0	8	3	11	6	1	9	4	12	7	2	10	5
9	0	9	5	1	10	6	2	11	7	3	12	8	4
10	0	10	7	4	1	11	8	5	2	12	9	6	3
11	0	11	9	7	5	3	1	12	10	8	6	4	2
12	0	12	11	10	9	8	7	6	5	4	3	2	1

TABLE 2.2: Multiplication in $GF(13)$

Definition 2.1.11.

"**Trapdoor function** is a function that is easy to compute in one direction but difficult to compute in the reverse direction if some special information known as 'Trapdoor' is not known."[47]

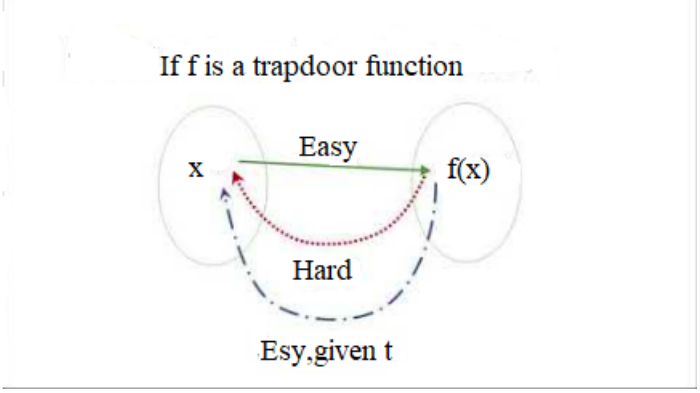

FIGURE 2.1: Trapdoor Function

Definition 2.1.12.

A **hash function** is a function that takes a collection of arbitrary-sized inputs

and fits them into a table or other data structure with fixed-size components. The widely used hashing algorithms are Secure Hash Algorithm (SHA).

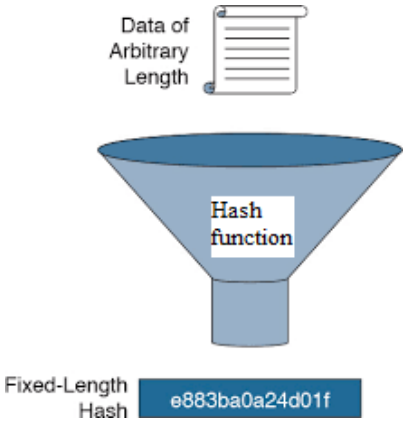

FIGURE 2.2: Hash Function

Hash Function Properties

There are some properties of hash function are as follows:

1. It is easy to compute $H(t)$, where t is the message.
2. If $H(t)$ is given it is impossible to find t. So, it is one way hash function.
3. In weak collision resistance, if t and $H(t)$ are given it is very difficult to find t' such that $H(t) = H(t')$.
4. In strong collision resistance, it is computationally in-feasible to find two different inputs t_1, t_2 such that $H(t_1) = H(t_2)$.

Theorem 2.1.13.

*"**Fermat's theorem** states that, If p is prime and a is a positive integer not divisible by p then $a^{p-1} - 1 \equiv 0 \mod p$". [1]*

Definition 2.1.14.

"An **integer factorization problem** is defined as, let m be a given number and $m \in Z$, the problem of decomposition of m to the product of prime p_α and q_β such that $m = p_\alpha q_\beta$." [46]

2.2 Cryptographic Background

The word 'cryptology' is a mixture of two Greek words, kryptos (hidden) and logos (words). It has two further types: cryptography and cryptanalysis.

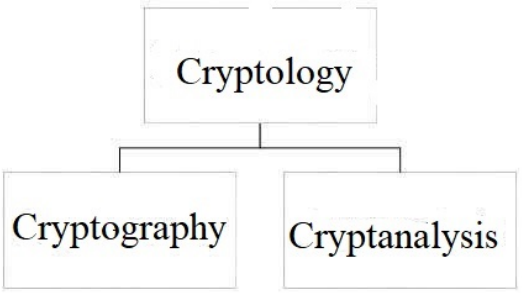

FIGURE 2.3: Cryptology

2.2.1 Cryptography

Cryptography is the study of the transmission of a message in such a way that it cannot be read by any unauthorized party. The original message is called plaintext and encoded message is called ciphertext. An algorithm is necessary to convert the plaintext message into ciphertext message is called an encryption algorithm [2]. The algorithm for decryption converts the ciphertext back into plaintext. For encryption and decryption cryptographic schemes require special information, which is exchanged between sender and recipient is known as a key. A cryptosystem consists of a message space, a ciphertext space, a key space, an encryption algorithm and a decryption algorithm. Based on key distribution, cryptography is classified into two main groups. Symmetric key cryptography (Secret key cryptography) and Asymmetric key cryptography (Public key cryptography).

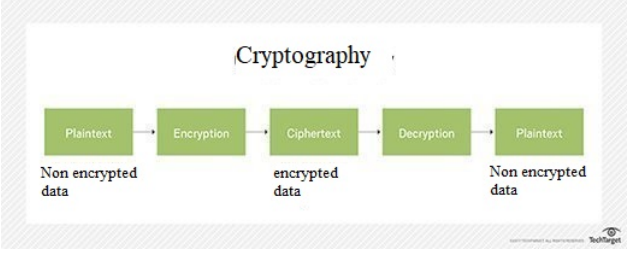

FIGURE 2.4: Cryptography

2.2.2 Symmetric key Cryptography

Symmetric key cryptography uses only one key to carry out encryption or decryption. A collection of data that is used to encrypt and decrypt the data in a symmetrical encryption is called a secret key. It often referred to as a private key. It can be transmitted via secure channel between two parties. Example includes: Data Encryption Standard (DES) [48], Dual Data Encryption Standard (2DES) [49], Triple Data Encryption Standard (TDES) [50], Advanced Encryption Standard (AES) [51].

The main advantages of symmetric key cryptography are high speed, strength of algorithms and availability of algorithms. The disadvantages are key management, key distribution and limited security.

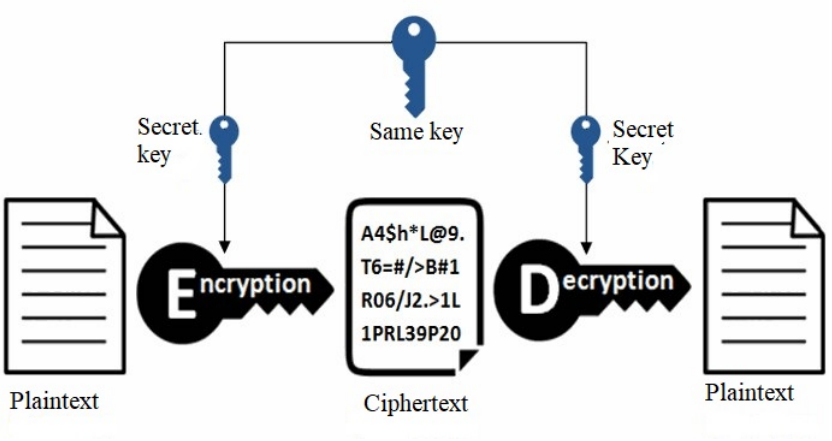

FIGURE 2.5: Symmetric key

2.2.3 Asymmetric key Cryptography

The biggest problem with private key encryption is that you need to have a way to get the key to the party with whom you are sharing data. If someone gets their hands on key, they can decrypt everything encrypted with that key. In 1976, Whitfield Diffie and Martin Hellman introduced a new scheme known as asymmetric key cryptography [6]. Two keys are used in asymmetric key cryptography, where one key is used for data encryption and the other key is used for decryption. A person generates two keys, one is kept secret, called a secret key, and the other key is made public, called the public key. Since the encryption key is public, everyone

can encrypt data, but only the individual with the decryption key can decrypt the data. RSA [7] and DSA [52] are the examples of asymmetric key cryptography.

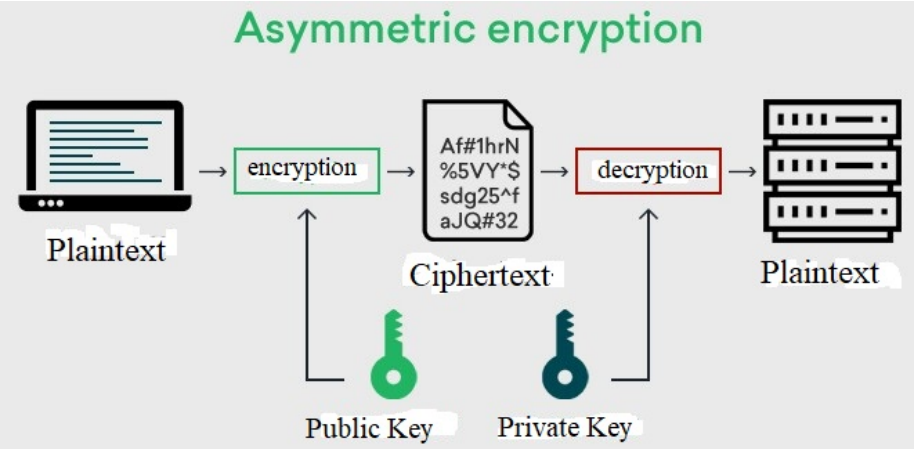

FIGURE 2.6: Asymmetric key

2.2.4 ElGamal Encryption Scheme

The ElGamal encryption scheme [8] in cryptography is an public key encryption algorithm that is based on public-key cryptography [6]. It was proposed in 1985 by Taher Elgamal. A variant of the ElGamal signature scheme is the Digital Signature Algorithm (DSA) [52], which should not be confused with ElGamal encryption. Its security depends on the difficulty associated with computing discrete logarithms of a certain problem in G.

Encryption Algorithm

In this algorithm the domain parameters are (u, v, g) and private/public key pair (b, B) where $B = g^b \mod u$ and encoded message t in the range $0 \leq t \leq u - 1$.

1. Choose an integer m in the range $1 \leq m \leq u - 1$.
2. Compute $c_1 = g^m \mod u$
3. Compute $c_2 = tB^m \mod u$
4. Return ciphertext (c_1, c_2)

Decryption Algorithm

In this algorithm the domain parameters are (u, v, g) with receiever's private key b and ciphertext (c_1, c_2).

