

CAPITAL UNIVERSITY OF SCIENCE AND
TECHNOLOGY, ISLAMABAD



Analysis and Improvement of Some Signcryption Schemes Based on Elliptic Curve

by

Malik Zia Ullah Bashir

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

in the

Faculty of Computing

Department of Mathematics

2021

Copyright © 2021 by Malik Zia Ullah Bashir

All rights reserved. No part of this thesis may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, by any information storage and retrieval system without the prior written permission of the author.

Dedicated to My Father and Mother (Late)

List of Publications

It is certified that following publication(s) have been made out of the research work that has been carried out for this thesis:-

1. **Malik Zia Ullah Bashir**, Rashid Ali, “A Multi Recipient Aggregate Sign-cryption Scheme Based on Elliptic Curve”, *Wireless Personal Communications*, vol 115(2), pp. 1465-1480, Nov 2020.
2. **Malik Zia Ullah Bashir**, Rashid Ali, “Cryptanalysis and improvement of a blind multi-document signcryption scheme”, *Cryptologia*, vol 13, pp. 1-5, 18 May 2020.
3. **Malik Zia Ullah Bashir**, Rashid Ali, “Cryptanalysis and improvement of blind signcryption scheme based on elliptic curve”, *Electronics Letters*, vol. 55(8), pp. 457–459, 18 April 2019.
4. **Malik Zia Ullah Bashir**, Rashid Ali, “Cryptanalysis and improvement of an elliptic curve based signcryption scheme for firewalls”, *PLoS ONE*, vol. 13, pp. 1-11, 13 Dec 2018.

Malik Zia Ullah Bashir

Registration No. DMT153001

Acknowledgements

At the end of my thesis, I would like to thank all those people who made this thesis possible and an unforgettable experience for me. Foremost, I would like to express my sincere gratitude to my supervisor Dr. Rashid Ali for the continuous support for my PhD study and research. I am very thankful for his patience, motivation, enthusiasm, continuous advice and encouragement throughout the course of this thesis. His guidance helped me in all the time of research and writing of this thesis. I feel a great honor in his supervision and once again thankful for the guidance and great effort that he put to train me in the field of research. Beside this i am thankful to head of Mathematics department Dr. Sageer Ahmad for providing us a learning and creative environment.

Most importantly, none of this would have been possible without the love and patience of my family. I have no words to say thanks to my father, brother, sister and my wife who encouraged me and prayed for me throughout the time of my study and research. They are the constant source of love, concern, support and strength for me. This thesis is heartily dedicated to my mother whose soul has rest in peace.

May the Almighty God richly bless all of you.

(Malik Zia Ullah Bashir)

Abstract

Signcryption is a new technique in cryptography that fulfills the security requirements of encryption and digital signature in a single logical step. This helps in reducing the computational and communicational cost as compared to traditionally used Signature-then-Encryption technique. Signcryption schemes offer different security attributes of public verifiability, non-repudiation, authentication integrity, confidentiality, forward secrecy and unforgeability. Blind signcryption schemes are extension of signcryption schemes and they offer security attributes of anonymity and untraceability in-addition to the properties offered by any signcryption scheme. Previously, many signcryption schemes were introduced and each of them offer different level of security. attributes and computational cost depending upon the requirement. Some of these schemes are proved to be insecure and need further improvements. Due to rapid increase and advancement in cryptographic attacks, it is an essential requirement to check and analyze the security strength of existing signcryption schemes. The aim of this study is to analyze the security aspects of some recently introduced signcryption schemes. For this purpose, different attacks are mounted on these schemes for any possible security flaws and issues. After the successful cryptanalysis of the schemes, the security flaws and issues are highlighted. The modified and improved versions of these schemes are proposed to fix the security flaws and issues. The security analysis of the modified schemes are also performed to show their resistance against the existing attacks. Moreover, a new aggregate signcryption scheme based on elliptic curve is proposed together with its four different versions. The analysis of the proposed scheme shows that it is more efficient as compared to existing signcryption schemes. The security analysis shows that the proposed scheme is secure and it offers the features of non-repudiation, unforgeability, message confidentiality, forward secrecy, integrity, authentication and unforgeability. Different existing attacks are also mounted on the proposed scheme to show its resistance against them.

Contents

Author's Declaration	v
Plagiarism Undertaking	vi
List of Publications	vii
Acknowledgement	viii
Abstract	ix
List of Figures	xiii
List of Tables	xiv
Abbreviations	xv
Symbols	xvi
1 Introduction	1
1.1 Background and Literature Review	1
1.1.1 Cryptography	3
1.1.2 Signcryption	4
1.1.3 Blind Signcryption	5
1.1.4 Cryptanalysis	7
1.2 Thesis Contribution	8
1.3 Thesis Outline	9
2 Mathematical Background	11
2.1 Preliminaries	11
2.2 Modular Arthemetic	13
2.3 Hash Function	15
2.3.1 Types of Hash Functions	17
2.4 Elliptic Curve	17
2.4.1 Elliptic Curve over R	18
2.4.2 Elliptic Curve over a Finite Field	20

2.5	Hyperelliptic Curve	22
2.5.1	Divisor	23
2.5.2	Mumford representation	25
2.5.3	Cantors Algorithm	26
3	Overview of Cryptography	32
3.1	Cryptography	32
3.1.1	Private (Symmetric) Key Cryptography	32
3.1.2	Public (Asymmetric) Key Cryptography	33
3.2	Digital Signature	34
3.2.1	Blind Signature	36
3.2.2	Aggregate Signature	37
3.3	Elliptic Curve based Cryptosystem	39
3.4	Signature-then-Encryption	42
3.5	Signcryption	44
3.5.1	Security Attributes of Signcryption Scheme	44
3.5.2	Zheng's Elliptic Curve based Signcryption Scheme	46
3.6	Cryptanalysis	50
3.6.1	Chosen Plaintext Attack	50
3.6.2	Known Plaintext Attack	51
3.6.3	Chosen Ciphertext Attack	51
3.6.4	Known Ciphertext Attack	51
3.6.5	Forgery Attack	52
3.6.6	Man-In-The-Middle Attack	53
3.6.7	Man-At-The-End Attack	54
3.6.8	Brute Force Attack	54
4	Cryptanalysis and Improvement of an Elliptic Curve based Sign- cryption Scheme for Firewalls	56
4.1	Signcryption scheme of Iqbal et al. [86]	57
4.2	Cryptanalysis	59
4.2.1	Proof of Correctness	61
4.3	Modified Signcryption Scheme	62
4.3.1	Proof of Correctness	64
4.4	Analysis of Modified Scheme	65
4.4.1	Security Attributes	65
4.4.2	Attack Analysis	68
4.5	Conclusion	70
5	Cryptanalysis and Improvement of Blind Signcryption Scheme based on Elliptic Curve	72
5.1	Elliptic Curve based Blind Signcryption Scheme [87]	73
5.2	Cryptanalysis	76
5.2.1	Proof of Correctness	77
5.3	Modified Blind Signcryption scheme	78

5.3.1	Proof of Correctness	79
5.4	Analysis of Modified Scheme	80
5.4.1	Security Attributes	81
5.4.2	Attack Analysis	83
5.4.3	Computational Cost	84
5.5	Conculsion	85
6	Cryptanalysis and Improvement of a Blind Multi-Document Sign- cryption Scheme	87
6.1	Blind Signcryption [4]	88
6.2	Cryptanalysis	91
6.2.1	Proof of Correctness	92
6.2.2	Security Flaws	93
6.3	Modified Blind Signcryption Scheme	94
6.3.1	Proof of Correctness	96
6.4	Analysis of Modified Scheme	97
6.4.1	Security Attributes	97
6.4.2	Resistance against the Attack	100
6.5	Conclusion	100
7	A Multi Recipient Aggregate Signcryption Scheme based on El- liptic curve	102
7.1	Proposed Aggregate Signcryption Scheme	103
7.1.1	Signcryption Scheme for Single Message (Version-1)	104
7.1.2	Aggregate Signcryption Scheme for Multiple Messages (Version- 2)	108
7.1.3	Multi-Recipient Signcryption Scheme for Single Message (Version- 3)	111
7.1.4	Multi-Recipient Aggregate Signcryption Scheme for Multi- ple Messages (Version-4)	115
7.2	Analysis of the Proposed Scheme	118
7.2.1	Security Attributes	118
7.2.2	Computational Cost	120
7.2.3	Attack Analysis	121
7.3	Conclusion	125
8	Conclusion and Future Work	126
8.1	Future Work	128
	Bibliography	130

List of Figures

2.1	Cryptographic Hash Function	16
2.2	Elliptic Curve	18
2.3	ECC Point Addition	19
2.4	ECC Point Doubling	20
2.5	Hyperelliptic Curve	22
2.6	Geometrical Representation of Divisor	24
3.1	A Typical Cryptosystem	33
3.2	Symmetric Encryption Model	33
3.3	Asymmetric Encryption Model	34
3.4	Digital Signature Model	35
3.5	Blind Signature Model	36
3.6	Aggregate Signature Model	38
3.7	Diffie Hellman Key Exchange Protocol	41
3.8	Sign-then-Encryption Model	43
3.9	Signcryption Model	45
3.10	Chosen Plaintext Attack Model	51
3.11	Chosen Ciphertext Attack Model	52
3.12	Ciphertext Only Attack Model	52
3.13	Man-In-The-Middle Attack	53
3.14	Man-At-The-End Attack Model [104]	54
3.15	Brute Force Attack [104]	55
4.1	Cryptanalysis Model	60
6.1	Structure of Blind Signcryption Scheme [4] and its Cryptanalysis . .	93
7.1	Comparison of computational Time (in ms) of Major Operations of Proposed Scheme with Existing Schemes	122
7.2	Man in the Middle Attack	124

List of Tables

2.1	Comparison of Cryptographic Hash Functions	16
2.2	Points of the Elliptic Curve E defined over the Finite Field \mathbb{F}_{73} . .	22
3.1	Comparison of Key Size Of ECC and RSA [102]	39
4.1	Global Parameters of the Scheme [86]	57
4.2	Comparision of Modified Scheme with Existing Schemes	69
5.1	Global Parameters of the Scheme [87]	74
5.2	Comparision of Modified Scheme with the Proposed Scheme [87] . .	83
5.3	Comparison of Major Operations of Modified Scheme with the Scheme Proposed in [87]	85
6.1	Global Parameters of the Proposed Scheme [4]	88
7.1	Global Parameters of the Proposed Scheme	104
7.2	Comparision of Signcryption Scheme with Existing Schemes [86] . .	120
7.3	Comparison of Major Operations involved in the Proposed Scheme and Existing Schemes	121
7.4	Average Computational Time (in ms) of Major Operations involved in Proposed Scheme and existing schemes	121

Abbreviations

ECC	Elliptic Curve Cryptography
RSA	Rivest Shamir Adleman
DES	Data Encryption Standard
AES	Advanced Encryption Standard
ECC	Elliptic Curve Cryptosystem
DH	Diffie-Hellman
DLP	Discrete Logarithm Problem
IFP	Integer Factorization Problem
ECDLP	Elliptic Curve Discrete Logarithm Problem
ECDH	Elliptic Curve Diffie-Hellman
ISO	International Organization for standardization
ECPA	Elliptic Curve Point Addition
ECPM	Elliptic Curve Point Multiplication

Symbols

h	One Way Hash Function
Kh	Keyed One Way Hash Function
mod	Modular Operator
n	Order of Base Point G
E	Elliptic Curve
\mathcal{O}	Point at Infinity
H	Hyperelliptic Curve
D	Divisor of Hyperelliptic Curve
m	Plaintext Message
c	Ciphertext Message
$\text{gcd}(a, b)$	Greatest Common Divisor of a and b
\mathbf{Z}	Set of Integers
p	Large Prime Number
\mathbb{F}_p	Finite Field

Chapter 1

Introduction

This chapter provides the background and comprehensive literature review of cryptography, public key cryptography, signcryption and cryptanalysis in order to have the excellent understating of this thesis for the readers.

1.1 Background and Literature Review

Cryptology is the scientific way of generating and breaking the secret codes. It deals with the comprehensive study of cryptanalysis and cryptography. Cryptography is the combination of two Greek words *kryptos* and *graphein* which mean hidden and writing respectively. The term is therefore used for the science of secret communication in the presence of an unauthorized third party. On the other hand, cryptanalysis is the study of finding the meaning of encrypted information without having any secret information of participants.

The need to produce hidden messages has been with us since we came out of the caves and began living in small communities. The initial form of cryptography was found in the cradle of civilizations of Greece, Rome and Egypt. As early as 1900 BC, the Greek way of cryptography was to loop a tape around a stick, and then write a message on a wounded tape [1]. When the tape was unwound, the writing would become meaningless. The recipient had a stick of the same diameter, and

then used it to decode the message. The Roman style of cryptography was also known as the Caesar Shift Cipher or Substitution cipher [1]. Julius Caesar (100 BC–44 BC) was famous for sending the coded messages to his army commanders posted in the battle field. In the Caesar Shift Cipher, each character of a message is replaced by another character to create a coded message. The variation used by Caesar was a shift of three alphabets in such a way that character C was substituted by F, D was substituted by G, and so on.

There have been three distinct ages in the history of cryptography. The first came the age of handbook (Manual) cryptography, which began with the origins of the subject to World War I. In this phase, the ciphers were limited to a few pages in size and applicable for just a few thousand characters. The basic principles of both cryptanalysis and cryptography were well known, but the achieved security was still restricted by what could be performed manually. The initial appearance of cryptography was the simple writing of a message in such a way that unauthorized people were unable to read. The second age was the mechanization of cryptography, began shortly after World War I. The associated technologies included telegraph, telephone and rotor machines used in the Second World War. The third stage, which dates from the last two decades of the 20th century to present, marked the most significant changes in the expansion of cryptography to the digital era. Modern cryptography not only deals with confidentiality of data but also deals the security attributes of integrity, authentication, unforgeability, non-repudiation, public verification and forward secrecy (For detail see Section 3.5.1). Today, the cryptography is widely used in different aspects of human life such as digital signature, electronic voting systems, electronic cash payment systems, online shopping, cell phones, remote controls, cash machine, credit cards, secure email, transfer of money between banks, satellite TV and immobilizer system in cars [2–4].

1.1.1 Cryptography

Cryptography is divided into two branches namely: public (Asymmetric) key cryptography and private key (Symmetric) cryptography. In symmetric cryptography, only one key is involved in encryption and decryption process while in asymmetric cryptography two related keys are used, one is known as private key and other is called public key. The state of the art symmetric encryption schemes are DES [5], Triple DES [6] and currently in use AES [7]. The well known examples of public key cryptography are ECC [8], ElGamal [9] and RSA [10]. In 1985, Victor Miller and Neal Koblitz independently introduced the Elliptic Curve Cryptography (ECC) [8]. The main advantage of ECC is that it uses parameters of smaller size as compared to ElGamal [9] and RSA [10]. Longa and Miri [11] introduced a flexible technique for accelerating the computation of elliptic curve points over finite field. This flexible technique provides more benefits for parallel schemes such as 160-bit Non Adjacent Form (NAF) scalar multiplication with parallel Single Instruction Multiple-Data (SIMD) operations reducing the computational cost by 63% to 70 %. Bailey and Paar [12] proposed an efficient method for working with elliptic curve arithmetic using finite fields. Rao and Setty [13] introduced two different methods for mapping of the alphanumeric characters to xy -coordinate of the elliptic curve. King [14] proposed a new method for converting the arbitrary message of any size into an elliptic curve point without any modification in the original message. Namiq et al. [15] introduced a new encryption scheme that uses computations in elliptic curve group over finite fields. In their scheme, the ciphertext is transmitted through public channel in the form of elliptic curve point. The proposed scheme uses two secret keys to increase the security as compared to one secret key based encryption schemes. In their scheme, if sender's long term private key is compromised even then an adversary will not get the contents of the message without the second secret key. Recently, some researchers proposed some new encryption schemes and their related applications [16–19].

1.1.2 Signcryption

Digital signature and encryption are the basic tools of cryptography which provide the guarantee of authentication, confidentiality and integrity. In a traditionally used signature-then-encryption technique, the task of both authentication and encryption is fulfilled by first signing the digital document and then encrypting the signed document for transmission over an unsecured network. It has disadvantages of high computational cost and low efficiency. In 1997 Zheng [20] introduced a new cryptographic scheme named as a Signcryption scheme. This scheme provides the security attributes of encryption and digital signature in single step. In Zheng's signcryption scheme [20], the sender derives the secret key for symmetric encryption by using receiver's public key. After receiving the signcrypted text, receiver gets the same secret key by using his private key. Zheng [20] analysis shows that proposed scheme reduces 50% computational overheads and 85% computational cost as compared to traditionally used signature-then-encryption scheme.

After this, various signcryption schemes were introduced over the years, each scheme having its own benefits and drawbacks. Zheng and Imai [21] introduced a first signcryption scheme that uses elliptic curves over a finite field. It reduces the communication and computational cost in comparison with the other cryptosystems like ElGamal [9] and RSA [10]. Bao and Deng [22] modified the signcryption scheme of Zheng [21] in such a way that the judge can authenticate the signature of a message without using the private key of recipient. Ahirwal and Jain [23] introduced a new signcryption scheme that uses elliptic curve for generation of digital signature and encryption. A new technique is developed for signature generation which has less computational cost as compared to existing schemes that depends upon usage of hash functions. Jung et al. [24] analysis shows that Zheng signcryption scheme [20] lost message confidentiality if the secret key of the sender is compromised. He proposed a new signcryption scheme to overcome the drawbacks of Zheng [20] scheme with additional forward secrecy property. Gamage et al. [25] modified Zheng's signcryption scheme [20] in such a way that anyone can authenticate the signature of the corresponding ciphertext. The proposed signcryption scheme is based upon discrete logarithm problem

(DLP) for firewalls authentications but does not provide multi-reciever functionality. Toorani and Shirazi [26] introduced the elliptic curve based signcryption scheme with additional forward secrecy property. For some more recent authentication and signcryption protocols and their related applications, we refer to the work presented in [27–47]. Selvi [48] introduced an identity based signcryption technique for multiple receivers by using bilinear pairing. Mohammad et al. [49] introduced a new elliptic curve based signcryption scheme with froward secrecy and encrypted message authentication. The security of their proposed signcryption scheme relies upon elliptic curve discrete logarithm problem. Han et al. [50] proposed a multi-party signcryption scheme which uses the computation on an elliptic curve. Their scheme provides security properties of integrity, unforgebility, confidentiality, non-repudiation and authentication. For some more multi-recipient signcryption schemes we refer to the work presented in [51–55].

Boneh et al. [56] introduced a new aggregate signature scheme that reduces the size of certificate chains. If there are n distinct users and n distinct messages then aggregating all distinct n signatures to a single short signature in such a way that each user assures the authenticity of received message. The proposed scheme reduces the communication and computational cost as compared to single signature schemes. Horng et al. [57] introduced a new efficient certificate less aggregate signature scheme for vehicular sensor networks. The proposed scheme achieves the conditional privacy preservation and it is secure against adaptively chosen plaintext attack. Their scheme has less computational overhead as compared to existing aggregate signature schemes. For some more aggregate signature/signcryption protocols we refered to [58–61].

1.1.3 Blind Signcryption

In 1983, Chaum [3] extended the concept of digital signature and introduced a new scheme called “Blind Signatures for Untraceable Payments” based on RSA algorithm [10]. According to Chaum’s blind signature scheme, three participants the sender (requester), the signer and the receiver are involved. The sender blinds

the message and sends it to the signer for signature. The signer signs the message without reading the contents of the original message and sends it to the sender. Then sender unblinds the message and transmits the signed encrypted message to the receiver.

A blind signcryption scheme is the combination of blind digital signature and encryption in a single logical step. It provides two additional properties of anonymity and untraceability in-addition to the properties that are offered in signcryption scheme. Recall that, signcryption technique [20] is the combination of two cryptographic functions encryption and digital signature, which are performed simultaneously to reduce the computational cost. Any signcryption scheme is expected to provide the security attributes of non-repudiation, message integrity, confidentiality, unforgeability and authentication.

For multiple digital documents, signing each document separately requires extra computational and communication cost in the process of digital signature. To overcome this issue, a single signature generated from multiple documents is required for blind signature on multiple digital documents. The single blind signature is generated from all the multiple messages and cannot be verified if only some of the messages are known. So verification of digital signature requires all the multiple digital documents. The blind signcryption scheme due to its properties can be used mostly in e-voting system, e-cash payment system and e-bidding [4].

Huang and Chang [62] proposed a new efficient blind signcryption scheme for electronic cash payment systems. Their scheme reduces the computational cost for an online judge and hence it is suitable for resource constrained devices such as mobile units. Nikooghadam and Zakerolhosseini [63] introduced an untraceable blind signature scheme that uses the computations of elliptic curves. The computational cost and security of their scheme is quite less when compared to existing blind signature schemes. The security of proposed scheme relies on ECDLP that is computationally infeasible to solve for a well-chosen curve. Pointcheval and Stern [64] introduced a new blind signature scheme that is based upon integer factorization problem. The proposed scheme is secure but is not efficient for small devices. Dhanashree and Agrawal [65] uses elliptic curve to generate blind signature for

electronic voting system. The proposed scheme involves hash function for blinding a message. Awasthi and Lal [66] introduced a new blind signcryption scheme, which offers anonymity, confidentiality, untraceability and unlinkability. Delos and Quisquater [67] proposed an efficient signature scheme in which the operation of signature on digital document is designated to multiple signers. After signing the document, each signer interact with a combiner to generate a multi-signature. This type of signature scheme is used when the owner and the signer of the document are different entities and signature from multiple signers are required. Recall that, in a blind signcryption scheme, the power of signature is delegated to single signer and the document is blinded before it transmits to the signer-end. Moreover, a blind signcryption not only provides a blind signature but also has the facility of encryption as well. Chakraborty and Mehta [68] introduced a elliptic curve based blind signature scheme with double blinding. Due to smaller key size of elliptic curve, this scheme can be implemented in resource constrained devices. Tsai and Su [69] proposed a new elliptic curve based blind signature scheme. The proposed scheme provides low communication and computational cost as compared to existing blind signature schemes. The proposed scheme gets a higher level of security as well as high processing speed. Huifang et al. [70] introduced a new blind proxy signcryption scheme for powerful server to overcome the computational burden as compared to low power device. For some more blind signature and blind signcryption schemes, we refer the work presented in [71–74].

1.1.4 Cryptanalysis

The obvious presence of cryptanalyst in the communication channel required a strong authentication mechanism for safe and secure communication. The cryptanalyst tries to find the contents of an original message by seeking the weaknesses in cryptosystem, without having any secret information. In today's world, some of the attacks applied on cryptosystems are known plaintext attack, man-in-the-middle attack, chosen plaintext attack, forgery attack, ciphertext only attack and man-at-the-end attack (For detail see Section 3.6). Cheng and Wen [75] modified

a signcryption scheme of Liu et al. [76]. Their analysis shows that the proposed scheme [76] is not secure and unable to provide unforgeability against the chosen ciphertext attack. The cryptanalysis by Hu et al. [77] shows that the presented scheme in [78] is not secure and vulnerable to offline password guessing attack. To overcome this security issue, they proposed an improved version of the scheme. Their analysis shows that the improved scheme is secure against the existing attacks and is more efficient than the scheme proposed in [78]. For the cryptanalysis of some other signcryption schemes, we refer to the work presented in [79–85].

1.2 Thesis Contribution

Due to rapid increase and advancement of attacks on modern cryptosystems, there is a need to analyze the security of existing signcryption schemes. In this research, the investigation on the security strength of the existing signcryption schemes [4, 86, 87] is carried out for any possible security flaws and issues. After the successful cryptanalysis of these signcryption schemes, their modified versions are introduced to fix the security flaws. More precisely, the entire study is carried out:

Iqbal et al. [86] introduced a new efficient elliptic curve based signcryption scheme for firewalls. They claim that their scheme is secure and no one can duplicate the original message. The cryptanalysis of the scheme [86] is performed for highlighting the possible security flaws. After mounting the existing attacks, the claimed security attributes of non-repudiation, unforgeability, integrity and authentication of the scheme are compromised. To fix the security flaws, a modified version of the scheme is proposed. The security analysis of the proposed scheme is also carried out to show its resistance against the existing attacks.

In [87], a new elliptic curve based blind signcryption scheme is introduced. The claimed security attributes of the scheme proposed in [87] are confidentiality, sender anonymity, message integrity, authentication, unforgeability, signer non-repudiation, forward secrecy, blindness and message untraceability. The security analysis of the scheme [87] is carried out and found it to be insecure against the

existing attacks. The claimed security attributes of authentication, message integrity, signer non-repudiation and unforgeability of the scheme are compromised. After the successful cryptanalysis, a modified version of the scheme is introduced together with its security analysis. The modified scheme is further tested against the existing attacks and found it to be secure.

A new hyperelliptic curve based blind signcryption scheme is introduced in [4]. It is capable of transmitting the multiple digital documents at receiver's end. The investigation of the security strength of scheme [4] is performed for highlighting the possible weakness in the scheme. The successful cryptanalysis shows that the scheme proposed in [4] is not secure and unable to provide the claimed security attributes of authentication and message integrity. To counter these flaws a modified scheme is introduced together with its security analysis.

A new multi-recipient aggregate signcryption scheme based upon elliptic curve is also introduced. The proposed scheme consists of four different versions and therefore capable of sending the, single message to receiver (Version-1), multiple messages to receiver (Version-2), single message to multiple recipients (Version-3), multiple messages to multiple recipients (Version-4). The security and cost analysis of the scheme is also performed. It has been noticed that the proposed scheme is more efficient as compared to existing schemes (for details see Section 7.2.2). The proposed scheme is also found to have resistance against many known attacks.

1.3 Thesis Outline

The thesis is organized as follows:

- In Chapter 2 and 3, we describe the preliminaries related to cryptography and number theory respectively. In these chapters, basic definitions and concepts with explanatory examples are presented for better understanding the rest of the thesis.

- In Chapter 4, the security of elliptic curve based signcryption scheme for firewalls [86] are analyzed. After the successful cryptanalysis, a improved version of the scheme is introduced to fix the security flaws. The contents of this chapter are published in “Plos One” [88].
- In Chapter 5, the cryptanalysis and improvement of a blind signcryption scheme based on elliptic curve [87] is presented. The security of improved scheme is also analyzed in this chapter. The contents of this chapter has been published in the international journal “Electronics Letters” [89].
- Chapter 6 describes the cryptanalysis and improvement of a blind multi-document signcryption scheme [4]. The proof of cryptanalysis of the improved scheme is also given in this chapter. The contents of this chapter has been published in the journal “Cryptologia” [90].
- In Chapter 7, a new multi recipient elliptic curve based aggregate signcryption scheme are proposed. The security and cost analysis of the proposed scheme is presented in this chapter. The contents of this chapter are published in the international journal “Wireless Personal Communications” [91].

Chapter 2

Mathematical Background

Algebra and number theory plays a very significant role in the development of modern cryptography. Therefore, the basic concepts, definitions and tools from these areas of mathematics are explained in this chapter to investigate their applications in the subject of cryptography. In the next section, some basic definitions and concepts related to these areas are stated.

2.1 Preliminaries

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

- (A1) Closure: If a and b belong to G , then $a.b$ is also in G .
- (A2) Associative: $a.(b.c) = (a.b).c$ for all a, b, c in G .
- (A3) Identity element: There is an element e in G such that $a.e = e.a = a$ for all a in G .