1. Compute

$$\begin{aligned} t &= c^{u-b-1}c_2 \mod u \\ &= (g^m)^{u-b-1}.t(g^b)^m \\ &= t[(g^{u-1})^m(g^m)^{-b}](g^m)^b \\ &= t(1)^m(g^m)^{-b}(g^m)^b, \quad since g^{u-1} \equiv 1 \\ &= t.1 \quad since(g^m)^{-b}(g^m)^b = 1 \\ &= t \end{aligned}$$

2. Return t

Example 2.2.1.
The detail example of above algorithm is given below:

Encryption Phase
The domain parameters of this encryption are $u = 283, v = 47, g = 60$, Bob's public key, $B = 216$ and encoded message, $t = 101$, such that $0 \leq t \leq 282$

1. Alice selects a random integere $m = 36$ in the range $[2, 45]$

2. Alice computes $c_1 = g^m \mod u = 60^{36} \mod 283 = 78$.

3. Alice computes $c_2 = tB^m \mod u = (101)(216^{36}) \mod 283 = 218$.

4. Alice sends ciphertext $(c_1, c_2) = (78, 218)$ to Bob

Decryption Phase
The domain parameters of decryption algorithm are $u = 283, v = 47, g = 60$, Bob's private key $b = 7$ and ciphertext $(c_1, c_2) = (78, 218)$

1. Bob computes

$$\begin{aligned} t &= c^{u-b-11}c_2 \mod u \\ &= (78^{283-7-1})(218) \mod 283 \\ &= (116)(218) \mod 283 \\ &= 101 \end{aligned}$$

2. Return $t = 101$

2.3 Cryptanalysis

Cryptanalysis is a technique used to crack the cryptosystem in order to extract plaintext. It is also being analyzed to validate how efficient and stable a cryptosystem is. The person doing cryptanlysis is referred to as a cryptanalyst. Cryptanalysis is necessary if any of the following properties are lacking in a cryptosystem:

1. Confidentiality
2. Integrity
3. Authentication
4. Non-repudiation

There are several kinds of attacks, some of them are listed below:

1. Brute Force Attack

In this attack, an attacker is aware of the ciphertext and the decryption algorithm. The attacker attempts to obtain the plaintext from ciphertext with every key from the set of all possible keys. As the attacker has to search for any key available in the key space, this attack takes a lot of time to accomplish the target. Attempting all possible keys in a reasonable time frame is not feasible, then this attack is not possible. That is, key space should be large enough to counter such attack.

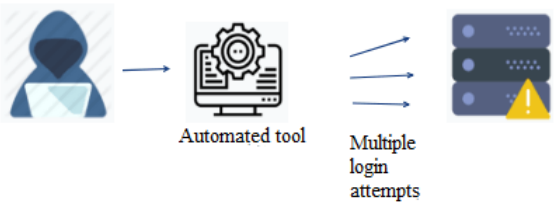

FIGURE 2.7: Brute Force Attack [53]

2. Ciphertexts only Attack

In this attack, the attacker knows only the ciphertext. Normally, the corresponding plaintexts are not known. To obtain the corresponding plaintexts, he utilises these known ciphertexts.

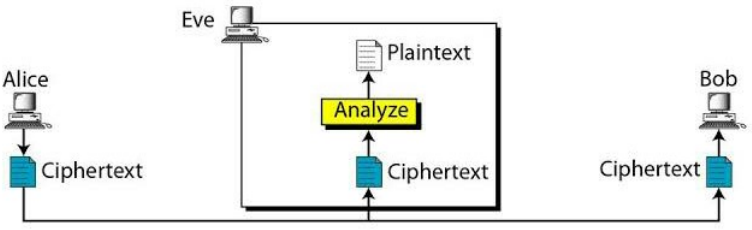

FIGURE 2.8: Ciphertexts only Attack [54]

3. Chosen Ciphertexts Attack

In this attack the attacker has access to some ciphertexts to decrypt and tries to obtain plaintext. He may try to get the key or the plaintexts of other ciphertexts based on this known information.

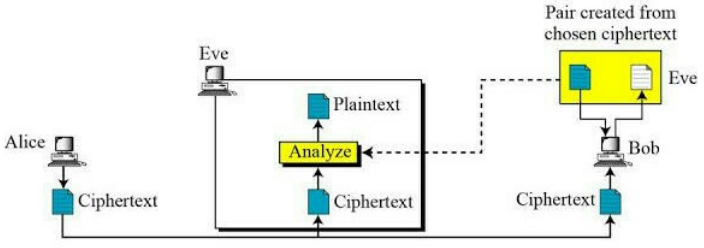

FIGURE 2.9: Chosen Ciphertexts Attack [55]

4. Chosen Plaintext Attack

In this attack, the attacker understands the plaintext and the corresponding ciphertext from which he tries to guess the key or obtains as much information as possible.

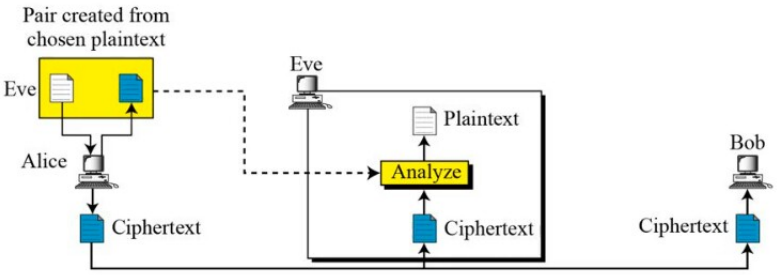

FIGURE 2.10: Chosen Plaintext Attack [56]

5. Known Plaintext Attack

In this attack, there is some portion of plaintext as well as corresponding ciphertext known to the attacker, which is further examined to get the full plaintext or the decyption key.

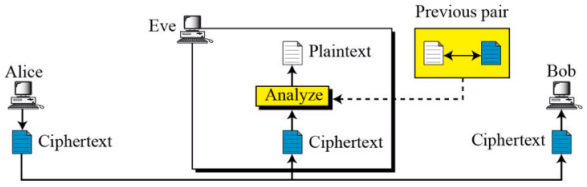

FIGURE 2.11: Known Plaintext Attack [56]

6. Man-in-the-Middle Attack

When two parties attempt to agree on a key for safe communication. In order to agree on a key without understanding them, the attacker positions himself between them. The attacker chooses two keys to deceive the two groups. He uses one of the keys by pretending to be the second party to make the first party agree on the exchange of information. To mislead the second group, the other key is used. In fact , the two sides assume that they are communicating with each other, but it is the attacker who gets the data from both ends and then attacks the conversation.

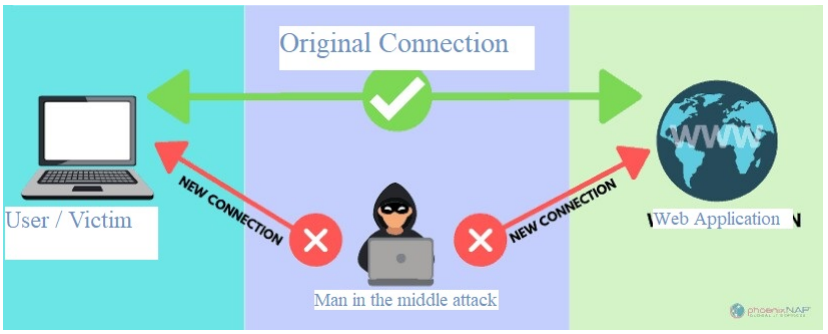

FIGURE 2.12: Man in the Middle Attack [57]

7. Forgery Attack

In this attack, without knowing the secret signing key of the signer the attacker tried to forge a signature for the message. The word 'forgery' generally used to message related attacks in digital signature.

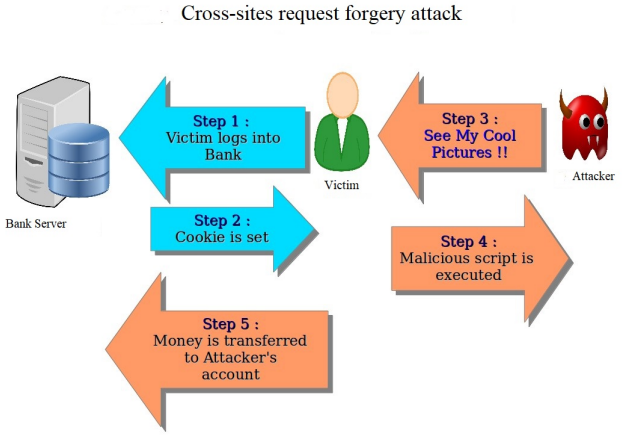

FIGURE 2.13: Forgery Attack [58]

2.4 Elliptic Curve Cryptography

Elliptic curve cryptography (ECC) is a public-key cryptography technique based on the algebraic structure of elliptic curves over finite fields. Over the past few years, ECC has been gaining popularity steadily because of its capacity to provide the same degree of protection. As the demand for devices to remain safe rises due to the increase in the size of keys, drawing on limited mobile resources, this trend will probably continue. That's why understanding ECC in context is so

important. For this reason, ECC is considered to be a public key cryptography's next generation implementation and more stable than RSA. Adopting ECC to ensure high levels of both efficiency and protection also makes sense.

Symmetric	**RSA**	**ECC**
56	512	112
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

TABLE 2.3: Nist recomended key sizes [59]

The major advantage of the use of elliptic curves is that the same security level can be achieved by working in a field of 160 bits and hence elliptic curve solves the problem of computational complexity to achieve the desired security extent. In the next section, elliptic curve will be discussed in detail.

2.4.1 Weierstrass Equation

The equation of the type

$$y^2 + u_1xy + u_3y = x^3 + u_2x^2 + u_4x + u_5$$

defined over a some field $\mathbb{F}$, such as field of real numbers $\mathbb{R}$, complex numbers $\mathbb{C}$ or any finite field $\mathbb{F}_p$ and known as Weierstrass equation [60]. Where $u_1, u_2, u_3, u_4, u_5, u_6$ are called Weierstrass coefficients.

$$-v_2^2v_8 - 8v_4^3 - 27v_6^2 + 9v_2v_4v_6 \neq 0$$

where,

$$v_2 = u_1^2 + 4a_2$$

$$v_4 = 2u_4 + u_1u_3$$

$$v_6 = u_3^2 + 4u_6$$

$$v_8 = u_1^2 u_6 + 4u_2 u_6 - u_1 u_3 u_4 + u_2 u_3^2 - u_4^2$$

2.4.2 Eliptic Curve over $\mathbb{F}_p$

In cryptography generally our focus is on simplified form of Weierstrass equation which is

$$y^2 = x^3 + ax + b \mod p \tag{2.1}$$

Where a and b are Weierstrass coefficients and they are selected from a finite field $\mathbb{F}_p$. The curve is said to be smooth if the discriminant $4a^3 - 27b^2 \neq 0$ and this curve is called elliptic curve.