(A4) Inverse element: For each a in G there is an element a' in G such that
 $a.a' = a'.a = e.$ " [92]

Example 2.1.1. 1. Set of real numbers R forms a group under addition.
 3. Set of integers $\mathbb{Z}_p = \{0, 1, 2, \dots, p-1\}$ under multiplication modulo p .
 2. Set of rational numbers $Q \setminus \{0\}$ is a group under usual multiplication.

Definition 2.1.2. Ring

"A ring R is a set together with two composition laws $+$ and \times such that

1. R is a commutative group with respect to $+$.
2. \times is associative and has a unit element 1, which is different from 0, the unit of $+$.
3. \times is distributive over $+$, that is for all $x, y, z \in R$
 $x(y + z) = xy + xz.$
 $(y + z)x = yx + zx."$ [93]

Definition 2.1.3. Field

"A field K is a commutative ring such that every nonzero element is invertible." [93]

Example 2.1.2. 1. Real numbers R is a field.
 2. The set $\mathbb{Z}_7 = \{0, 1, 2, \dots, 6\}$ under multiplication modulo 7.

Definition 2.1.4. Discrete Logarithm Problem (DLP)

In an equation $y = g^x \pmod{p}$, when g, x, p are given, it is easy to compute y . Many fast and efficient algorithms are available for computing the values of y . But given the values of y, g and p , it is very difficult to compute x . This problem of computing x is called discrete logarithm problem.[92]

Definition 2.1.5. Integer Factorization Problem (IFP)

The decomposition of an integer N (known) in to the prime numbers $\prod p_i^{e_i}$ (unique up to reordering) is called integer factorization problem (IFP). It is an well known and old problem.[94]

2.2 Modular Arithmetic

Most of the cryptographic schemes are based on number theory and arithmetic operations that are performed under a modulo positive integer n . Modular arithmetic is commonly used in public key cryptography. Discussion of its properties are now being described in detail.

Set of Residues \mathbb{Z}_n

The set of residues \mathbb{Z}_n has the non-negative integer values between 0 to $n - 1$, where n is any integer.

$$\mathbb{Z}_n = \{0, 1, 2, 3 \dots n - 1\}$$

The additive inverse of x_1 in the set \mathbb{Z}_n is x_2 if $x_1 + x_2 = 0 \pmod n$. The additive inverse of x_1 under $\pmod n$ is also calculated as $x_2 = n - x_1$. For example the additive inverse of 5 in \mathbb{Z}_{13} is $13 - 5 = 8$

The multiplicative inverse of x_1 in the set \mathbb{Z}_n is x_2 if $x_1 \times x_2 = 1 \pmod n$. Also the multiplicative inverse of x_1 exists in \mathbb{Z}_n if $\gcd(x_1, n) = 1 \pmod n$. For example, in \mathbb{Z}_{12} the multiplicative inverse of 5 exists as $\gcd(5, 12) = 1 \pmod{12}$. The inverse of 4 in \mathbb{Z}_{12} does not exist as $\gcd(4, 12) = 4 \neq 1 \pmod{12}$. Note that the additive inverse is always possible in \mathbb{Z}_{12} but the multiplicative inverse of some members does not exist. The different variants of \mathbb{Z}_n are described below.

1. \mathbb{Z}_n^* : The set of residues \mathbb{Z}_n^* has all those non-negative integer values between 0 to $n - 1$ that are co-prime with n .

$$\mathbb{Z}_n^* = \{x \in \{0, 1, 2, \dots, n - 1\}, \gcd(x, n) = 1\}$$

In \mathbb{Z}_n^* , the additive and multiplicative inverse of all elements exists.

2. \mathbb{Z}_n : The set of residues \mathbb{Z}_n has non-negative integer values between 0 to $n - 1$, where n is any integer.

$$\mathbb{Z}_n = \{0, 1, 2, 3, \dots, n - 1\}$$

In \mathbb{Z}_n , the additive inverse of each element exist, but the multiplicative inverse of non-zero elements may exist.

Definition 2.2.1. Finite Field

Finite field or Galois field is invented by Évariste Galois in 1905. Finite field satisfies the properties of field with only consists of finite number of elements. Usually it is denoted by \mathbb{Z}_n . The most commonly used example of finite field is set of integers under mod p .

Theorem 2.1 (Division Algorithm). *Let n and d be two integers then there exist unique integers q and r such that*

$$n = qd + r,$$

Where q is quotient and $0 \leq r < d$ is remainder. If $r = 0$ then d divides n .

Algorithm 2.2.1 (The Euclidean Algorithm). By repeating the Division Algorithm again and again one can able to compute the greatest common divisor of two integers p and q . This method is known as Euclidean Algorithm and described below.

The Euclidean Algorithm

- 1: **Input:** Positive integer p and q
- 2: **Output:** $\gcd(p, q)$
- 3: $P \leftarrow p, Q \leftarrow q$
- 4: **If** $Q = 0$ **return** $P = \gcd(p, q)$
- 5: $R = P \bmod Q$
- 6: $P \leftarrow Q$
- 7: $Q \leftarrow R$

8: Go to Step 2

Algorithm 2.2.2 (The Extended Euclidean Algorithm). The extension of Euclidean Algorithm is known as Extended Euclidean Algorithm and is used to compute modular inverses. It has vast applications in public key cryptography.

The Extended Euclidean Algorithm

- 1: **Input:** Positive integer p and q such that $p > q$
 - 2: **Output:** The multiplicative inverse of $q \mod p$
 - 3: $(Q_1, Q_2, Q_3) \leftarrow (1, 0, c); (R_1, R_2, R_3) \leftarrow (0, 1, d)$
 - 4: **If** $R_3 = 0$ **return** $Q_3 = \gcd(p, q)$ **no inverse.**
 - 5: **If** $R_3 = 1$ **return** $R_3 = \gcd(p, q); R_2 = q^{-1} \mod p$.
 - 6: $T = \frac{Q_3}{R_3}$ (quotient when Q_3 is divided by R_3)
 - 7: $(S_1, S_2, S_3) \leftarrow (Q_1 - TR_1, Q_2 - TR_2, Q_3 - TR_3)$
 - 8: $(Q_1, Q_2, Q_3) \leftarrow (R_1, R_2, R_3)$
 - 9: $(R_1, R_2, R_3) \leftarrow (S_1, S_2, S_3)$
 - 10: **Go to Step 2**
-

Theorem 2.2 (Euler's Theorem). "Let N and x be integers such that x is coprime to N , then $x^{\phi(N)} \equiv 1 \mod N$. This result was first proved by Fermat when the modulus N is a prime p . In this case, It reduces to $x^{p-1} \equiv 1 \mod p$ for x prime to p . Therefore this restricted version is often referred to as Fermat's little theorem."

[\[95\]](#)

2.3 Hash Function

A value of one way function is easily computed in one direction but it is difficult to compute in its reverse direction. Mathematically, if x is given then it is easy to calculate $f(x)$ but with given $f(x)$ getting x is very hard to compute. A one way function which takes arbitrary length of data and maps it to some fixed length of data is called one way hash function. The output of hash function is named as hash value. The hash values are easily computed but difficult to invert. There are

different forms of cryptographic hash functions depending upon input and output size. The hash functions that are commonly used are Secure hash algorithm (SHA), SHA-1, SHA-2, SHA-3, message digest 4 (MD4) and MD5. Comparison of different hash functions are described in Table 2.1.

Algorithm	Output Size	Block Size	Message size	Rounds	Collision
SHA	160	512	$2^{64} - 1$	80	yes
SHA-1	160	512	$2^{64} - 1$	80	2^{63} Attack
SHA-256	256	512	$2^{64} - 1$	64	No
SHA-224	224	512	$2^{64} - 1$	64	No
SHA-512	512	1024	$2^{128} - 1$	80	No
SHA-384	384	1024	$2^{128} - 1$	80	No

TABLE 2.1: Comparison of Cryptographic Hash Functions

One way hash functions are widely used in cryptography because of its unique

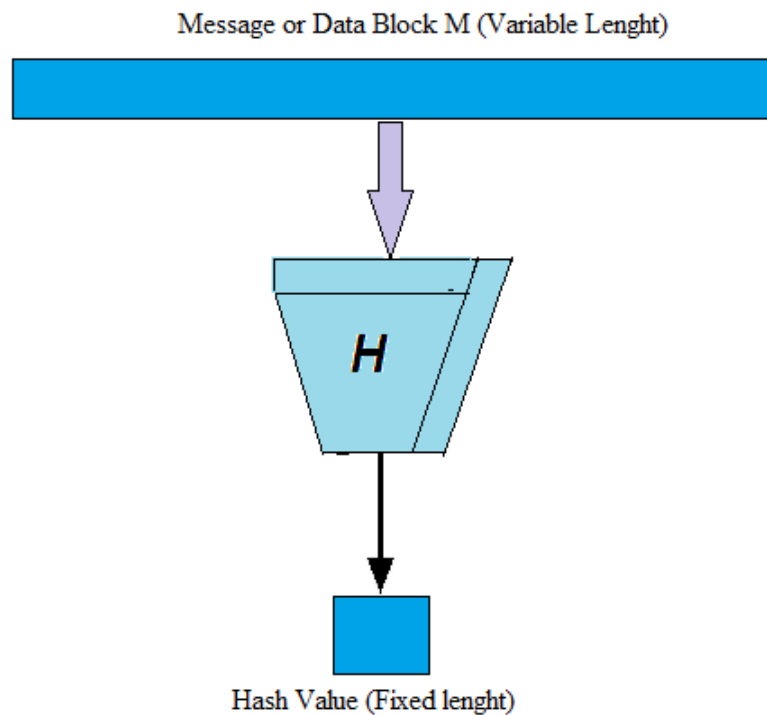


FIGURE 2.1: Cryptographic Hash Function

properties as described below:

1. Efficiency

In a hash function, for given any input, the output (hash value) is easily computed.

2. Pre-image resistance

In a hash function, given any output (hash value) it is infeasible to compute its corresponding input value.

3. Collision resistance

Given any input x_1 , it is infeasible to get another input x_2 such that both inputs have the same hash value.

4. Sensitivity

Small changes in input data produces the major changes in output data.

2.3.1 Types of Hash Functions

Hash functions consists of two types.

1. Keyed Hash Function

The keyed hash function requires the message and the secret key to returns a output called one way keyed hash value.

2. Unkeyed Hash Function

The unkeyed hash function requires only a message as a input and returns a hash value without any secret key.

Elliptic and Hyperelliptic curves are increasingly used in cryptography for last many decades. In next two sections, we will discuss elliptic and hyperelliptic curves in detail.

2.4 Elliptic Curve

The elliptic curve over the real number R and elliptic curve over the finite field \mathbb{F}_p are now discussed in detail.

2.4.1 Elliptic Curve over R

In 1985, Victor Miller and Neal Koblitz independently introduced the elliptic curve cryptography [8]. The points that satisfies the elliptic curve over a real number R have the equation

$$y^2 = x^3 + ax + b \quad (2.1)$$

with additional point \mathcal{O} at infinity as described in Figure 2.2. Here a, b must belongs to R with the property $4a^3 + 27b^2 \neq 0$.

The set E consists of all those points (x, y) satisfying equation 2.1 forms elliptic

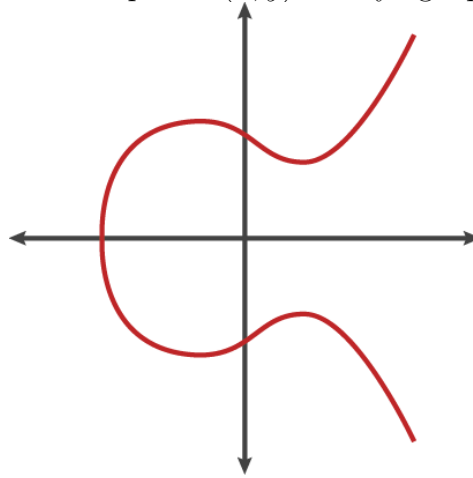


FIGURE 2.2: Elliptic Curve

curve group.

Elliptic Curve Point Addition

Consider two distinct points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ on elliptic curve E . The sum of these two points P_1 and P_2 is denoted by $P_3(x_3, y_3)$ as described in Figure 2.3 and can be calculated by using the following process:

1. Draw a straight line passing through points P_1 and P_2 . This straight line passes through another third point of elliptic curve E .
2. The negative of the third point on the elliptic curve is used as the point addition.

The sum of points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ is denoted by $P_3(x_3, y_3)$ and its co-ordinates are calculated as

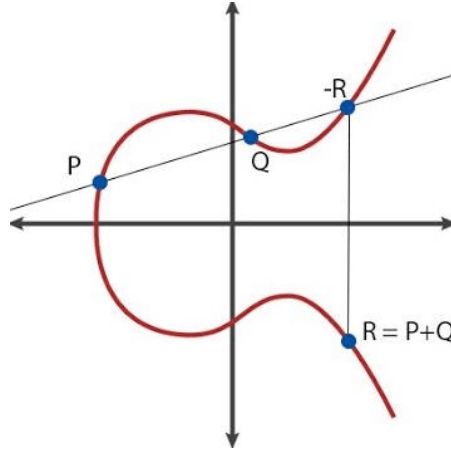


FIGURE 2.3: ECC Point Addition

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

$$y_3 = m(x_1 - x_3) - y_1$$

where,

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

Elliptic Curve Point Doubling

Let $P(x_1, y_1) \in E$ then self addition on elliptic curve $P + P = 2P$ is calculated by using the following process (Figure 2.4):

1. Draw a tangent at point p which passes through the second point of elliptic curve E .
2. The negative of the second point on the elliptic curve E is used as the point doubling.

The sum of point $P(x_1, y_1)$ to itself is denoted by $Q(x_2, y_2)$ and its coordinates are calculated as:

$$x_2 = m^2 - 2x_1$$

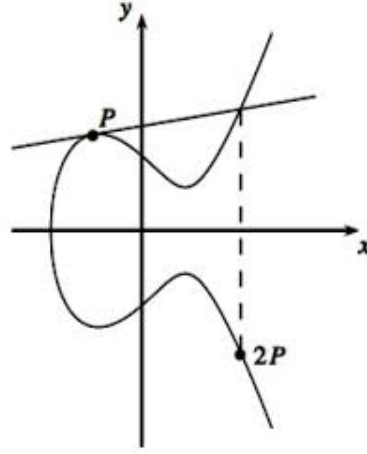


FIGURE 2.4: ECC Point Doubling

$$y_2 = m(x_1 - x_2) - y_1$$

where

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

2.4.2 Elliptic Curve over a Finite Field

The points that satisfies the elliptic curve over a finite field \mathbb{F}_p have the equation

$$y^2 = x^3 + ax + b \mod p \quad (2.2)$$

with additional point \mathcal{O} at infinity. Here a, b must belongs to \mathbb{F}_p with the property $4a^3 + 27b^2 \neq 0$. The set $E_p(a, b)$ consists of all those points (x, y) satisfying equation 2.2 forms elliptic curve group modulo p . The elements of $E_p(a, b)$ also forms a cyclic group and is generated by single element called the base point G . The smallest non-negative integer n is called order of G such that $nG = \mathcal{O}$ (infinity). Domain parameters of E are (a, b, p, G, n) . According to HASSE theorem [96], the number of points on elliptic curve E defined over \mathbb{F}_p are restricted in the interval $p + 1 - 2\sqrt{p} \leq |E| \leq p + 1 + 2\sqrt{p}$. For further details on elliptic curves, we refer to [92, 93, 97]. The set of elliptic curve points forms a abelian group under addition. Consider two distinct points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ on elliptic curve E .

1. For point addition of $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$, draw a straight line passing through these points. This straight line passes through another third point of elliptic curve E . The point $P_3(x_3, y_3)$ is negative of the third point on the elliptic curve is used as the point addition. This addition can algebraically be calculated as:

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

$$y_3 = m(x_1 - x_3) - y_1 \mod p$$

$$m = \frac{y_2 - y_1}{x_2 - x_1} \mod p$$

2. For point doubling of $P_1(x_1, y_1)$, draw a tangent at point $P_1(x_1, y_1)$ that passes through the second point of elliptic curve E . The point $P_2(x_2, y_2)$ which is negative of the second point on the elliptic curve E is used as the point doubling and its coordinates are calculated as:

$$x_2 = m^2 - 2x_1$$

$$y_2 = m(x_1 - x_2) - y_1$$

where,

$$m = \frac{3x_1^2 + a}{2y_1} \mod p$$

Elliptic Curve Discrete Logarithm Problem

Elliptic curve discrete logarithm problem (ECDLP) is, given points A and B in elliptic curve E , finding the integer k such that $kA = B$. The number k is then called discrete logarithm of B to the base A . It is computationally infeasible to find k when both A and B are known. The entire security of ECC depends upon ECDLP. This problem is used to define elliptic curve Diffie-Hellman protocol (ECDH)[98].

Example 2.4.1. Points on elliptic curve Consider an elliptic curve $y^2 = x^3 + 5x - 12 \mod 73$ over a finite field \mathbb{F}_{73} . All 64 points on the elliptic curve are given

in Table 2.2.

(0,34)	(0,39)	(1,33)	(1,40)	(2,15)	(2,58)	(4,27)	(4,46)	(5,24)
(5,49)	(7,1)	(7,72)	(9,18)	(9,55)	(10,4)	(10,69)	(12,30)	(12,43)
(16,21)	(16,52)	(18,17)	(18,56)	(23,15)	(23,58)	(27,13)	(27,60)	(29,33)
(29,40)	(30,36)	(30,37)	(31,2)	(31,71)	(35,25)	(35,48)	(37,23)	(37,50)
(38,9)	(38,64)	(43,33)	(43,40)	(44,36)	(44,37)	(48,15)	(48,58)	(53,8)
(53,65)	(56,10)	(56,63)	(57,22)	(57,51)	(61,5)	(61,68)	(66,11)	(66,62)
(68,35)	(68,38)	(69,14)	(69,59)	(70,26)	(70,47)	(72,36)	(72,37)	\mathcal{O}

TABLE 2.2: Points of the Elliptic Curve E defined over the Finite Field \mathbb{F}_{73}

2.5 Hyperelliptic Curve

A hyperelliptic curve C of genus g defined over a finite field \mathbb{F}_q is of the form:

$$C : y^2 + h(x)y = f(x) \quad (2.3)$$

where $h(x)$ and $f(x)$ are polynomials with coefficients in \mathbb{F}_q (Figure 2.5). The degree of $h(x)$ is at most g and $f(x)$ has degree atleast $2g+1$. For non-singularity, no points of curve C should simultaneously satisfy the equations: $2y + h(x) = 0$ and $h' - f'(x) = 0$. The number of non intersecting curves drawn on surface

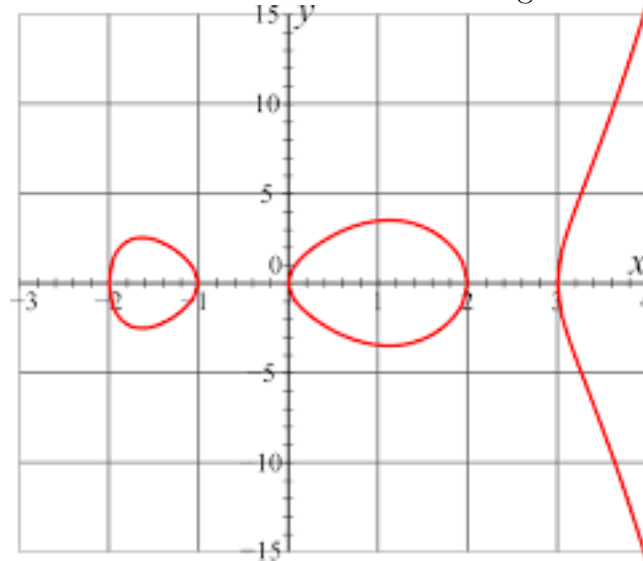


FIGURE 2.5: Hyperelliptic Curve

without touching each other is called genus of curve. It decides the computational

time involved in the implementation aspects of the curve. A curve of genus 2 is considered to be suitable for secure and efficient computations. Following are the curves of different genus.

1. $y^2 = x^3 + x + 1$ has genus one and called elliptic curve.
2. $y^2 + xy = x^5 + b_1x^3 + b_2x^2 + b_3x + b_4$ has genus 2 and $h(x) = x$.
3. $y^2 = x^9 + b_1x^7 + b_2x + b_3$ has genus 4. For further details on hyperelliptic curves, we refer to [93, 97].

Special, Opposite and Ordinary point

Let $P(x, y)$ be any point on the hyperelliptic curve (2.3) then its opposite point on the curve is $\bar{P}(x, -y - h(x))$. A point \mathcal{O} is called point at infinity and its opposite point is denoted by $\bar{\mathcal{O}}$ such that $\mathcal{O} = \bar{\mathcal{O}}$. A point P is called special if $P = \bar{P}$ otherwise called ordinary point.

Consider the hyperelliptic curve $y^2 + xy = x^5 + 2x^4 + x^3 - 5x^2 + 10$ defined over \mathbb{Z}_{11} . The points lie on the curve are

$(1, 6), (1, 4), (5, 10), (5, 7), (4, 5), (4, 2), (8, 3), (8, 0), (9, 5), (9, 8)$.

Here $P = \bar{P} = (8, 0)$ is a special point and rest of the points are ordinary.

2.5.1 Divisor

The divisor $D = \sum m_n P$ for the hyperelliptic curve H is an arbitrary linear combination of distinct points $P_1, P_2, P_3, \dots, P_n$ on C and $m_1, m_2, m_3, \dots, m_n \in \mathbb{Z}$ with only some $m_n = 0$. The integer $\deg(D) = \sum m_n$ is called the degree of the divisor D . The order of the divisor D is an integer $m_n = \text{ord}_P(D)$.

If $P(x_1, y_1)$ be any point on the hyperelliptic curve then the divisor of this point is calculated as

$$D = \begin{cases} P + \bar{P} - 2\infty & P \neq \bar{P} \\ P - 2\infty & P = \bar{P} \end{cases}$$

Consider different points P_1, P_2 and Q_1, Q_2 on hyperelliptic curve H . By using the interpolation, find a curve that passes through these four points and also two additional points R_1', R_2' on the hyperelliptic curve. The reflection of these two additional points on the curve are R_1 and R_2 as shown in Figure 2.6. These divisors are represented as

$$D_1 = P_1 + P_2 - 2\infty$$

$$D_2 = Q_1 + Q_2 - 2\infty$$

Third divisor from these divisors are calculated as

$$D_3 = (P_1 + P_2 - 2\infty) + (Q_1 + Q_2 - 2\infty) = R_1 + R_2 - 2\infty$$

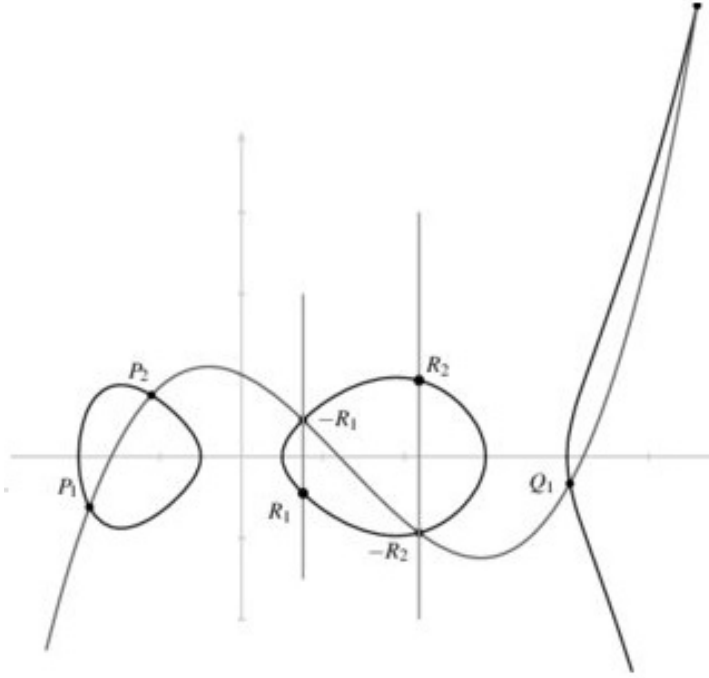


FIGURE 2.6: Geometrical Representation of Divisor

Definition 2.5.1. Divisor Group

The collection of all divisors of a hyperelliptic curve H forms a group under addition and is denoted by $D = D(H)$. The addition is defined as

$$\sum m_i P_i + \sum n_i P_i = \sum (m_i + n_i) P_i.$$

The subgroup which contains the divisors of degree zero is denoted by D^0 .

Hyperelliptic Curve Discrete Logarithm Problem

Let D_1 and D_2 are two divisors of hyperelliptic curve H then finding the integer c , such that $cD_1 = D_2$, is a hyperelliptic curve discrete logarithm problem (HECDLP). It is computationally infeasible to find c . The entire security of HECC depends upon HECDLP.[93, 97]

G.C.D of divisors

The greatest common divisor of $D_1 = \sum m_i P_i$ and $D_2 = \sum n_i P_i$ is defined as

$$\gcd(D_1, D_2) = \sum \min(m_i, n_i) P_i$$

Definition 2.5.2. Semi-reduced divisor

A divisor $D = \sum m_i p_i - (\sum m_i) \infty$ is called semi reduced divisor if

1. $m_i \geq 0$ for each $i \in N$.
2. If $p = \bar{p}$ then each $m_i = 1$.
3. Only p or \bar{p} are used in the sum if $p \neq \bar{p}$.

Definition 2.5.3. Reduced divisor

A semi reduced divisor $D = \sum m_i p_i - (\sum m_i) \infty$ is called reduced divisor if $\sum m_i \leq g$. The group operation is performed on the reduced divisors for the practical implementation of HECC.

2.5.2 Mumford represenation

For implementation point of view, working with divisors is not easy. Cantor [99] used Mumford representation of the divisors for efficient computation and representation of the divisors.

Let $D = \sum m_i p_i - (\sum m_i) \infty$ be a semi reduced divisor in which $p_i(x_i, y_i) \in H$ and $\alpha(u) = \prod (u - x_i)^{m_i}$. A unique polynomial $\beta(u)$ has the following properties:

1. $\deg \alpha > \deg \beta$
2. $\beta(x_i) = y_i$ for each i such that $m_i \neq 0$.
3. $\alpha(u)$ divides $(\beta(u))^2 + \beta(u)h(u) - f(u)$

Then $D = \gcd(\text{div}(\alpha(u)), \text{div}(\beta(u)h(u)) - y)$ is the Mumford representation of the divisor D and can also be written as $\text{div}(a, b)$.

Example 2.5.1. Consider the hyperelliptic curve $y^2 = x^5 + 3x^4 - 7x^3 - 27x^2 - 18x \pmod{11}$. Let $D_1 = (3, 0) + (2, 1)$ be a divisor. The unique polynomials associated to D_1 are $\alpha(x) = (x - 3)(x - 2) = x^2 - 5x + 6$ and $\beta(x) = 10x + 3$. So polynomial representation of divisor D_1 is $(x^2 - 5x + 6, 10x + 3)$

Similarly the polynomials associated to $D_2 = (4, 2) + (2, 1)$ are $\alpha(x) = (x - 4)(x - 2) = x^2 - 6x + 8$ and $\beta(x) = 6x$. So polynomial representation of D_2 is $(x^2 - 6x + 8, 6x)$. The degree of both the polynomials is 2 which is equal to g .

2.5.3 Cantors Algorithm

Cantor [99] in 1987 introduced a method of finding the sum of two reduced divisors on hyperelliptic curve. Mumford representation of the divisors is used in the cantors algorithm. It consists of two phases namely composition and reduction of divisors. In composition phase, a new divisor is calculated which is the sum of two input divisors.