Point Addition

Suppose we've got two points, U and V, on an elliptic curve E. The following steps must be followed in order to add such points.

1. A straight line is passed from points U and V.
2. At some point, the straight line intersects the curve, say at S of E.
3. Next, we get the point W as the product of U and V. It's only appropriate to take the S negative, which is $-S = (x, -y)$

Point Doubling

Subsequent steps are used to apply a point U to itself,

1. Draw a U-tangent.
2. At some point, it intersects the curve, again regarded as S of E.
3. Next, to get the point $W = 2U$ as the product of adding U to itself, it is only necessary to take S negative.

Point at Infinity

For the addition of U to $-U$, the same method can be used. It is known to us that

$-U$ is essentially a reflection of U. So, it approaches infinity as straight line passes from them. To describe a particular point located at infinity that is recognised as a point towards infinity.

Mathematical Representation

To add a point $U(x_1, y_1)$ and $V(x_2, y_2)$ on elliptic curve (2.1). For graphical structure of point addition, a line must be drawn through them. Let the line pass through U and V, the point slope form of L is:

$$L : y = sx + c$$

For the slope s, the following steps must be followed:

Case 1: If $U \neq V$, then

$$s = \frac{y_2 - y_1}{x_2 - x_1} \tag{2.2}$$

Case 2: If $U = V$, then

$$s = \frac{3x_1^2 + a}{2y_1} \tag{2.3}$$

using basic algebra, the new point say $W(x_3, y_3)$ acquired by adding $U(x_1, y_1)$ and $V(x_2, y_2)$ has the following co-ordinates:

$$x_3 = s^2 - x_1 - x_2 \tag{2.4}$$

$$y_3 = s(x_1 - x_3) - y_1 \tag{2.5}$$

Example 2.4.1.

Let us consider the curve over $\mathbb{F}_{13}$, that is

$$y^2 = x^3 + 7x + 4 \mod 13 \tag{2.6}$$

Table 2.4 shows the points that lie on the curve (2.6). The elliptic curve points addition for $E_{\mathbb{F}_{13}}(7, 4)$ is shown in Table 2.5. Let $U(9, 4)$ and $V(12, 10)$ be two points on eliptic curve (2.6). Then the formulas in (2.4) and (2.5) provide us with

x	y^2	y_1	y_2	U(x,y)	U′(x,y)
0	4	2	11	(0,2)	(0,11)
1	12	5	8	(1,5)	(1,8)
2	0	0	0	(2,0)	-
3	0	0	0	(3,0)	-
4	5	-	-	-	-
5	8	-	-	-	-
6	2	-	-	-	-
7	6	-	-	-	-
8	0	0	0	(8,0)	-
9	3	4	9	(9,4)	(9,9)
10	8	-	-	-	-
11	8	-	-	-	-
12	9	3	10	(12,3)	(12,10)

TABLE 2.4: Eliptic curve points addition

the new point $W(x_3, y_3)$. Evaluate the slope s by

$$\begin{aligned} s &= \frac{10-4}{12-9} \mod 13 \\ &= \frac{6}{3} \mod 13 \\ &= 6(3^{-1}) \mod 13 \end{aligned}$$

By using Extended Euclidean algorithm,

$$\begin{aligned} s &= (6)(9) \mod 13 \\ s &= 2 \mod 13 \end{aligned}$$

Put the value of s in (2.4) and (2.5), gives us:

$$\begin{aligned} x_3 &= (2)^2 - 9 - 12 \mod 13 \\ x_3 &= 9 \mod 13 \\ y_3 &= 2(9-9) - 4 \mod 13 \\ y_3 &= 9 \mod 13 \end{aligned}$$

so, $W = (x_3, y_3) = (9, 9)$ is the addition of points. Now, let us add a point $U(9, 4)$

into itself. To compute (2.2) as:

$$
\begin{aligned}
s &= \frac{3(9)^2 + 7}{2(4)} \mod 13 \\
s &= 250 \times 8^{-1} \mod 13 \\
s &= 250 \times 5 \mod 13 \\
s &= 2
\end{aligned}
$$

Put the value of s in (2.4) and (2.5), gives us:

$$
\begin{aligned}
x_3 &= (2)^2 - 2(9) \mod 13 \\
x_3 &= 12 \\
y_3 &= 2(9 - 12) - 4 \mod 13 \\
y_3 &= 3
\end{aligned}
$$

so, $W = (x_3, y_3) = (12, 3)$

+	∞	**(0,2)**	**(0,11)**	**(1,5)**	**(1,8)**	**(2,0)**	**(3,0)**	**(8,0)**	**(9,4)**	**(9,9)**	**(12,3)**	**(12,10)**
∞	∞	(0,2)	(0,11)	(1,5)	(1,8)	(2,0)	(3,0)	(8,0)	(9,4)	(9,9)	(12,3)	(12,10)
(0,2)	(0,2)	(12,3)	∞	(8,0)	(9,9)	(12,10)	(9,4)	(1,8)	(1,5)	(3,0)	(2,0)	(0,11)
(0,11)	(0,11)	∞	(12,10)	(9,4)	(8,0)	(12,3)	(9,9)	(1,5)	(3,0)	(1,8)	(0,2)	(2,0)
(1,5)	(1,5)	(8,0)	(9,4)	(12,10)	∞	(9,9)	(12,3)	(0,11)	(2,0)	(0,2)	(1,8)	(3,0)
(1,8)	(1,8)	(9,9)	(8,0)	∞	(12,3)	(9,4)	(12,10)	(0,2)	(0,11)	(2,0)	(3,0)	(1,5)
(2,0)	(2,0)	(12,10)	(12,3)	(9,9)	(9,4)	∞	(8,0)	(3,0)	(1,8)	(1,5)	(0,11)	(0,2)
(3,0)	(3,0)	(9,4)	(9,9)	(12,3)	(12,10)	(8,0)	∞	(2,0)	(0,2)	(0,11)	(1,5)	(1,8)
8,0	(8,0)	(1,8)	(1,5)	(0,11)	(0,2)	(3,0)	(2,0)	∞	(12,10)	(12,3)	(9,9)	(9,4)
(9,4)	(9,4)	(1,5)	(3,0)	(2,0)	(0,11)	(1,8)	(0,2)	(12,10)	(12,3)	∞	(8,0)	(9,9)
(9,9)	(9,9)	(3,0)	(1,8)	(0,2)	(2,0)	(1,5)	(0,11)	(12,3)	∞	(12,10)	(9,4)	(8,0)
(12,3)	(12,3)	(2,0)	(0,2)	(1,8)	(3,0)	(0,11)	(1,5)	(9,9)	(8,0)	(9,4)	(12,10)	∞
(12,10)	(12,10)	(0,11)	(2,0)	(3,0)	(1,5)	(0,2)	(1,8)	(9,4)	(9,9)	(8,0)	∞	(12,3)

TABLE 2.5: Addition of points of $E_{\mathbb{F}_{13}}(7, 4)$

2.4.3 Elliptic Curve Discrete Logarithm Problem

The elliptic curve discrete logarithm problem (ECDLP) is finding the number k such that $kP = Q$ given two points P and Q. Discrete logarithm of Q to the base P is the term given to the number k. It's impossible to come across k. ECDLP is responsible for ECC's complete security.

2.4.4 Diffie-Hellman Key Exchange Based for Eliptic Curve Group

Alice and Bob need to share their keys so they can encrypt and decrypt the messages in order to communicate in a secure fashion. In 1976, the idea was given by Diffie and Hellman [6] to exchange keys over a public network without compromising security. The scheme is designed with the help of a cyclic group of elliptic curve points and safety relies on the complexity of overcoming ECDLP. The following approach tells the whole story of Alice and Bob exchanging keys using Diffie-Hellman key exchange protocol.

1. Alice and Bob mutually selects an elliptic curve E over a finite field $\mathbb{F}_q$ with G is the base point of an eliptic curve E.
2. The random number $d_A \in \{1, 2, 3, \ldots n-1\}$ is selected by Alice as her secret key and compute her public key as $P_A = d_A G$.
3. Bob selects his private key $d_B \in \{1, 2, 3, \ldots n-1\}$ and calculates his public key $P_B = d_B G$.
4. They both exchange their public keys P_A and P_B with each other.
5. Alice computed $P_{AB} = d_B d_A$ where P_{AB} is used to find d_A and d_B as session key security.

Example 2.4.2. Let sender A wants to send a message $m = 21$ to recipient B. So, they must share their keys to encrypt and decrypt a message. Sender A and recipient B mutually selects an elliptic curve $y^2 = x^3 + 750x + 188 \mod 751$.

$G = (0, 376)$ is the base point of order $n = 727$ where $727(0, 376) = \mathcal{O}$. The elliptic curve has also 727 number of points.

1. Sender A selects private key $d_A = 12$ and compute public key as $P_A = d_A \cdot G = 12(0, 376) = (207, 215)$.
2. Recipient B selects private key $d_B = 17$ and compute public key as $P_B = d_B \cdot G = 17(0, 376) = (556, 631)$.
3. They both exchange their $P_A = (207, 215)$ and $P_B = (556, 631)$ with each other.
4. Sender A computed $P_{AB} = d_B d_A = 17(12) = 204$ is used to find $d_A = 12$ and $d_B = 17$ as session key security.

2.5 Eliptic Curve Encryption Decryption

Elliptic curve is a technique of asymmetric key cryptography. For secure communication, each user have its own public and secret key.

2.5.1 Global Settings

These global parameters that involved in communication between sender (Alice) and receiver (Bob).

1. The base point G such that $nG = \mathcal{O}$. where n is the smallest prime number and $\mathcal{O}$ is point at infinity.
2. A prime integer modulo q and constants u and v.

2.5.2 Key Generation Phase

1. Alice selects secret key d_A from the set $\{1, 2, \ldots n-1\}$ and compute public key as $P_A = d_A.G$
2. Bob selects his private key $d_B < n$ and compute his public key as $P_B = d_B.G$

2.5.3 Encryption Phase

Alice send a message t to Bob by using ECC scheme. For this t is converted into a elliptic curve point Q_t. Alice choose a random integer k and calculate the ciphertext Q_C as the elliptic curve pair of points using Bob's public key P_B as follows.

$$Q_C = (kG, Q_t + kP_B) \mod p$$

2.5.4 Decryption Phase

After receiving ciphertext Q_C, Bob decrypt the message back into original form by multiplying kG with private key of Bob d_B and than add the result into second ciphertext pair $Q_t + kP_B$

$$\begin{aligned} Q_t + kP_B - d_B(kG) &= Q_t + kd_BG - kd_BG \\ &= Q_t \end{aligned}$$

which is plaintext point, corresponding to plaintext message t.

Example 2.5.1.
Let us consider an elliptic curve $y^2 = x^3 - x + 188 \mod 751$. Let $G = (0, 376)$ be the base point. Total number of points and order of this curve is 727 where $727(0, 376) = \mathcal{O}$. Let Alice wants to send a message Q_t to Bob by using ECC encryption. Alice selects her secret key $d_A = 6$ and compute public key as $P_A = 6(0, 376) = (6, 390)$ Bob selects his secret key $d_B = 5$ and compute public key as $P_B = 5(0, 376) = (188, 657)$. Alice chooses the secret random number $k = 113$ to encrypt the original message $Q_t = (443, 253)$.

$$\begin{aligned} Q_C &= [kG, Q_t + kP_B] \mod p \\ &= [113(0, 376), (443, 253) + 113(188, 657)] \mod 751 \\ &= [(34, 633), (443, 253) + (529, 254)] \mod 751 \\ &= [(34, 633), (418, 18)] \end{aligned}$$

Alice sends $Q_C = [(34, 633), (418, 18)]$ to Bob. After receiving Q_C Bob decrypt it to get Q_t.

$$\begin{aligned} Q_t &= [Q_t + kP_B - d_B(kG)] \mod p \\ &= [(443, 253) + 113(188, 657) - 5(34, 633)] \mod 751 \\ &= [(443, 253) + (529, 254) - (529, 254)] \mod 751 \\ &= (443, 253) \end{aligned}$$

Chapter 3

Digital Signcryption

The way people perform safe and authenticated communications has been revolutionized by public key cryptography [6], discovered almost two decades ago. It is now possible for individuals who have never been interacted before to connect with each other through both open and inaccessible networks such as the internet in a safe and authenticated manner. In doing so, the same two-step technique has been implemented. The sender of the message will sign it with a digital signature scheme, and then encrypt the message using a private key encryption algorithm under a randomly selected message encryption key, before a message is sent out. Signature generation and encryption consume machine cycles, and add an original message with "expanded" bits as well. The cost of a cryptographic operation on a message is thus usually measured in the expansion rate of the message and in the computing time expended by both the sender and the receiver. The cost of sending a message in a safe and anthenticated manner using the existing traditional signature-then-encryption is approximately the sum of the cost for digital signature and that for encryption [15].

The question about the cost of delivery of secure and authenticated messages, namely whether it is possible to transfer a message of arbitrary length in a secure and authenticated manner at an expense lower than that required by signature-then-encryption. Since the invention of public key cryptography, this topic appears to have never been discussed in literature [28]. In 1997, Yuliang Zheng [10] descovers a new primitive called signcryption which executes the functions of both digital

signature and encryption at the same time, and it costs significally less than the required traditionally signature then encryption scheme. It will effectively eliminate computing costs and communication overheads [15]. Signcryption includes both digital signatures and encryption mechanisms with properties in a manner that is more suitable than independent signing and encryption. This means that, under a specific security paradigm, at least certain dimensions of its reliability (such as computing time) are better than any combination of digital signature and encryption schemes [15]. Over the years, there are so many signcryption schemes that have been suggested, each with their own challenges and drawbacks, while delivering varying standards of security and computational costs.

3.1 Features of Signcryption Scheme

There are usually three algorithms in a signcryption scheme: Key Generation **(Gen)**, Signcryption **(SC)**, and Unsigncryption **(USC)**. A typicall signcryption scheme provides the following properties:

1. **Correctness:** Every signcryption scheme should be provable accurate.

2. **Efficiency**: In any signcryption scheme computing cost and communication overhead should be lower than best signature then encryption schemes with the same offered features.

3. **Security:** The security characteristics of an encryption scheme and a digital signature should be met simultaneously by a signcryption scheme. These additional features primarily include: confidentiality, unforgeability, integrity, and non-repudiation. Other features such as public authentication and forward secrecy of message confidentiality are offered by some signcryption schemes. The security of such features normally relies on the underlying hard problem. For instance, the elliptic curve discrete logarithm problem in the scheme is based on ECC.

3.2 Zheng's Signcryption Scheme

Zheng [10] discovers a new cryptographic premitive called "signcryption" which fulfills both the functions of digital signature and encryption in a logically single step. Its cost is significantly lower than that needed by signature-then-encryption technique. In the following section, we take a brief look of this scheme.

Notations

- t : Original message
- u : A large prime number
- v : A large prime factor of $u - 1$
- x : A randomly chosen number from $\{1, 2, \ldots v - 1\}$
- h : One way hash function to get 128-bit hash values
- H_k : Keyed hash function
- E : A private key encryption algorithm
- D : A private key decryption algorithm

Key Generation

1. Alice selects private key d_A from the range $\{1, 2, \ldots u - 1\}$ and compute public key $P_A = x^{d_A} \mod u$.
2. Bob selects private key d_B from the range $\{1, 2, \ldots u-1\}$ and compute public key $P_B = x^{d_B} \mod u$.

Algorithm 3.2.1. (Signcryption)
Input: (t, d_A, P_B)
Output: (c, s)

1. Alice selects an integer $x \in \{1, 2, \ldots u - 1\}$.

2. She use public key of Bob P_B, the integer x and one way hash function h to compute

$$z = h(P_B x) \mod u$$

3. She divides 128 bit value into two parts of 64 bits. They can be numbered z_1 and z_2.

4. Alice encrypts the message t by using the public key encryption scheme E with the key z_1. It will give ciphertext

$$c = Ez_1(t)$$

5. She uses the key z_2 message t and one way keyed hash H_k value to compute r.

$$r = H_k z_2(t)$$

6. The signature parameter s is computed by Alice. By using x, the secret key d_A, the large prime number v and r to get

$$s = \frac{x}{r + d_A} \mod v$$

7. The values c, r and s are now available to Alice. In order to complete the task, she send these values to Bob.

Algorithm 3.2.2. (Unsigncryption)
Input: (d_B, P_A, c, s)
Output: t

1. Bob recieves the values c, r and s from Alice. It uses the values r and s, his secret key d_B, Alice's public key P_A, and u to calculate a hash value of 128 bits.

$$z = h(P_A.x^r)^s d_B \mod u$$

Then 128 bit hash value is divided into two 64 bit pieces that give him (z_1, z_2). This key pair is the same as the key pair generated on Alice side.

2. For decryption of ciphertext c, Bob uses the key z_1, which will give him the message t.

$$t = D_{z_1}(c)$$

3. Bob will verify the evaluation

$$r = H_k z_2(t)$$

It means that the message t was actually signed and sent by Alice, if its match. Unless Bob understands that either Alice did not sign the message or an attacker intercepted and changed it.

3.3 An Efficient and Authentication Signcryption Scheme Based on Elliptic Curves

In this section, "An efficient and authentication signcryption scheme based on elliptic curves" by Kumar and Gupta [43] will be reviewed. Before describing this scheme, some notations which are very helpful to understand this scheme are presented.

Notations

- G : Elliptic curve base point
- n : Order of base point G where $n \cdot G = \mathcal{O}$
- h : One way hash function
- E_k : Symmetric Encryption using key k
- D_k : Decryption algorithm using key k
- m : message
- c : ciphetext

Key Generation

Alice (Sender)

1. Alice selects her secret key d_A from the set $\{1, 2, \ldots, n-1\}$.
2. Alice compute her public key as $P_A = d_A \cdot G$.

Bob (Receiver)

1. Bob selects his secret key d_B from the set $\{1, 2, \ldots, n-1\}$.
2. Bob compute his public key as $P_B = d_B \cdot G$.

Algorithm 3.3.1. (Signcryption)
Input: (d_A, P_B, m)
Output: (c, s, R)

1. Select random integer $k \in \{1, 2, \ldots, n-1\}$.
2. Using the integer k calculate the elliptic curve point $K = k \cdot P_B = (k_1, k_2)$ mod p.
3. Compute $c = E_{k_1}(m \parallel E_{d_A}(h(m)))$.
4. Compute $r = h(c, k_2)$.
5. Compute $s = \frac{k}{r+d_A} \mod n$.
6. Compute $R = r \cdot G$.
7. Send message (c, s, R).

Algorithm 3.3.2. (Unsigncryption)
Input: (d_B, P_A, c, s, R)
Output: (K, m)

1. Calculate $K = d_B \cdot s \cdot R + d_B \cdot s \cdot P_A = (k_1, k_2)$.

2. Calculate $r' = h(c, k_2)$.

3. Calculate $m = D_{k_1}(c)$.

4. Accecpt the ciphertext as valid if $R = r' \cdot P_A$.

3.3.1 Correctness

This scheme is devoted to the proof of the correctness of the scheme.

Theorem 3.1. *Message decryption is valid if the receiver confirms the following equation* $K = d_B \cdot s \cdot (R + P_A) \mod n$.

Proof:

$$\begin{aligned} d_B \cdot s \cdot (R + P_A) &= d_B \cdot \frac{k}{r + d_A}(r \cdot G + d_A \cdot G) \mod n \\ &= d_B \cdot \frac{k \cdot G}{r + d_A}(r + d_A) \\ &= k \cdot (d_B \cdot G) \\ &= k \cdot P_B \\ &= K \end{aligned}$$

3.3.2 Security Analysis

The following are the components of the security analysis to be considered.

1. Confidentiality: Confidentiality refers to the process of protecting the content of a communication against unauthorized access. If an unauthorized person wishes to deduce the secret key k_1 of Step (2) of Signcryption Algorithm 3.3.1. He must required to solve ECDLP, which is impossible to solve.

2. Authentication: The scheme provides authentication by including the certificate authority in the verification of both the receiver's and sender's public keys. To check the validity of received messages, the receiver utilises the sender's public key P_A.