Composition Algorithm

- Input: Two reduced divisors $D_1 = (\alpha_1, \beta_1)$ and $D_2 = (\alpha_2, \beta_2)$ of a hyperelliptic curve H .
- : Output: A semi-reduced divisor $D = \text{div}(\alpha, \beta)$ such that $D \sim D_1 + D_2$

The Cantors algorithm [99] is described in following steps.

1. Compute the polynomials $d_1, t_1, t_2 \in F_q[x]$ by using the Extended Euclidean Algorithm as

$$d_1 = \gcd(\alpha_1, \alpha_2)$$

$$d_1 = t_1\alpha_1 + t_2\alpha_2$$

2. Again using the Extended Euclidean Algorithm calculate the polynomials $d, \gamma_1, \gamma_2 \in F_q[x]$ as

$$d = \gcd(d_1, \beta_1 + \beta_2 + h)$$

$$d = \gamma_1 d_1 + \gamma_2 (\beta_1 + \beta_2 + h)$$

3. Compute

$$a_1 = \gamma_1 t_1$$

$$a_2 = \gamma_1 t_2$$

$$a_3 = \gamma_2$$

$$d' = a_1 d_1 + a_2 \alpha_2 + a_3 (\beta_1 + \beta_2 + h)$$

4. Compute

$$\alpha = \frac{\alpha_1 \alpha_2}{d'^2}$$

$$\beta = \frac{a_1 \alpha_1 \beta_2 + a_2 \alpha_2 \beta_1 + a_3 (\beta_1 \beta_2 + f)}{d'}$$

Reduction Algorithm

- Input: A semi-reduced divisor $D = \text{div}(\alpha, \beta)$
- Output: A unique reduced divisor $D' = (a', b')$ such that $D' \sim D$

The reduction algorithm is described in following steps.

1. Compute

$$\begin{aligned}\alpha' &= \frac{f - \beta h - \beta^2}{\alpha} \\ \beta' &= -h - \beta \pmod{\alpha'}\end{aligned}$$

2. If $\deg(\alpha') > \deg(g)$ then $\alpha = \alpha'$ and $\beta = \beta'$, and repeat Step 5 until $\deg(\alpha') < \deg(g)$.
3. Make α' monic through dividing its leading coefficient.
4. Get the unique reduced divisor $D' = (\alpha', \beta')$

Example 2.5.2. Consider the hyperelliptic curve $y^2 = x^5 + 3 \pmod{7}$ of genus 2. Let $P_1 = (1, 2)$, $P_2 = (3, 1)$, $P_3 = (3, 6)$ and $P_4 = (6, 3)$ be four different points on the curve. Consider the divisor $D_1 = P_1 + P_2 - 2\infty$, $D_2 = P_1 + P_3 - 2\infty$, $D_3 = P_1 + P_4 - 2\infty$. The Mumford representation of above mentioned divisors are

$$\begin{aligned}D_1 &= (\alpha_1, \beta_1) = (y^2 + 3y + 3, 3y + 6) \\ D_2 &= (\alpha_2, \beta_2) = (y^2 + 3y + 3, 2y) \\ D_3 &= (\alpha_3, \beta_3) = (y^2 + 6, 3y + 6)\end{aligned}$$

These divisors are reduced and cantors algorithm is used to compute the sum of divisors. For computation of $D_1 + D_2$, proceed as follows.

1. Compute the polynomials $d_1, t_1, t_2 \in F_q[x]$ by using the Extended Euclidean Algorithm as

$$\begin{aligned}d_1 &= \gcd(\alpha_1, \alpha_2) \\ d_1 &= \gcd(y^2 + 3y + 3, y^2 + 3y + 3) \\ d_1 &= y^2 + 3y + 3\end{aligned}$$

Also d_1 is represented by the linear combination of α_1 and α_2 .

$$\begin{aligned} d_1 &= t_1\alpha_1 + t_2\alpha_2 \\ y^2 + 3y + 3 &= t_1(y^2 + 3y + 3) + t_2(y^2 + 3y + 3) \end{aligned}$$

We get $t_1 = 1$ and $t_2 = 0$.

2. Compute γ_1 and γ_2 as follows.

$$\begin{aligned} d &= \gcd(d_1, \beta_1 + \beta_2 + h) \\ d &= \gcd(y^2 + 3y + 3, 5y + 6) \end{aligned}$$

Also d can be written as a linear combination of γ_1 and γ_2 .

$$\begin{aligned} d &= \gamma_1 d_1 + \gamma_2(\beta_1 + \beta_2 + h) \\ 5y + 6n &= \gamma_1(y^2 + 3y + 3) + \gamma_2(5y + 6) \end{aligned}$$

So we get $\gamma_1 = 0$ and $\gamma_2 = 1$.

3. Compute

$$a_1 = \gamma_1 t_1 = 0$$

$$a_2 = \gamma_1 t_2 = 0$$

$$a_3 = \gamma_2 = 1$$

$$d' = a_1 d_1 + a_2 \alpha_2 + a_3(\beta_1 + \beta_2 + h)$$

$$d' = 0 + 0 + 1(5y + 6)$$

$$d' = 5y + 6$$

$$\alpha = \frac{\alpha_1 \alpha_2}{d^2}$$

$$\alpha = \frac{(y^2 + 3y + 3)^2}{(5y + 6)^2}$$

$$\alpha = 2y^2 + 3y + 2$$

$$\beta = \frac{a_1 \alpha_1 \beta_2 + a_2 \alpha_2 \beta_1 + a_3(\beta_1 \beta_2 + f)}{d} \mod \alpha$$

$$\beta = \frac{y^5 + 6y^2 + 5y + 3}{3y + 6} \mod 2y^2 + 3y + 2$$

$$\beta = 3y + 6$$

$$\alpha' = \frac{f - \beta h - \beta^2}{\alpha}$$

$$\alpha' = \frac{y^5 + 5y^2 + 6y + 2}{2y^2 + 3y + 2}$$

$$\alpha' = 4y^3 + y^2 + 5y + 1$$

$$\beta' = -h - \beta \mod \alpha'$$

$$\beta' = 4y + 1$$

4. As $\deg(\alpha') > g$ so put $\alpha = \alpha' = 4y^3 + y^2 + 5y + 1$ and $\beta = \beta' = 4y + 1$ and repeating the Step 5.

$$\begin{aligned}
 \alpha' &= \frac{f - \beta h - \beta^2}{\alpha} \\
 \alpha' &= \frac{y^5 + 3 - (4y + 1)^2}{4y^3 + y^2 + 5y + 1} \\
 \alpha' &= 2y^2 + 3y + 2 \\
 \alpha' &= y^2 + 5y + 1 \\
 \beta' &= -h - \beta \pmod{\alpha'} \\
 \beta' &= -4y - 1 \pmod{y^2 + 5y + 1} \\
 \beta' &= 3y + 6
 \end{aligned}$$

5. $D' = (a', b') = (y^2 + 5y + 1, 3y + 6)$ is the sum of the divisors D_1 and D_2 .

For cryptographic point of view, divisors are used in the computations of hyper-elliptic curve based cryptosystems.

Chapter 3

Overview of Cryptography

3.1 Cryptography

Cryptology is the scientific way of generating and solving secret codes. It deals with the comprehensive study of cryptography and cryptanalysis. Cryptography is the knowledge of secret communication in the presence of unauthorized third party. In cryptography, the original message is called cleartext or plaintext. The process of converting a cleartext message in to the coded form to hide its meaning from others is known as encryption. The encrypted message is known as ciphertext. The method of regenerating the cleartext message from ciphertext is known as decryption. The secret key is used in encryption and decryption process. The typical cryptosystem is described in Figure 3.1. Depending upon the keys used, cryptography is divided in to two branches; namely public (Asymmetric) key cryptography and private (Symmetric) key cryptography.

3.1.1 Private (Symmetric) Key Cryptography

In private key cryptography, same secret key is used in the process of encryption and decryption. The benefits of symmetric key cryptography are its fast and simple computations and disadvantage is that security relies upon each participant

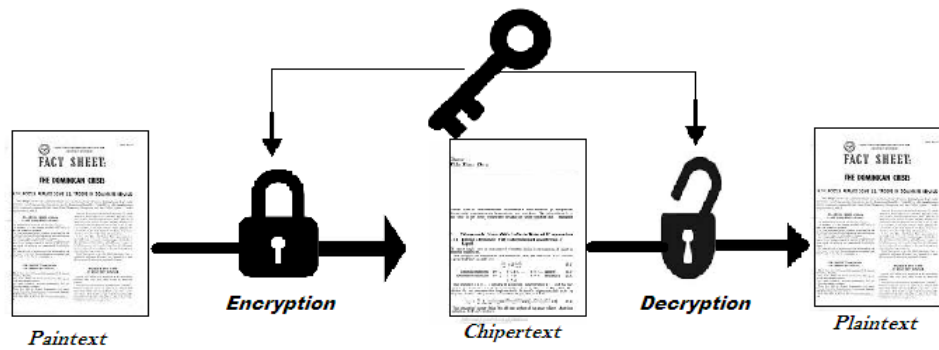


FIGURE 3.1: A Typical Cryptosystem

involves in the communication to keep the secret keys confidential. The well known examples of private key cryptography are Data Encryption Standard (DES) [100] and Advanced Encryption Standard (AES) [101]. The typical symmetric encryption model is described in Figure 3.2.

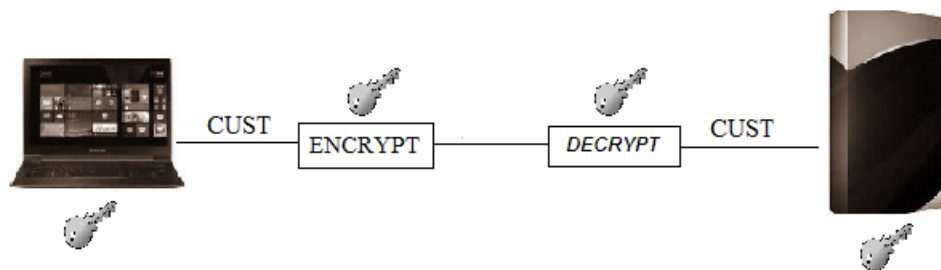


FIGURE 3.2: Symmetric Encryption Model

3.1.2 Public (Asymmetric) Key Cryptography

In 1976, Whitefield Diffie and Martin Hellman [98] introduced the public key cryptography. They introduced a new mechanism that involves two different keys, one is called a public key and is known to everybody, and the other key is kept secret

by the owner and is known as a private key. The benefit of public key cryptography is to overcome the key sharing issue and disadvantage of complex and slow computations. The well known examples are ECC [8], EL-Gamal [9] and RSA [10]. The asymmetric encryption model is described in Figure 3.3.

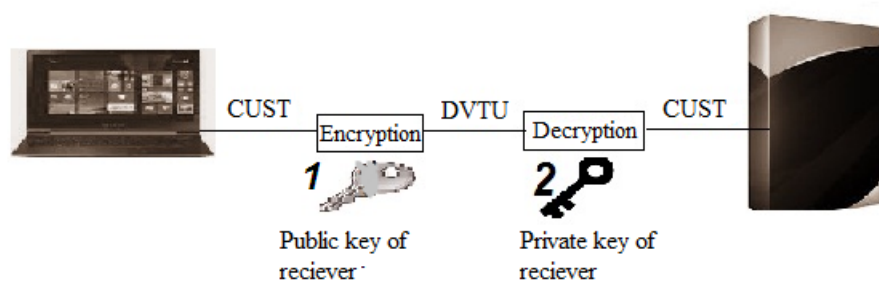


FIGURE 3.3: Asymmetric Encryption Model

In public key cryptography, digital signature is used for providing the authentication of data and sender. In next section, digital signature and its different variants will discuss in detail.

3.2 Digital Signature

A mathematical code that is attached to an electronic document for its verification is known as digital signature. The digital signature gives the assurity that the message is not tampered or replaced during the communication. Without the sender's private key, it is computationally infeasible to develop a valid signature. The digital signature process is described in Figure 3.4.

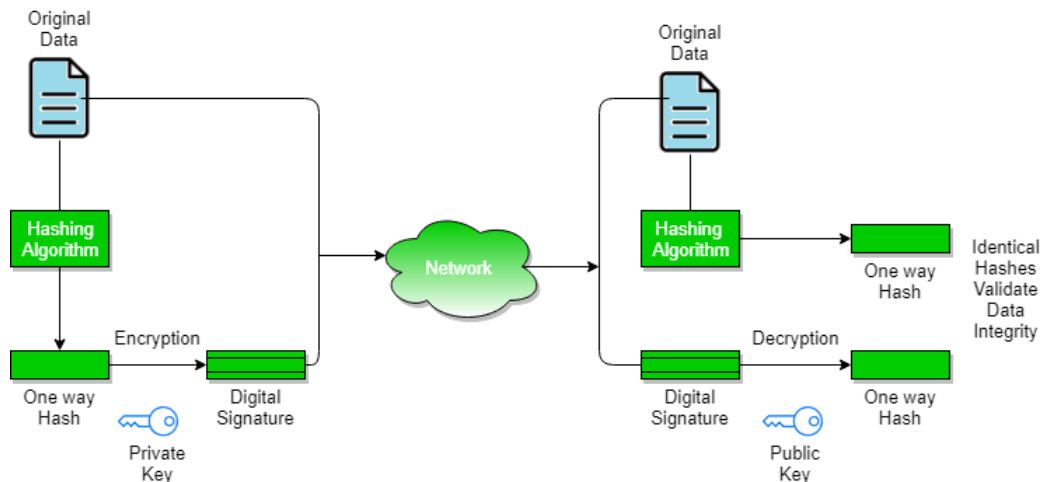


FIGURE 3.4: Digital Signature Model

Suppose Alice (sender) wants to produce a digital signature for a document and sends it to Bob. The process of generating the digital signature is described in the following steps:

Alice

1. Selects a digital document to be signed.
2. Generates the hash value of this document.
3. To compute the digital signature, she uses her private key to encrypt the hash value.
4. Sends the original digital document as well as its signature to Bob.