3. Integrity: Integrity is the process of ensuring that data is not altered by unauthorised individuals while in transit. If the message content is altered, the ciphertext C is replaced with C', and the associated message m is replaced with m' in Step (3) of 3.3.1. It is computationally infeasible due to the one-way hash function's characteristic. When the message is verified, this modification is discovered, and the message is denied. As a result, the other message's integrity is validated.

4. Unforgeability: Dishonest Bob is the most potent attacker in this technique to fabricate a signcrypted message, because he is the only person who knows the private key d_B, which is necessary to directly verify a signcryption from Alice. A signcrypted text (c, s, R) is provided. Bob may decrypt the cipher string c using his private key d_B and acquire (m, s, R). ECDSA is unforgeable against adaptive attack. As a result, it is unforgeable.

5. Non-repudiation: Non-repudiation is the guarantee that someone cannot refute anything. In this scenario, if the sender denies that the communication was sent, the recipient might submit (R, s, c) requested by the judge to verify. If equation $(k_1, k_2) = s - d_B R$ holds during the judge verification phase, the judge can decide that the signature was produced by the sender. The property of non-repudiation is therefore ensured.

6. Forward secrecy: If an opponent gets d_A, he or she will be unable to decode previous communications. Prior to the breach, previously recorded values of (c, s, R) cannot be decrypted since the adversary using d_A will need to calculate d_B to decode. Solving the ECDLP, which is computationally infeasible, is required to calculate d_B.

7. Public verification: Only Alice's public key is required for verification. Every system user is supposed to have access to all public keys through a certifying authority or a public directory. An interactive zero-knowledge key exchange protocol is required for the proposed scheme. This attribute is very essential for security.

3.4 Generalized Signcryption

For a secure message delivery not all messages need both confidentiality and authenticity. Some messages may just require signatures, while others may only require encryption. In 2006, Han and Yang [30] proposed a scheme which provides the flexibility of signcryption scheme, encryption scheme, signature scheme as required and this scheme is known as generalized signcryption. When both confidentiality and authenticity are required at the same time, this scheme performs double functions and when just confidentiality and authenticity are required, it performs a single encryption/signature function without any modifications or additional calculation. In specific circumstances, a generalized signcryption scheme will be equal to a signature or encryption scheme. There are three possible scenarios: (i) signcryption (ii) signature-only (iii) encryption-only. The challenge of identifying the three situations is a significant one. Performing the authentication process in a public key environment necessitates the knowledge of the sender's public and private keys. The encryption procedure necessitates the knowledge of a specific recipient (public key and private key).

3.4.1 Elliptic Curve Based Generalized Signcryption Scheme

In this section, elliptic curve based generalized signcryption scheme (ECGSC) will be discussed. It is the first generalized signcryption scheme proposed by Han and Yang [30].

Global Parameters

The scheme parameters are given below.

1. G is the base point of order n.
2. $P = x'G$ represents the scalar multiplex.
3. The term *parallel* refers to the connection of two messages.

4. $\in T$ indicates selecting an element from a set.

5. Bind refers to the identity of Alice and Bob.

6. $(0,1)^v$ indicates the binary sequence of length v.

7. $M_{enc}, M_{mac}, M_{sign}$ is a binary sequence.

8. $I : (0,1)^* \to \mathbb{Z}_q^*$ and $M : \mathbb{Z}_q^* \to (0,1)^{\mathbb{Z}+*}$ denotes two hash functions.

9. $LI(.) : (0,1)^* \to (0,1)^{v+z}$ are long digest hash functions.

10. $NBD_{k'} : (0,1)^v \times (0,1)^u \times (0,1)^z$ denotes message authenticate function which has key k'. $\mid k' \mid = u, \mid n \mid = v, v+ \mid NBD(.) \mid = \mid LI(x_2') \mid$ -These hash functions have property $I(0) \to 0, M(0) \to 0, LI(0) \to 0, NBD(0) \to 0$

Key Generation

1. Alice selects private key d_A from the range $\{1, 2, \ldots n-1\}$ and compute public key as $P_A = d_A.G$.

2. Bob selects private key d_B from the range $\{1, 2, \ldots n-1\}$ and compute public key as $P_B = d_B.G$.

Algorithm 3.4.1. (Generalized Signcryption)
Input: (m, d_A, P_B)
Output: w

1. $k' \in_T \{1, 2, ..., n-1\}$

2. $T = k'G = (x_1', y_1'), r' = x_1' \mod q$

3. $k'.P_B = (x_2', y_2')$

4. $M_{enc} = LI(x_2'), (M_{mac}, M_{sign}) = M(y_2')$

5. If $d_A = 0, s = 0$, Else $s = k'^{-1}(I(n \parallel Bind \parallel M_{sig}) + r'd_A) \mod n$

6. $f = NBD_{M_{mac}}(m)$

7. $c = (m \parallel f) \bigoplus M_{enc}$

8. Return $z = (c, T, s)$.

Algorithm 3.4.2. (Generalized Unsigncryption)

Input: (d_B, P_A, z)

Output: m

1. $r = x'(T)(T's\ x'$ axiom).

2. $(x'_2, y'_2) = d_B T$

3. $M_{enc} = LI(x'_2), (M_{mac}, M_{sig}) = M(y'_2)$

4. $(m \parallel f) = c \bigoplus M_{enc}$

5. $f' = NBD_{M_{mac}}(m)$, If $f \neq f'$, return $\perp$ else if $s = 0$, return m.

6. $t_1 = s^{-1}I(m \parallel Bind \parallel M_{sig}), t_2 = s^{-1}r'$

7. $T' = t_1 G + t_2 P_A$, If $T' \neq T$, return $\perp$,else return m.

Signature only Mode

ECGSC scheme will become ECDSA scheme when $d_B = 0, P_B = 0$.
SC$(m, d_A, 0)$

1. $k' \in_T \{1, 2, ..., n-1\}$

2. $T = k'G = (x'_1, y'_1), r' = x'_1 \mod q$

3. $s = (k')^{-1}(I(m) + r'd_A) \mod n$

4. $m = (m \parallel 0) \bigoplus 0$

5. Return $z = (m, T, s)$.

Verification

Any receiver can verify as follows:
$DSC(z, 0, P_A)$

1. $(m \parallel 0) = m \bigoplus M_{enc}$

2. $t_1 = s^{-1}I(m), t_2 = s^{-1}r'$

3. $T' = t_1G + t_2P_A$,

4. If $T' \neq T$, return $\perp$.

Encryption only Mode

ECGSC scheme will become encryption scheme when $d_A = 0, P_A = 0$.
$\mathrm{SC}(m, 0, P_B)$

1. $k' \in_T \{1, 2, ..., n-1\}$

2. $T = k'G = (x'_1, y'_1), r' = x'_1 \mod q$

3. $k'.P_B = (x'_2, y'_2)$

4. $M_{enc} = LI(x'_2), (M_{mac}, M_{sign}) = M(y'_2)$

5. $f = NBD_{M_{mac}}(m)$

6. $c = (m \parallel f) \bigoplus M_{enc}$

7. Return $z = (c, T)$.

Decryption

To get the plaintext m the receiver runs the decryption algorithm.
$DSC(z, d_B, 0)$

1. $(x'_2, y'_2) = d_B T$
2. $M_{enc} = LI(x'_2), (M_{mac}, M_{sig}) = M(y'_2)$
3. $(m \parallel f) = c \bigoplus M_{enc}$
4. $f' = NBD_{M_{mac}}(m)$
5. If $f \neq f'$, return $\perp$.

3.4.2 Correctness

Theorem 3.2. ***(Encryption only Mode)***
The message decryption is valid if receiver confirms the following equation $k'P_B = d_B T$.

Proof:

$$d_B T = d_B(k'G) = k'(d_B G) = k'P_B$$

Theorem 3.3. ***(Signature only Mode)***
If the receiver confirms the following equation $T = t_1 G + t_2 P_A$ *then the signature is valid.*

Proof: *Consider the equation* $s^{-1} = k'(I(n \parallel Bind \parallel M_{sig}) + r'd_A)^{-1}$.
Let $h = (I(n \parallel Bind \parallel M_{sig})) \Rightarrow s^{-1} = k(h + r'd_A)^{-1}$ *then* t_1 *and* t_2 *becomes*

$t_1 = k'(h + r'd_A)^{-1}h$ *and* $t_2 = k'(h + r'd_A)^{-1}r'$

Now consider the equation

$$\begin{aligned} t_1G + t_2P_A &= t_1G + t_2d_AG \\ &= G(t_1 + t_2d_A) \\ &= G(k'h(h + r'd_A)^{-1} + k'r'd_A(h + r'd_A)^{-1}) \\ &= k'G(h + r'd_A)(h + r'd_A)^{-1} \\ &= k'G \\ &= T \end{aligned}$$

Chapter 4

A New Generalized Signcryption Scheme Based on Elliptic Curves

In this chapter, a proposed generalized signcryption scheme will be discussed. The proposed scheme is the extension of Kumar and Gupta's scheme [43]. Later, a toy example will be discussed for better understanding of this scheme.

4.1 The Proposed Generalized Signcryption Scheme

The signcryption scheme of Kumar and Gupta [43] works effectively when both confidentiality and authenticity are required. The proposed scheme provides extra features, it works in signcryption only mode if both confidentiality and authenticity are required and works in encryption only mode or signature only mode if one of them is required. The following steps described this scheme.

Global Settings

The golabal parameters are presented to the participants Alice and Bob. These parameters are shown in Table 4.1.

p	A large prime number greater than 2^{160}
$\mathbb{F}_p$	Finite field of order p
$E_p(a,b)$	Elliptic curve defined on $\mathbb{F}_p$
G	A base point of order n where $nG = \mathcal{O}$
h	A one way hash function
E_k	Symmetric Encryption using key k
D_k	Decryption algorithm using key k

TABLE 4.1: Global parameters

Key Generation

Alice (Sender)

1. Alice selects her secret key d_A from the set $\{1, 2, \ldots, n-1\}$.
2. Alice compute her public key as $P_A = d_A \cdot G \mod p$.

Bob (Receiver)

1. Bob selects his secret key d_B from the set $\{1, 2, \ldots, n-1\}$.
2. Bob compute his public key as $P_B = d_B \cdot G \mod p$.

Algorithm 4.1.1. (Generalized Signcryption)
Input: (d_A, P_B, m)
Output: (R, c, s, Q)

1. Select random integers $r_1, r_2 \in \{1, 2, ..., n-1\}$.
2. Using the integer r_1 calculate the elliptic curve point

$$T = r_1 \cdot P_B = (T_1, T_2) \mod p$$

3. Using the integer r_2 calculate the elliptic curve point $R = r_2 \cdot G \mod p$.

4. Using private key of Alice d_A and public key of Bob P_B compute

$$k = d_A \cdot P_B = (k_1, k_2) \mod p$$

5. Using key k_1 and R compute the elliptic curve point $k^* = k_1 \cdot R = (k_3, k_4)$ mod p.

6. Using symmetric key encryption, encrypt the plaintext message m into ciphertext c by using k_3 as $c = E_{k_3}(m) \mod n$.

7. By using one way hash function calculate $r = h(c, T_2) \mod n$.

8. Compute the digital signature $s = \frac{r_1}{r+d_A} \mod n$.

9. Compute $Q = r \cdot G$.

10. Send message (R, c, s, Q) to Bob.

Algorithm 4.1.2. (Generalized Unsigncryption)
Input: (d_B, P_A, R, c, s, Q)
Output: (m, T)

1. Validate the Alice public key P_A by using Bob's certificate.

2. Using private key of Bob d_B and public key of Alice P_A regenerate

$$k = d_B \cdot P_A = (k_1, k_2) \mod p$$

3. Using key k_1 and R compute the elliptic curve point $k^* = k_1 \cdot R = (k_3, k_4)$ mod p.

4. Using symmetric key encryption, decrypt the ciphertext message c into plaintext by using key k_3 as $m = D_{k_3}(c) \mod n$.

5. For the verification of digital signature s, calculate

$$T = d_B \cdot s \cdot (Q + P_A) = (T_1, T_2) \mod n$$

6. Calculate $r' = h(c, T_2) \mod n$.

7. Accept the ciphertext as valid if $r = r'$ otherwise reject.

The different steps of signcryption are also explained in Table 4.4.

Signcryption	**Unsigncryption**
$T = r_1 \cdot P_B = (T_1, T_2)$	$k = d_B \cdot P_A = (k_1, k_2)$
$R = r_2 \cdot G$	$k^* = k_1 \cdot R = (k_3, k_4)$
$k = d_A \cdot P_B = (k_1, k_2)$	$m = D_{k_3}(c)$
$k^* = k_1 \cdot R = (k_3, k_4)$	$T = d_B \cdot s \cdot (Q + P_A) = (T_1, T_2)$
$c = E_{k_3}(m)$	$r' = h(c, T_2)$
$r = h(c, T_2)$	Accept the ciphertext if $r = r'$
$s = \frac{r_1}{r+d_A} \mod n$	
$Q = r \cdot G \mod p$	
Send (R, c, s, Q)	

TABLE 4.2: Generalized Signcryption

Signature only Mode

Setting $r_2 = 0$ the proposed scheme will become ECDSA scheme.

1. Select random integer $r_1 \in \{1, 2, ..., n-1\}$.

2. Using the integer r_1 compute $T = r_1 P_B = (T_1, T_2) \mod p$.

3. Get $c = m$.

4. Compute the hash value $r = h(c, T_2) \mod n$.

5. $s = \frac{r_1}{r+d_A} \mod n$.

6. Calculate $Q = rG \mod p$.

7. Send text (s, Q) to Bob.

Verification

Receiver can verify the content of received message (s, Q).

1. Using Bob's certificate to validate the sender's public key.
2. Calculate $T = d_B s(Q + P_A) = (T_1, T_2) \mod n$.
3. By using one way hash function, calculate $r' = h(c, T_2) \mod n$.
4. Accept ciphertext as valid if $r = r'$.

Signature	**Verification**
$T = r_1 \cdot P_B = (T_1, T_2) \mod p$	$T = d_B \cdot s \cdot (Q + P_A) = (T_1, T_2) \mod n$
$r = h(c, T_2) \mod n$	$r' = h(c, T_2) \mod n$
$s = \frac{r_1}{r + d_A} \mod n$	Accept if $r = r'$
$Q = r \cdot G \mod p$	

TABLE 4.3: Signature only mode

Encryption only Mode

Setting $r_1 = 0$ then the proposed scheme will become encryption scheme.

1. Select a random integer $r_2 \in \{1, 2, ..., n-1\}$.
2. Compute $R = r_2 \cdot G \mod p$.
3. Compute $k = d_A \cdot P_B = (k_1, k_2) \mod p$.
4. Compute $k^* = k_1 \cdot R = (k_3, k_4) \mod p$.
5. Compute $c = E_{k_3}(m) \mod n$.
6. Send text (c, R) to Bob.

Decryption

To get the plaintext message m the receiver runs the decryption algorithm.

1. Using Bob's certificate to validate the sender's public key.
2. Regenerate the key $k = d_B \cdot P_A = (k_1, k_2) \mod p$.
3. Compute the key $k^* = k_1 \cdot R = (k_3, k_4) \mod p$.
4. Compute $m = D_{k_3}(c) \mod n$.

Encryption	**Decryption**
$R = r_2 \cdot G \mod p$	$k = d_B \cdot P_A = (k_1, k_2) \mod p$
$k = d_A \cdot P_B = (k_1, k_2) \mod p$	$k^* = k_1 \cdot R = (k_3, k_4) \mod p$
$k^* = k_1 \cdot R = (k_3, k_4) \mod p$	$m = D_{k_3}(c) \mod n$
$c = E_{k_3}(m) \mod n$	

TABLE 4.4: Encryption only mode

4.1.1 Correctness

The correctness of the scheme described in the following theorems.

Theorem 4.1. ***(Signature only Mode)***

If the receiver confirms the following equation than the signautre is valid.

$$T = r_1 \cdot P_B \mod p$$

Proof:

$$\begin{aligned} T &= d_B \cdot s \cdot (Q + P_A) \\ &= d_B \cdot \frac{r_1}{r + d_A} \cdot (r \cdot G + d_A \cdot G) \\ &= d_B \cdot \frac{r_1 \cdot G}{r + d_A}(r + d_A) \\ &= r_1 \cdot (d_B \cdot G) \\ &= r_1 \cdot P_B \end{aligned}$$

Theorem 4.2. ***(Encryption only Mode)***

If the receiver confirms the following equation than message decryption is valid.

$$d_B \cdot P_A = d_A \cdot P_B \mod p$$

Proof:

$$\begin{aligned} d_B \cdot P_A &= d_B \cdot d_A \cdot G \\ &= d_A \cdot (d_B \cdot G) \\ &= d_A \cdot P_B \end{aligned}$$

4.2 A toy Example

In this section, a toy example is given to illustrate how a message is signcrypted using above proposed scheme.

Example 4.2.1.

Alice want to send a message $m = 15$ to Bob in a confidential and authenticated manner. For this consider an eliptic curve $y^2 = x^3 + 750x + 188 \mod 751$ where $a = 750$, $b = 188$ and $p = 751$. The elliptic curve group generated by $E_p(a, b) = E_{751}(750, 188)$.

Let $G = (0, 376)$ be the base point of elliptic curve and $n = 727$ be the order of base point G where $727(0, 376) = \mathcal{O}$. The total number of points of this elliptic curve is also 727. The following steps must be performed for signcrypting a message.

Key Generation Phase

1. Alice choose her private key $d_A = 10$ and compute public key as

$$P_A = d_A \cdot G = 10(0, 376) = (57, 332) \mod 751$$

2. Bob choose his private key $d_B = 2$ and compute public key as

$$P_B = d_B \cdot G = 2(0, 376) = (1, 376) \mod 751$$

Generalized Signcryption

1. Select randomly integers $r_1 = 3$ and $r_2 = 4$.

2. Compute $T = r_1 \cdot P_B = 3(1, 376) = (6, 390) = (T_1, T_2) \mod 751$.

3. Compute $R = r_2 \cdot G = 4(0, 376) = (2, 373) \mod 751$.

4. Compute $k = d_A \cdot P_B = 10(1, 376) = (731, 529) = (k_1, k_2) \mod 751$.

5. Compute $k^* = k_1 \cdot R = 731(2, 373) = (197, 107) = (k_3, k_4) \mod 751$.

6. Using symmetric key encryption AES compute

$$c = E_{k_3}(m) = E_{197}(15) = 592 \mod 727$$

7. Using SHA-1 calculate the hash value

$$r = h(c, T_2) = h(592, 390) = 429 \mod 727$$

8. Calculate the signature s as:

$$\begin{aligned} s &= \frac{r_1}{r + d_A} \mod n \\ &= \frac{3}{429 + 10} \mod 727 \\ &= 3(439)^{-1} \\ &= 3(260) \\ &= 53 \end{aligned}$$

9. Compute $Q = r \cdot G = 429(0, 376) = (182, 667) \mod 751$

Generalized Unsigncryption

1. Compute $k = d_B \cdot P_A = 2(57, 332) = (731, 529) = (k_1, k_2) \mod 751$.
2. Compute $k^* = k_1 \cdot R = 731(2, 373) = (197, 107) = (k_3, k_4) \mod 751$.
3. Compute $m = D_{k_3}(c) = D_{197}(592) = 15 \mod 727$.
4. Regenerate the key T as:

$$\begin{aligned} T &= d_B \cdot s \cdot (Q + P_A) \mod n \\ &= (2)(53)[(182, 667) + (57, 332)] \mod 727 \\ &= 106(232, 701) \\ &= (6, 390) \\ &= (T_1, T_2) \end{aligned}$$

5. Compute $r' = h(c, T_2) = h(592, 390) = 429 \mod 727$.
6. As $r = r'$, therefore accecpt the ciphertext.

Chapter 5

Analysis of the Proposed Scheme

In this chapter, the security attributes "confidentiality, authenticity, integrity, non-repudiation, unforgeability, forward secrecy" will be discussed. Later on, the computational cost of the proposed scheme with different existing scheme and attack analysis will also be discussed.

5.1 Security Attributes

All of the security requirements are met by the proposed scheme, which are based on the assumptions that it is difficult to solve ECDLP and ECDHP.

5.1.1 Confidentiality

The proposed scheme offers secrecy if an attacker attacks to original message then the private key k_3 must be retrieved. If he gets k_1, an attacker will compute k_3 from Step (5) in Generalized Signcryption Algorithm 4.1.1. He must know the d_A (private key of A) for k_1, which is not possible. Therefore, he can not obtain the original message.