Bob

1. To decrypt the digital signature, Bob will use the public key of Alice to get the hash value that was computed at Alice end.
2. Computes the hash value of a document that is received from Alice.
3. Accepts the digital document as a valid if the hash value computed in Step 1 and Step 2 are same.
4. If these two hash values are different then believes that the received document is tampered or replaced during the transmission.

3.2.1 Blind Signature

In 1983, Chaum [3] introduced a first blind signature scheme for sender's privacy. The proposed scheme is based on RSA algorithm [10] and involves three participants: a signer, a requester and a receiver. In a blind signature scheme, the power of the signature is delegated to single signer and the document is blinded before transmits it to the signer's end. In the proposed scheme, first the sender blinds the digital document and sends this document to the signer end. After receiving the blinded message, the signer signs the digital document without reading the contents of the original message and resend it to the sender. After this, the sender unblinds the signed document and transmits it to the receiver's end after encrypting. The blind signature process is described in Figure 3.5.

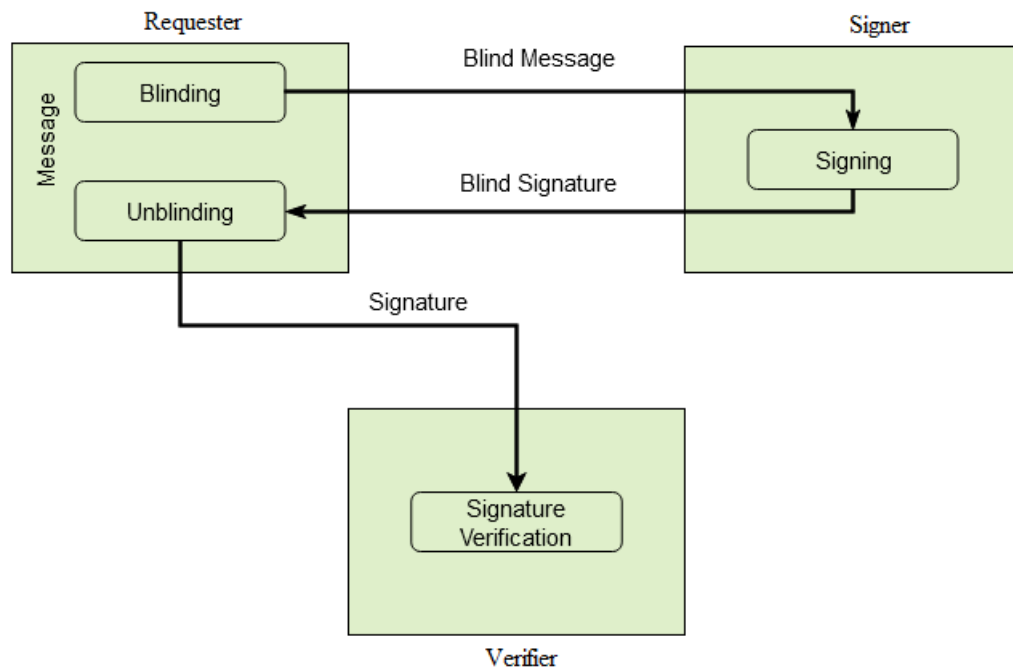


FIGURE 3.5: Blind Signature Model

Any blind signcryption scheme provides two additional properties of anonymity and untraceability in-addition to the properties that are offered in any signature scheme.

Sender anonymity

It is a signature protocol that allows user to transmit a signed messages between the signer and the user in such a way that the signer is not able to read the contents

of the original message.

Untraceability

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

The blind signature scheme due to these properties can be used mostly in e-voting, e-cash payment and e-bidding [4].

3.2.2 Aggregate Signature

For multiple digital documents, signing each document separately requires extra computational and communication cost in the process of digital signature. To overcome this issue, in 2003, Boneh et al. [56] introduced a new signature scheme named “Aggregate Signature” which minimizes the length of certificate chains. In their scheme, there are n distinct messages and n distinct users. Then aggregating all distinct n signatures to a single short signature in such a way that each user assures the authenticity of the received message. A single signature generated from the multiple documents is required to be correctly generated and accepted for authenticating multiple digital documents. This single blind signature is generated from all the multiple messages and cannot be verified if only some of the messages are known. So verification of the corresponding single digital signature requires all the multiple digital documents. The aggregate signature process is described in Figure 3.6.

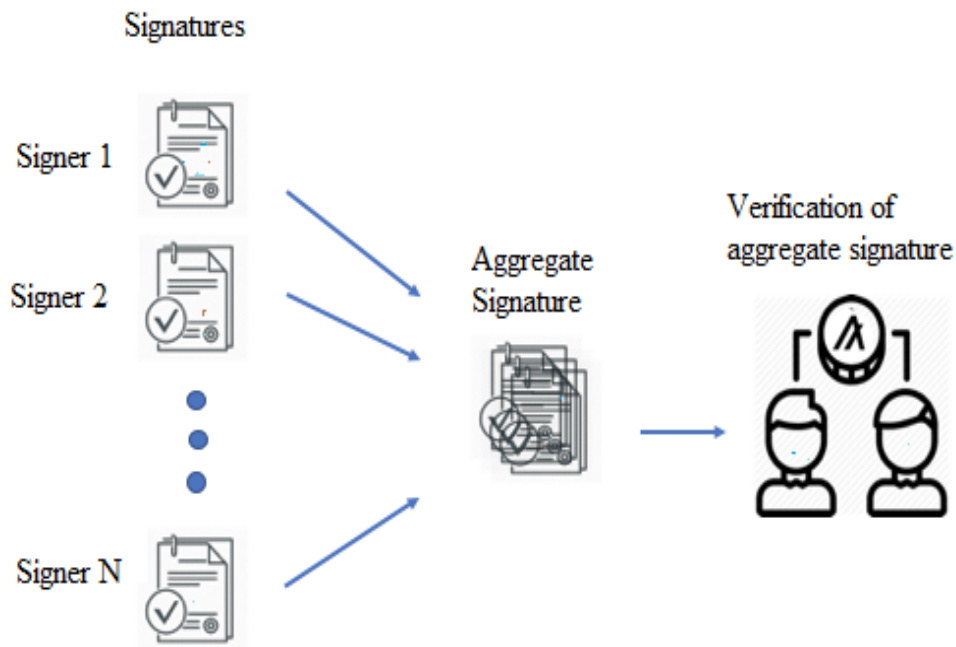


FIGURE 3.6: Aggregate Signature Model

The authentication of public key is an essential requirement of any public key cryptosystem. A Certificate Authority(CA) generates and issues the digital signature. It's a reliable third party between the owner of the public key and a party that depends upon the certificate. It gives the assurity to any party involved in the communication to believe that the specific public key is related to the user who claimed it.

The well known public key schemes are RSA [10], ELGamal [9] and ECC [8]. In 1978, Rivest, Shamir, and Adleman [10] introduced a first practical public key encryption and signature scheme called RSA. The security of RSA depends upon the complexity of factoring large integers known as the integer factorization problem (IFP). It is well known public key cryptosystem, which is used in many application to provide data security. Taher Elgamal [9] introduced a new public key cryptosystem like RSA in 1985. It uses the Diffe Hellman key exchange protocol to build a new encryption and decryption algorithm. The security of this cryptosystem depends upon discrete logarithm problem (DLP). A brief description of Elliptic curve cryptography (ECC) [8] is described in next section.

3.3 Elliptic Curve based Cryptosystem

In 1985, Elliptic curve cryptography (ECC) [8] was invented by Victor Miller and Neal Koblitz. The entire security of ECC relies upon ECDLP. ECC has many advantages over the existing cryptosystems like RSA [10] and Elgamal [9]. The attacks on elliptic curve are weaker than the available attacks on these mentioned cryptosystem. ECC uses the smaller keys in comparison of the other public key cryptosystems like Elgamal [9] and RSA [10] with same level of security. The comparison of ECC with RSA and Elgamal is described below in Table 3.1.

Bits of Security	RSA and DH Key Size	EC Key Size
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

TABLE 3.1: Comparison of Key Size Of ECC and RSA [102]

Elliptic curve cryptography protocols are standardized by different standard organizations such as:

- National Institute of Standard and Technology (NIST)
- American National Standard Institute (ANSI)
- Institute of Electrical and Electronics Engineers (IEEE)
- Internet Engineering Task Force (IETF)
- International Organization for Standardization (ISO)
- Standard for Efficient Cryptography Group (SECG)
- Federal Information Processing Standards (FIPS)

NIST recommended to use standard elliptic curves offering different level of security for elliptic curve digital signature algorithm (ECDSA) in FIPS 186-4 (United States Department of Commerce (NIST), 2013). Five elliptic Curves $y^2 = x^3 + ax + b \pmod p$ defined over the finite field \mathbb{F}_p are recommend to use: P-256, P-224,

P-192, P-521 and P-384 [103].

The computations on elliptic curve consists of four layers, each layer offer different level of computational cost.

1. The first modular arithmetic layer consist of basic modular arithmetic operations (In Section 2.1.1) which are computationally most costly.
2. The second group operation layer consists of point doubling and addition (In Section 2.3) and it has less computational cost as compared to first layer.
3. The third point multiplication layer can be implemented using the Double-and-Add method and has less computational cost as compared to first and second layer.
4. Fourth upper layer protocols like ECDH and ECDSA has less computational cost as compared to all the first three layers.

Most struggle should go in optimization of the modular arithmetic operations like modular subtraction, modular addition, modular inversion and modular multiplication.

Elliptic Curve Diffie-Hellman Key Exchange (ECDH)

The Diffie-Hellman key exchange protocol [98] is used to share the common secret key K between the sender (Alice) and the receiver (Bob). The key sharing protocol for elliptic curve uses the computations of elliptic curve to share the common key between the sender and the receiver and known as elliptic curve Diffie-Hellman key exchange protocol (ECDH). For the generation of common key, both Alice and Bob agreed on the elliptic curve E and the base point G . The key exchange protocol is described in Figure 3.7:

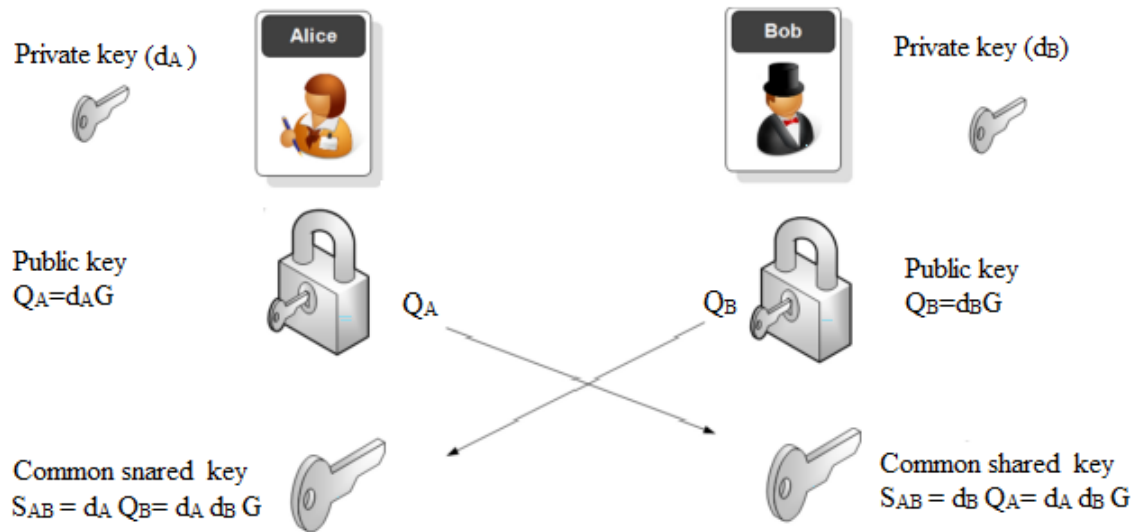


FIGURE 3.7: Diffie Hellman Key Exchange Protocol

Elliptic curve based cryptosystem

We describe the cryptosystem of [23] that uses the computations of elliptic curve. The receiver public key is used for encryption of data. The ciphertext is transmitted in the form of elliptic curve point. The receiver's private key is used for decryption of data. Consider a base point G of an elliptic curve E . The proposed scheme [23] is described below:

Key Generation

The participants select their private key and then compute the public keys as:

Alice

- Randomly chooses $z_A < n$ as her private key.
- Computes public key $X_A = z_A G$ as a point on the elliptic curve.

Bob

- Randomly chooses $z_B < n$ as her private key.
- Computes public key $X_B = z_B G$ as a point on the elliptic curve.

Encryption

Suppose Alice generates and transmits a message M to Bob through unsecured public network. She first chooses a random number $p \leq n$ and then ciphertext message C is generated with the help of public key X_B of Bob.

$$C = \{pG, M + pX_B\}$$

Sends C to Bob.

Decryption

After receiving the ciphertext message C , Bob multiplying the first part of ciphertext with receiver's private key z_B .

$$C = \{z_B pG, M + pX_B\}$$

$$C = \{pX_B, M + pX_B\}$$

The plaintext message M is obtain by subtracting the second part of ciphertext from the first part as;

$$M = (M + pX_B) - (pX_B)$$

For providing the encryption and authentication in the public key schemes, we discuss signature-then-encryption and signcryption models in next two sections.

3.4 Signature-then-Encryption

The secure and safe communication is the basic requirement of any public key scheme. It ensures that the transmitted data is not disclosed and tampered by the unauthorized third parties during the communication. Digital signature and

encryption are two basic tools of cryptography that gives the guarantee of authentication and confidentiality. In traditionally used signature-then-encryption technique, the task of both authentication and encryption is fulfilled by first signing the digital document and then signed document is encrypted for transmission in unsecured public network. The signature-then-encryption model is described in Figure 3.8. The Figure shows that first the sender of a message would sign the message with digital signature scheme and then encryption is performed with the help of private key cryptography. The encryption key is then encrypted by using the recipient public key and then sends it to the receiver end. On the receiver end, first asymmetric decryption is performed with the receiver's private key to get the encryption key and then use it to verify the authenticity of received message.