5.1.2 Authenticity

The proposed scheme provides authentication by including the certificate authority in the verification of both the receiver's and sender's public keys. To check the

authenticity of messages that have been received, the receiver utilises the sender's public key P_A.

5.1.3 Integrity

Integrity is provided by the proposed scheme. The recipient will check that the message m hasn't been tampered with throughout the transmission phase after getting the signcrypted text. In Step (4) of the Generalized Signcryption Algorithm 4.1.2, if an intruder alters the encoded message c to c', r' changes to r''. The produced r' in Step (6) of 4.1.2 is not going to be validated as a result of these modifications. If the encoded message c is modified, the recipient will be aware that the message has been moderated during transit.

5.1.4 Non-repudiation

When there is a disagreement between two parties, the recipient can transmit (c, s, Q) to check the validity of m. Utilising r' in Step (6) of 4.1.2, the judge will have the ability to authenticate the validity of m. In Step (8) of 4.1.1, the secret random number r_1 is used to generate the signature s, which is only known by the sender. As a result, Alice will be unable to refute that she is the sender of the communication.

5.1.5 Unforgeability

Unforgeability is a feature of the proposed scheme. Without the sender's secret key, the opponent is unable construct a appropriate (c, s, Q) of his choice. Suppose that an attacker takes any m' and creates (c, s, Q) of his choice. However, without knowing the private number r_1 in Step (2) of 4.1.1, he will be unable to create a valid signature. As a result, the signature will not be verified by the Generalized Unsigncryption Algorithm 4.1.2.

5.1.6 Forward Secrecy

If the sender's secret key d_A is exposed, an opponent will be unable to get any message m from it, because the preceding signcrypted text contains a concealed random number r_2. Interpreting r_2 necessitates solving ECDLP. In addition, every time m is to be signcrypted, the scheme needs a change in the random number r_2. This guarantees the capability of the proposed scheme for forward secrecy.

Schemes	**Confidentiality**	**Integrity**	**Unforgeability**	**Non repudiation**	**Authentication**	**Forward secrecy**
Zheng [10]	✓	✓	✓	✓	✗	✗
Elkamchochi [61]	✓	✓	✓	✓	✗	✗
Bao and deng [16]	✓	✓	✓	✓	✗	✗
Zheng and Imai [13]	✓	✓	✓	✓	✗	✗
Han et al. [30]	✓	✓	✓	✓	✓	✗
Zhou [62]	✓	✓	✓	✓	✓	✗
Gamage et al. [17]	✓	✓	✓	✓	✓	✗
Jung et al. [63]	✓	✓	✓	✓	✗	✗
Mohamed [27]	✓	✓	✓	✓	✓	✗
Kumar and Gupta [43]	✓	✓	✓	✓	✓	✓
Proposed	✓	✓	✓	✓	✓	✓

TABLE 5.1: Comparison of the proposed scheme with different existing schemes

5.2 Efficiency

We equate the proposed GSC with different existing schemes, and compare the computational cost. The numerical values in below table shows that the how many times an operation involved in a scheme.

Schemes	**HS**	**EM**	**EA**	**ME**	**MD**	**MM**	**MA**
Zheng [10]	4	-	-	3	1	2	1
Han [30]	4	5	1	-	2	4	3
Elkamchochi [61]	6	-	-	3	1	4	1
Bao and Deng [16]	6	-	-	5	1	1	1
Zheng and Imai [13]	4	3	1	-	1	3	1
Zhou [62]	6	6	7	-	1	4	2
Jung [63]	4	-	-	5	1	1	1
Mohamed [27]	6	6	1	-	1	-	1
Gamage [17]	4	-	-	5	1	1	1
Lal and Kushwa [64]	8	5	1	-	3	3	2
Kumar and Gupta [43]	4	4	1	2	1	2	1
Proposed	2	8	1	-	1	1	1

TABLE 5.2: Comparison of proposed scheme operations with different existing schemes

HS: One way hash function, EA: Elliptic curve point addition, EM: Elliptic curve point multiplication, MD: Modular division, ME: Modular exponentiation, MA: Modular addition, MM: Modular multiplication.

5.2.1 Computational Cost

The smaller key length is the main advantage of ECC which provides the same level of security over Elgamal [8] and RSA [7]. Another benefit is that it removes the need for storage. Signature generation in the proposed scheme requires only one hash value computation and basic arithmetic computations. Table 5.2 provides a comparison of the number of main operations included in the proposed scheme against existing schemes. In [65] using the "Cantroller Infineons SLE66CUX640P" a single eliptic curve point multiplication operation takes 83 milliseconds, while a single modular exponentiation takes 220 milliseconds. Table 5.3 compared the

computational cost of proposed scheme versus current signcryption schemes. Furthermore, the proposed scheme has extra features then existing schemes.

Schemes	**Computational time (ms)**	**Features**
Zheng [10]	$3 \times 220 = 660$	SC
Han [30]	$5 \times 83 = 415$	SC
Elkamchochi [61]	$3 \times 220 = 660$	SC
Bao and deng [16]	$5 \times 220 = 1100$	SC
Zheng and Imai [13]	$3 \times 83 = 249$	SC
Zhou [62]	$6 \times 83 = 498$	SC
Jung [63]	$5 \times 220 = 660$	SC
Mohamed [27]	$6 \times 83 = 498$	SC
Gamage et al. [17]	$5 \times 220 = 1100$	SC
Kumar and Gupta [43]	$4 \times 83 = 332$	SC
Lal and Kushwa [64]	$5 \times 83 = 415$	GSC
Proposed	$8 \times 83 = 664$	GSC

TABLE 5.3: Comparison of computational time (in ms) of the proposed generalized signcryption scheme (GSC) with existing signcryption (SC) schemes

5.3 Attack Analysis

In this section, the proposed scheme is analyzed, and it is revealed to be vulnerable to a number of known assault.

5.3.1 Chosen Plaintext Attack

This type of attack is utilized, when an adversary picks a message and gets the ciphertext associated with it. To figure out the hidden key, the attacker explores

the link among the original message and the corresponding ciphertext. This is a powerful attack since the adversary may deduce the private key from the produced ciphertext using any message. An adversary receives a pair of "(c, m)", he attempts to estimate the private key d_A. Given m and c, the attacker must solve ECDLP in order to find a d_A, which is computationally infeasible.

5.3.2 Ciphertext only Attack

An attacker receives the ciphertext message from publicly accessible information in this attack model and attempts to produce the actual plaintext message m or private key d_A. Later on, once the private key is revealed, he obtains all the original messages as ciphertext. If an opponent receives the ciphertext in the proposed scheme, he then seeks to acquire the private key or the plaintext message. In order to find d_A, the adversary must solve ECDLP when ciphertext c and public parameter r is given, which is computationally infeasible. Consequently, the attacker cannot generate the original message m without knowing the secret key of the sender.

5.3.3 Chosen Ciphertext Attack

In this attack, an intruder can pick different ciphertexts of his choice and can obtain their corresponding plaintexts. The attacker's simple objective is to retrieve the private key or get the communication involved in the secret parameters. An attacker selects the ciphertext c of his choice in the proposed scheme and obtains its corresponding original message m. Being given c and m, d_A is not possible to find, since it requires another k_1. In Step (5) of 4.1.1, the attacker tries to find $k^* = k_1 R$, then he must have r_2 which is the secret random number. But $R = r_2 G$, calculating r_2 again requires solving ECDLP, which with the specified settings of the proposed scheme is computationally infeasible. So, the proposed scheme is resistant to this assault.

5.3.4 Forgery Attack

An attacker intercepts the sender's and receiver's network correspondence in this attack model. The attacker's goal is to alter or substitute the original message with the intended message in such a manner that the unsigncryption algorithm verifies it correctly. Assume an attacker intercepts network communication between the sender and the recipient in the proposed scheme. The intruder changes and produces the signcrypted text of his choosing $(c^{'}, s^{'}, r^{'})$ and transmits it to the recipient, but the unsigncryption algorithm is unable to check the validity of the received message. In particular, in Generalized Signcryption Algorithm 4.1.1, the generation of the s involves hidden number r_1, and secret key of sender d_A that are not known to an adversary. Therefore, the bogus signcrypted text can not be checked by 4.1.2 without using these secret parameters. Hence, the forgery attack on the proposed scheme can not be mounted.

5.3.5 Man in the Middle Attack

An opponent engages himself in the conversation between the sender and the recipient. The adversary's goal is to either generate a shared mutual secret key or alter the sent data. In communication, a powerful authentication protocol is used for protection against this kind of attack. In the proposed scheme, assume that an opponent tries to manipulate the mutual secret key generation process. He choose his secret key d_M for this reason and calculates his public key $P_M = d_M G$ as an elliptic curve point. He must establish a separate and trustworthy connection with both the sender and the recipient after eavesdropping on the sender's and recipient's network communications. Firstly, through his public key P_M, the adversary attempts to establish a shared private key. But for either of the sender or recipient, he will be unable to create a valid mutual secret key, since it contains the hidden number r_2 in Step (3) of 4.1.1, which only a genuine sender has access to.

Chapter 6

Conclusion

In this thesis, firstly we review the Kumar and Gupta's signcryption scheme [43]. This scheme provides confidentiality and authenticity at same time. We extend this signcryption scheme to a generalized signcryption scheme. The proposed scheme provides the flexibility of signcryption scheme, encryption scheme, signature scheme as required. When both confidentiality and authenticity are required at the same time, the proposed scheme performs double functions and when just confidentiality and authenticity are required, it performs a single encryption/signature function without any modifications or additional calculation. Compared to the existing signcryption schemes, the proposed generalized signcryption scheme is more efficient because it has extra additional features. The proposed scheme's security depends on ECDLP, which is very secure. The proposed scheme's security shows that it is resistant to a variety of known assaults. The suggested scheme provides all of the security attributes. In future aspects generalized signcryption can be extended to a blind signcryption.

Bibliography

[1] W. Stallings, *Cryptography and network security, 4/E.* Pearson Education India, 2006.

[2] R. L. Rivest, "The rc5 encryption algorithm," in *International Workshop on Fast Software Encryption.* Springer, 1994, pp. 86–96.

[3] 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.

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

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

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

[7] E. Milanov, "The rsa algorithm," *RSA Laboratories*, pp. 1–11, 2009.