In this approach, the process of encryption and generation of signature requires

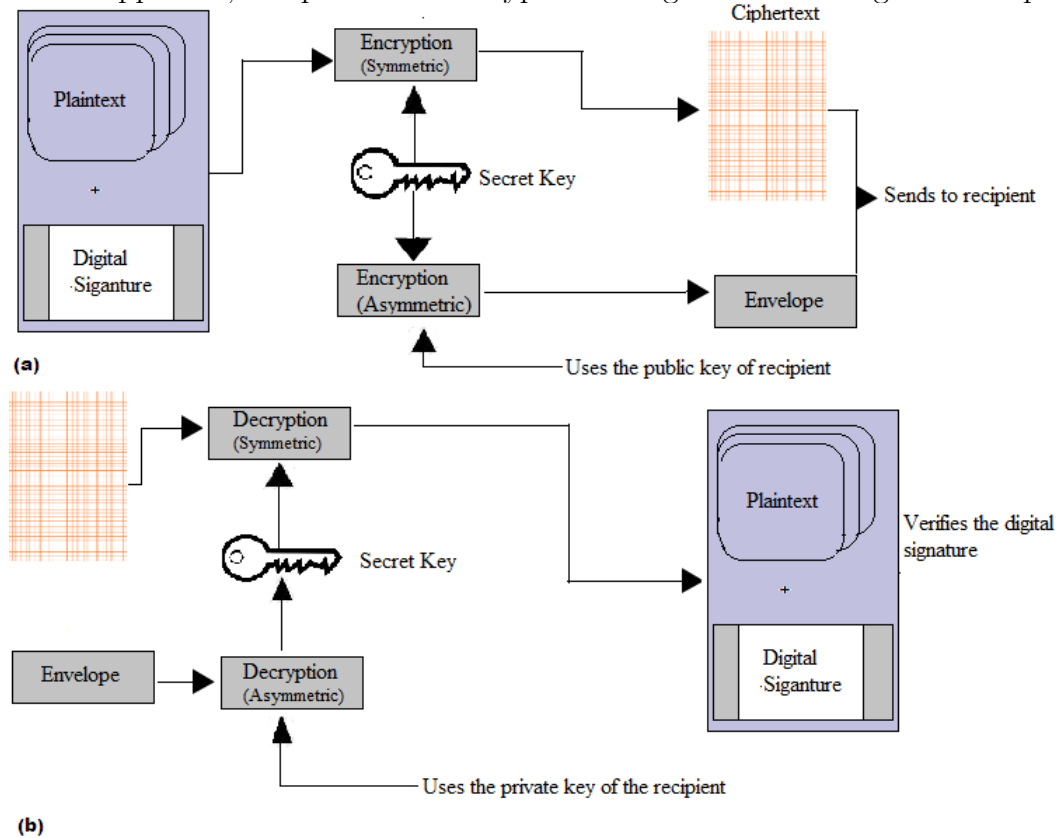


FIGURE 3.8: Sign-then-Encryption Model

extra machine cycles and more bits are added in the original message. The same amount of operations are required in decryption and signature verification process. So it has the drawbacks of low efficiency and high computational cost.

3.5 Signcryption

In 1997, Zheng [20] introduced a new cryptographic technique called “Signcryption”, that combines the role of both digital signature and encryption in a single step. In signcryption scheme [20], the sender drives his secret key with the help of receiver’s public key. The encryption is performed by using the private (symmetric) key cryptography with the help of common shared secret key. After receiving the signcrypted data, receiver gets the same secret key by using his private key. Signcryption has benefits of less computational and communication cost in comparison with the signature-then-encryption technique. Zheng’s [20] analysis shows that signcryption scheme reduces 50% computational overheads and 85% computational cost as compared to traditionally used signature-then-encryption scheme. The signcryption model is described in Figure 3.9. Any signcryption scheme mainly involves three algorithms;

1. Key Generation Algorithm: A key generation algorithm chooses a uniform random private keys and generates their corresponding public keys.
2. Signcryption Algorithm: A signcryption algorithm takes a plaintext message, the sender’s private key and public key of the receiver to produce a ciphertext message and signature (called signcrypted text).
3. Unsigncryption Algorithm: An unsigncryption algorithm takes the receiver’s private key and sender’s public key to get the plaintext message and then confirms its authenticity.

3.5.1 Security Attributes of Signcryption Scheme

Digital signature and encryption are two basic security properties of any signcryption scheme. Such properties include unforgeability, confidentiality, integrity and non-repudiation. Public verifiability and forward secrecy are additional features that are provided depending upon the requirements. These security attributes are described below:

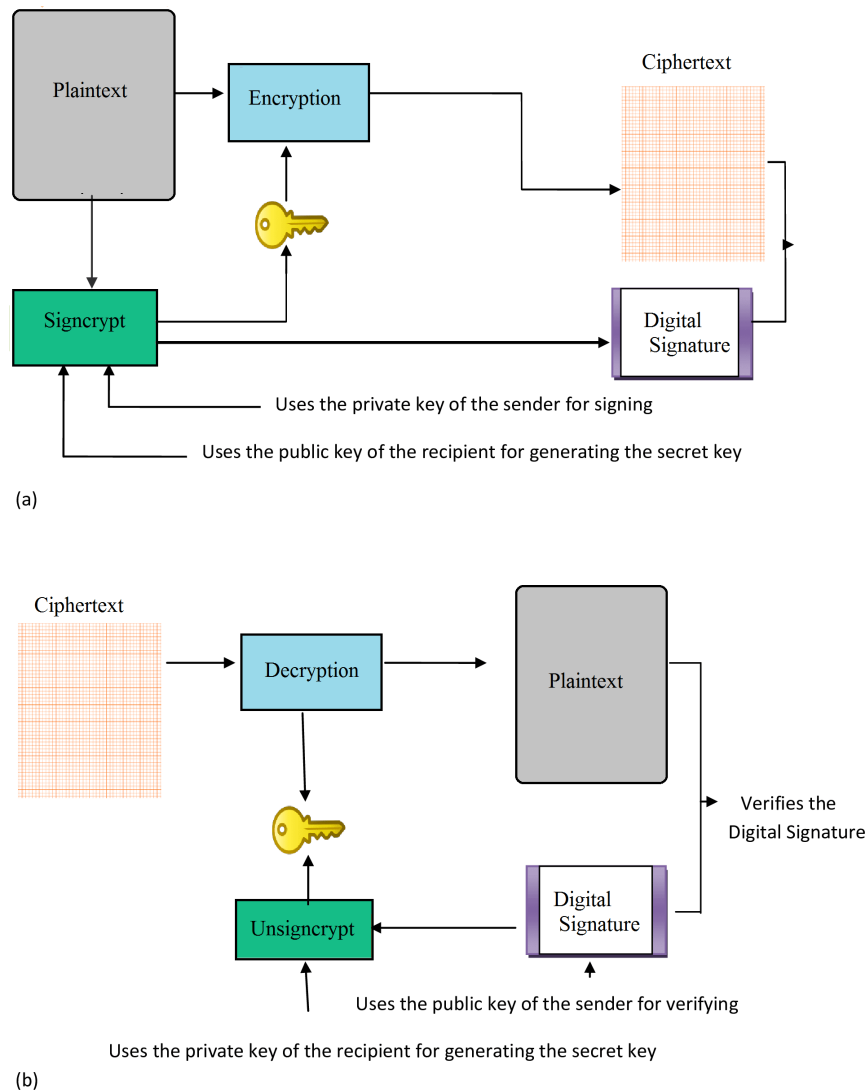


FIGURE 3.9: Signcryption Model

Confidentiality

For an attacker, It should be infeasible to get any information about the data without the knowledge of the sender's or receiver's secret key.

Authentication

It is a process of providing the proof of identity of the sender to the receiver so that the recipient could assure that the message is sent by the authentic person.

Integrity

The recipient should be able to prove that the received data is the same as it was generated by the sender.

Unforgeability

For an attacker, it should be computationally infeasible to create a fake digital signature in such a way that it can be verified by the unsignryption algorithm.

Non-repudiation

The recipient must have the facility to confirm to the judge that the signcrypted text is generated by the authentic sender.

Public Verifiability

Without the use of private key of sender, the recipient or any third party can have the facility of checking the validity of signcrypted text.

Forward Secrecy

If sender's secret key is disclosed, the adversary will not be able to generate any previous data from the ciphertext.

3.5.2 Zheng's Elliptic Curve based Signcryption Scheme

In 1998, Zheng and Imai [21] introduced a first signcryption scheme that uses elliptic curves over a finite field. In their scheme, private key is chosen randomly and public key is generated from elliptic curve point multiplication. For digital signature, they used two different standards called shortened elliptic curve digital signature standard 1 (SECDSS1) and shortened elliptic curve digital signature standard 2 (SECDSS2). They proposed to use the symmetric encryption and decryption. Their scheme reduces the communication and computational cost in comparison of the other public key systems like Elgamal [9] and RSA [10]. The proposed scheme is illustrated in the following phases:

Global Parameters

Both Bob and Alice agree on the following parameters.

Variables	Description
q, n	Large prime numbers greater than 2^{128}
h	Keyed one way hash function
H	One way hash function
G	A generator, which generates a group, of order n
E	Elliptic curve over finite field \mathbb{F}_p
E_k and D_k	Symmetric encryption/decryption algorithm with secret key k

Key Generation

Both Alice and Bob selects and generates their private and public keys as follows.

Alice

- Randomly chooses $r_a < n$ as her private key .
- Computes public key $P_a = r_a G$ as a point on the elliptic curve.

Bob

- Randomly chooses $r_b < n$ as his private key .
- Computes public key $P_b = r_b G$ as a point on the elliptic curve.

Signcryption

Suppose Alice wants to transmits a message M to Bob through unsecured public network. For this purpose, Alice performed the following steps to generate the signcrypted text.

1. Selects a random number $\alpha \in \{1, 2, 3, \dots, n - 1\}$
2. Computes the hash value $H(\alpha P_b) = (k_1, k_2)$

3. Obtains the ciphertext message $C = E_{k_1}(M)$ by using the symmetric encryption.
4. Using the one way hash function, computes the hash value $x = h_{k_2}(M, bind-inf)$
5. Computes the signature parameter $s = \frac{\alpha}{x+r_a}$, if Shortened Elliptic Curve Digital Signature Standard 1 (SECDSS1) is used.

OR

Computes the signature parameter $s = \frac{\alpha}{1+xr_a}$, if Shortened Elliptic Curve Digital Signature Standard 2 (SECDSS2) is used.

6. Send (C, x, s) to receiver.

Unsigncryption

The unsigncryption process is described in following steps.

1. Computes $k = sr_b \mod n$.
2. If Shortened Elliptic Curve Digital Signature Standard 1 (SECDSS1) is used then computes the shared secret key as $H(kP_a + kxG) = (k_1, k_2)$

OR

If Shortened Elliptic Curve Digital Signature Standard 2 (SECDSS2) is used then computes the shared secret key as $H(kG + kxP_a) = (k_1, k_2)$

3. Obtain the original plaintext message $M = D_{k_1}(C)$ by using the symmetric decryption.
4. If $h_{k_2}(M, bind-inf) = x$ then it assures the authenticity of M .

Verification

The both two versions SECDSS1 and SECDSS2 of proposed signcryption scheme [21] are correctly verifiable.

Standard-1 (SECDSS1)

If SECDSS1 is used then sender and receiver of a message generate the same secret key (k_1, k_2) and then use it to generate and verify the digital signature.

$$\begin{aligned}
 (k_1, k_2) &= H(kP_a + kxG) \\
 &= H(sr_b r_a G + sr_b xG) \\
 &= H(sr_b G(r_a + x)) \\
 &= H(r_b G(\frac{\alpha}{r_a + x})(r_a + x)) \\
 &= H(\alpha r_b G) \\
 &= H(\alpha P_b)
 \end{aligned}$$

Version-2 (SECDSS2)

If SECDSS2 is used then sender and receiver of a message generate the same secret key (k_1, k_2) and then use it to generate and verify the digital signature.

$$\begin{aligned}
 (k_1, k_2) &= H(kG + kxP_a) \\
 &= H(sr_b G + sr_b x r_a G) \\
 &= H(r_b s G(1 + x r_a)) \\
 &= H(r_b G(\frac{\alpha}{1 + x r_a})(1 + x r_a)) \\
 &= H(r_b \alpha G) \\
 &= H(\alpha P_b)
 \end{aligned}$$

Zheng's [21] analysis shows that the signcryption scheme eliminates the communication and computational cost in comparison of the signature-then-encryption

scheme. By using the computations of elliptic curve, signcryption reduces 58% computational and 40 % communication cost when it is compared with traditionally used signature-then-encryption scheme. Zheng's signcryption provides the security attributes of non-repudiation, integrity, unforgeability and confidentiality.

3.6 Cryptanalysis

Cryptanalysis is the branch of cryptology that deals with the security analysis of cryptographic schemes. It involves the deep understanding of schemes and then finding the security weakness in the cryptosystem. Cryptanalyst exploits the security weakness and try to find the meaning of encrypted information with or without any secret information. Several cryptographic attacks are identified in the history and these are classified as the passive attack or active attack. In passive attack, cryptanalyst only observes the network communication and try to break the confidentiality of the data. Whereas in active attack the cryptanalyst try to break the confidentiality as well as try to delete, modify and replace the original data with his desired data. Different forms of active and passive attacks are exist in literature depends upon the kind of information a cryptanalysis has. These attacks are discussed in detail as described below.

3.6.1 Chosen Plaintext Attack

This type of attack is applicable when an attacker chooses any message of his choice and gets the ciphertext of it. An attacker analyze the relationship between plaintext and its corresponding ciphertext to guess the secret key. This type of attack is powerful as the attacker input any message of his choice to guess the key from ciphertext. The attack model is described in Figure [3.10](#).