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

[9] R. C. Merkle, "A certified digital signature," in *Conference on the Theory and Application of Cryptology.* Springer, 1989, pp. 218–238.

[10] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) cost (signature)+ cost (encryption)," in *Annual international cryptology conference.* Springer, 1997, pp. 165–179.

[11] N. Kobliz, "Elliptic curve cryptography," *Math. Comput.*, vol. 48, pp. 203–209, 1987.

[12] V. S. Miller, "Advances in cryptologycrypto 85 proceedings," *Use of elliptic curves in cryptography*, pp. 417–426, 1986.

[13] Y. Zheng and H. Imai, "How to construct efficient signcryption schemes on elliptic curves," *Information processing letters*, vol. 68, no. 5, pp. 227–233, 1998.

[14] S. D. Galbraith and P. Gaudry, "Recent progress on the elliptic curve discrete logarithm problem," *Designs, Codes and Cryptography*, vol. 78, no. 1, pp. 51–72, 2016.

[15] J. H. An, Y. Dodis, and T. Rabin, "On the security of joint signature and encryption," in *International conference on the theory and applications of cryptographic techniques*. Springer, 2002, pp. 83–107.

[16] F. Bao and R. H. Deng, "A signcryption scheme with signature directly verifiable by public key," in *International Workshop on Public Key Cryptography*. Springer, 1998, pp. 55–59.

[17] C. Gamage, J. Leiwo, and Y. Zheng, "An efficient scheme for secure message transmission using proxy-signcryption," in *Proceedings of the Twenty Second Australasian Computer Science Conference*, 1999, pp. 18–21.

[18] B. Nayak, "Signcryption schemes based on elliptic curve cryptography," Ph.D. dissertation, 2014.

[19] R.-J. Hwang, C.-H. Lai, and F.-F. Su, "An efficient signcryption scheme with forward secrecy based on elliptic curve," *Applied Mathematics and computation*, vol. 167, no. 2, pp. 870–881, 2005.

[20] D. Boneh and I. E. Shparlinski, "On the unpredictability of bits of the elliptic curve diffie-hellman scheme," in *Annual International Cryptology Conference*. Springer, 2001, pp. 201–212.

[21] J.-B. Shin, K. Lee, and K. Shim, "New dsa-verifiable signcryption schemes," in *International Conference on Information Security and Cryptology*. Springer, 2002, pp. 35–47.

[22] R. Tso, T. Okamoto, and E. Okamoto, "An improved signcryption scheme and its variation," in *Fourth International Conference on Information Technology (ITNG'07)*. IEEE, 2007, pp. 772–778.

[23] A. K. Singh, "A review of elliptic curve based signcryption schemes," *International Journal of Computer Applications*, vol. 102, no. 6, 2014.

[24] E. A. Hagras, D. El-Saied, and H. H. Aly, "A new forward secure elliptic curve signcryption key management (fs-ecskm) scheme for heterogeneous wireless sensor networks," *International Journal of Computer Science and Technology*, vol. 2, no. 2, pp. 19–23, 2011.

[25] S. Bala, G. Sharma, and A. K. Verma, "An improved forward secure elliptic curve signcryption key management scheme for wireless sensor networks," in *IT Convergence and Security 2012*. Springer, 2013, pp. 141–149.

[26] F. Amounas and E. H. Kinani, "An efficient signcryption scheme based on the elliptic curve discrete logarithm problem," *International Journal of Information and Network Security*, vol. 2, no. 3, p. 253, 2013.

[27] E. Mohamed and H. Elkamchouchi, "Elliptic curve signcryption with encrypted message authentication and forward secrecy," *International Journal of Computer Science and Network Security*, vol. 9, no. 1, pp. 395–398, 2009.

[28] R. Ahirwal, A. Jain, and Y. Jain, "Signcryption scheme that utilizes elliptic curve for both encryption and signature generation," *International Journal of Computer Applications*, vol. 62, no. 9, 2013.

[29] S. Das and B. Samal, "An elliptic curve based signcryption protocol using java," *International Journal of Computer Applications*, vol. 66, no. 4, 2013.

[30] Y. Han, X. Yang, P. Wei, Y. Wang, and Y. Hu, "Ecgsc: elliptic curve based generalized signcryption," in *International Conference on Ubiquitous Intelligence and Computing*. Springer, 2006, pp. 956–965.

[31] X. an Wang, X. Yang, and J. Zhang, "Provable secure generalized signcryption," *Journal of computers*, vol. 5, no. 5, p. 807, 2010.

[32] G. Yu, X. Ma, Y. Shen, and W. Han, "Provable secure identity based generalized signcryption scheme," *Theoretical Computer Science*, vol. 411, no. 40-42, pp. 3614–3624, 2010.

[33] P. Kushwah and S. Lal, "An efficient identity based generalized signcryption scheme," *Theoretical Computer Science*, vol. 412, no. 45, pp. 6382–6389, 2011.

[34] C. Zhou, W. Zhou, and X. Dong, "Provable certificateless generalized signcryption scheme," *Designs, codes and cryptography*, vol. 71, no. 2, pp. 331–346, 2014.

[35] M. H. Au, Y. Mu, J. Chen, D. S. Wong, J. K. Liu, and G. Yang, "Malicious kgc attacks in certificateless cryptography," in *Proceedings of the 2nd ACM symposium on Information, computer and communications security*, 2007, pp. 302–311.

[36] G. Wei, J. Shao, Y. Xiang, P. Zhu, and R. Lu, "Obtain confidentiality or/and authenticity in big data by id-based generalized signcryption," *Information Sciences*, vol. 318, pp. 111–122, 2015.

[37] C.-X. Zhou, "An improved multi-receiver generalized signcryption scheme," *International Journal of Network Security*, vol. 17, no. 3, pp. 340–350, 2015.

[38] Y. Han and X. Gui, "Adaptive secure multicast in wireless networks," *International Journal of Communication Systems*, vol. 22, no. 9, pp. 1213–1239, 2009.

[39] Y. Han, Y. Bai, D. Fang, and X. Yang, "The new attribute-based generalized signcryption scheme," in *International Conference of Young Computer Scientists, Engineers and Educators*. Springer, 2015, pp. 353–360.

[40] C.-X. Zhou, "Identity based generalized proxy signcryption scheme," *Information Technology and Control*, vol. 45, no. 1, pp. 13–26, 2016.

[41] C. Zhou, Z. Cui, and G. Gao, "Efficient identity-based generalized ring signcryption scheme." *TIIS*, vol. 10, no. 12, pp. 5553–5571, 2016.

[42] A. Zhang, L. Wang, X. Ye, and X. Lin, "Light-weight and robust security-aware d2d-assist data transmission protocol for mobile-health systems," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 3, pp. 662–675, 2016.

[43] M. Kumar and P. Gupta, "An efficient and authentication signcryption scheme based on elliptic curves," *MATEMATIKA: Malaysian Journal of Industrial and Applied Mathematics*, pp. 1–11, 2019.

[44] J. J. Rotman, *Journey into mathematics: An introduction to proofs*. Courier Corporation, 2013.

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

[46] K. S. McCurley, "The discrete logarithm problem," in *Proc. of Symp. in Applied Math*, vol. 42. USA, 1990, pp. 49–74.

[47] Z. Qi, J. P. Koenig, J. Levesque, and S. Kolisetty, "Elliptic curves and their applications in symmetric and asymmetric encryption."

[48] R. Davis, "The data encryption standard in perspective," *IEEE Communications Society Magazine*, vol. 16, no. 6, pp. 5–9, 1978.

[49] R. C. Merkle and M. E. Hellman, "On the security of multiple encryption," *Communications of the ACM*, vol. 24, no. 7, pp. 465–467, 1981.

[50] E. Barker and N. Mouha, "Recommendation for the triple data encryption algorithm (tdea) block cipher," National Institute of Standards and Technology, Tech. Rep., 2017.

[51] A. Nadeem and M. Y. Javed, "A performance comparison of data encryption algorithms," in *2005 international Conference on information and communication technologies*. IEEE, 2005, pp. 84–89.

[52] D. W. Kravitz, "Digital signature algorithm," Jul. 27 1993, uS Patent 5,231,668.

[53] S. Verbruggen, S. De Sutter, S. Iliopoulos, D. Aggelis, and T. Tysmans, "Experimental structural analysis of hybrid composite-concrete beams by digital image correlation (dic) and acoustic emission (ae)," *Journal of Nondestructive Evaluation*, vol. 35, no. 1, p. 2, 2016.

[54] F. A. Petitcolas, "Kerckhoffs' principle." 2011.

[55] R. Canetti, S. Halevi, and J. Katz, "Chosen-ciphertext security from identity-based encryption," in *International conference on the theory and applications of cryptographic techniques*. Springer, 2004, pp. 207–222.

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

[57] F. Callegati, W. Cerroni, and M. Ramilli, "Man-in-the-middle attack to the https protocol," *IEEE Security & Privacy*, vol. 7, no. 1, pp. 78–81, 2009.

[58] K.-J. Zhang, W.-W. Zhang, and D. Li, "Improving the security of arbitrated quantum signature against the forgery attack," *Quantum information processing*, vol. 12, no. 8, pp. 2655–2669, 2013.

[59] D. Mahto and D. K. Yadav, "Performance analysis of rsa and elliptic curve cryptography." *IJ Network Security*, vol. 20, no. 4, pp. 625–635, 2018.

[60] J. H. Silverman, *The arithmetic of elliptic curves*. Springer Science & Business Media, 2009, vol. 106.

[61] H. Elkamchouchi, M. Nasr, and R. Ismail, "A new efficient publicly verifiable signcryption scheme and its multiple recipients variant for firewalls implementation," in *2009 National Radio Science Conference*. IEEE, 2009, pp. 1–9.

[62] X. Zhou, "Improved signcryption scheme with public verifiability," in *2009 Pacific-Asia Conference on Knowledge Engineering and Software Engineering*. IEEE, 2009, pp. 178–181.

[63] M. Toorani and A. A. Beheshti, "An elliptic curve-based signcryption scheme with forward secrecy," *arXiv preprint arXiv:1005.1856*, 2010.

[64] S. Lal and P. Kushwah, "Id based generalized signcryption." *IACR Cryptol. ePrint Arch.*, vol. 2008, p. 84, 2008.

[65] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *Proceedings of the 1st ACM Conference on Computer and Communications Security*, 1993, pp. 62–73.