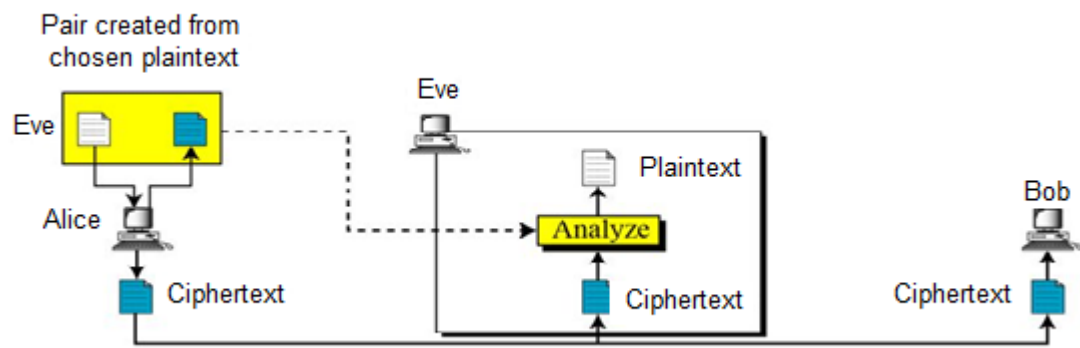


FIGURE 3.10: Chosen Plaintext Attack Model

3.6.2 Known Plaintext Attack

It is applicable when an adversary has original plaintext messages and their corresponding ciphertexts. An attacker wants to get the secret key or tries to make an algorithm that decrypt further messages.

3.6.3 Chosen Ciphertext Attack

In chosen ciphertext attack, an adversary gets the ciphertext messages as well as their corresponding plaintext messages. The basic aim of attacker is to recover the secret key or gets more information about the cryptosystem. Mostly this type of attack is feasible when an adversary has the limited access to the decryption machine. Mostly this attack model is implemented in public key cryptography. The attack model is described in Figure 3.11.

3.6.4 Known Ciphertext Attack

In this attack model, an attacker gets ciphertext message from publicly available information and tries to generate its original plaintext message or the secret key. An attacker gets all the plaintext messages form ciphertext, if the private key is

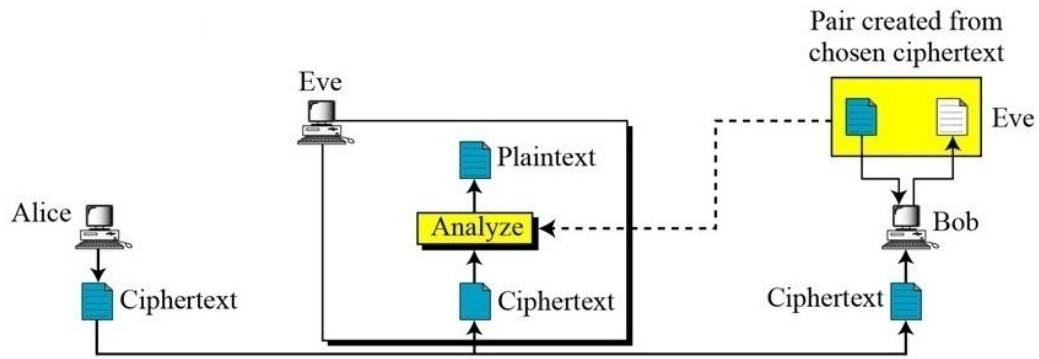


FIGURE 3.11: Chosen Ciphertext Attack Model

compromised. This attack model is commonly used in cryptography. The attack model is described in Figure 3.12.

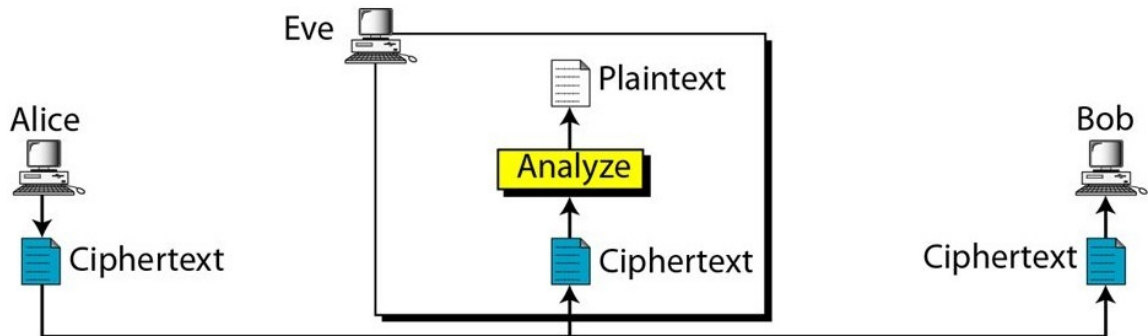


FIGURE 3.12: Ciphertext Only Attack Model

3.6.5 Forgery Attack

In this attack model, an adversary intercepts the network communication between the sender and the receiver. The aim of the attacker is to modify or replace the original message with his desired message in such a way that unsigncryption algorithm correctly verifies it. For this purpose, attacker generates a fake digital signature with the help of public parameters on his desired message in such a way that signature verification process correctly verifies the signature. After the verification of fake digital signature, receiver believes that the received message is not tampered during the transmission and sent by authentic person. In this way,

attacker transmits any message of his choice without the knowledge of sender and receiver.

3.6.6 Man-In-The-Middle Attack

In this attack model, an adversary indulges himself in between the communication of the sender and the receiver. The aim of the attacker is to establish a separate connections with each of the participants. In such connections, the sender and the receiver believes that they are communicating with each other but actually they are communicating with the attacker. In this way, the attacker generates separate keys for the sender and the receiver. Attacker uses these shared common keys to transmits any message of his choice. For protection against this type of attack, a strong authentication protocol is used in communication. The attack model is described in Figure 3.13.

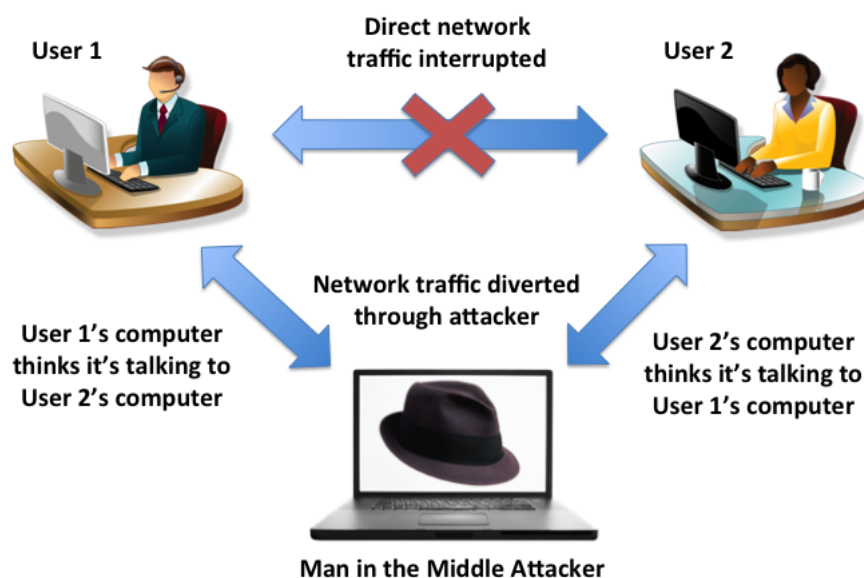


FIGURE 3.13: Man-In-The-Middle Attack

3.6.7 Man-At-The-End Attack

MATE attack is difficult to analyze, model and evaluate because the attacker has authorized and limitless access to the device. All the protections of the compromised device are stand up for specific period of time. The different forms of MATE attack are tempering attack, cloning attack, reverse engineering attack and exploiting the personal codes [104]. The attack model is described in Figure 3.14.

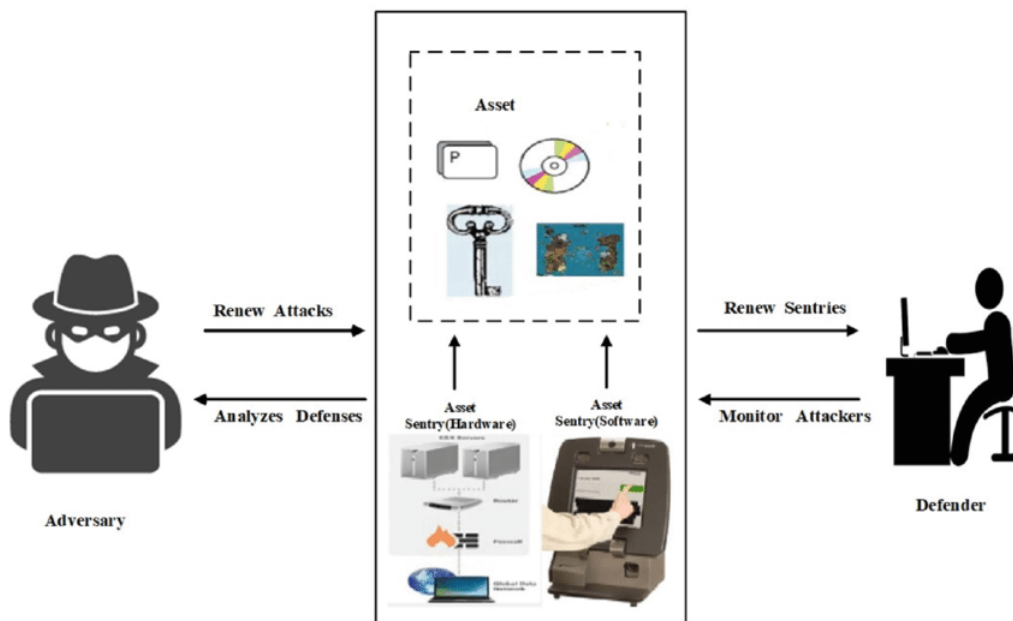


FIGURE 3.14: Man-At-The-End Attack Model [104]

For further details on these attacks we refer [104], [92].

3.6.8 Brute Force Attack

In this attack model, an attacker uses trial and error method to get the secret key or password. An adversary tries all the possible keys or passwords and check which ciphertext is correct. For this attack, specially designed computers are used to break the cryptosystem. The time required to break a cryptosystem depends upon the size of key used in encryption process. The attack model is described in Figure 3.15.



FIGURE 3.15: Brute Force Attack [104]

Chapter 4

Cryptanalysis and Improvement of an Elliptic Curve based Signcryption Scheme for Firewalls

In network security, firewall is a security mechanism that observes and controls the network traffic based on some predefined rules. A firewall sets up a barrier between internal network and another outside unsecured network, such as the internet. It provides an additional security layer for any signcryption scheme. This chapter focuses on the cryptanalysis of the elliptic curve based signcryption scheme for firwalls proposed by Iqbal et al. [86]. The analysis of the scheme shows that it is is not secure and has many security flaws. Anyone who knows the public parameters can modify the message without the knowledge of the sender and the receiver. Due to our successssful cryptanalysis, the claimed security attributes of non-repudiation, unforgeability, integrity and authentication are compromised. First, the scheme of Iqbal et al. [86] is described in Section 4.1 and its cryptanalysis is presented in Section 4.2. Later on, a modified version of the scheme is presented and its security against known attacks is also investigated.

4.1 Signcryption scheme of Iqbal et al. [86]

The basic aim of the proposed scheme of Iqbal et al. [86] is to present a new signcryption scheme for firewalls. The proposed scheme depends upon the elliptic curve for generation and verification of the digital signature. Their analysis shows that the proposed scheme is secure. The claimed security attributes of proposed scheme are integrity, message confidentiality, signature unforgeability, public verifiability, non-repudiation, and forward secrecy property. They also gave the cost analysis of proposed scheme and proves that proposed scheme is computationally efficient as compared to existing signcryption schemes. The scheme proposed by Iqbal et al. [86] is described below.

Global parameters

Both Bob and Alice agreed on the following global parameters as given in Table 4.1 [86].

Variables	Description
p^*	A large prime number greater than 2^{1024} .
$E_{p^*}(a, b)$	Elliptic curve over $GF(p^*)$.
G	A base point G of a group of a very large order q .
h	A one way hash function.
E and D	Symmetric encryption and decryption algorithms .
ID_i	Identifiers of sender and receiver from CA.

TABLE 4.1: Global Parameters of the Scheme [86]

Key Generation Phase

- Alice (Sender)
 - Selects an integer n_A randomly as a private key such that $n_A < q$.
 - Computes her public key $P_A = n_A G$ as a elliptic curve point.
- Bob (Receiver)
 - Selects an integer n_B randomly as a private key such that $n_B < q$.
 - Computes his Public key $P_B = n_B G$ as elliptic curve point.

Signcryption Phase

Let Alice (Sender) wants to transmit a message m to Bob (Receiver) over a public network. First Alice checks the Bob's certificate and verifies his public key P_B . Then she performs the following steps to generate and send a signcrypted text to Bob.

Alice

1. Chooses a random number $v \in \{1, 2, 3, \dots, q-1\}$.
2. Computes $R = vG = (x_R, y_R)$.
3. Computes $r = (v + n_A) \bmod q$.
4. Computes $Q = rP_B = (x_Q, y_Q)$.
5. Computes $k = h(x_Q || ID_A || y_Q || ID_B)$.
6. Computes ciphertext $C = E_k(m)$ by using symmetric encryption E_k with the secret key k .
7. Computes $t = h(C || x_R || ID_A || y_R || ID_B)$.
8. Computes $s = rt^{-1} \bmod q$.
9. Sends (C, R, s) to Bob.

Firewalls Signature Verification Phase

The proposed scheme enables firewalls to authenticate the signcrypted text (C, R, s) without reading the contents of the original message. Only the ciphertext and public parameters are required to verify the signature unforgeability. Firewalls authentication consists of the following steps:

1. Receives (C, R, s) from the Alice.
2. Computes the elliptic curve point $P^* = (R + P_A)$.
3. Computes $t = h(C || x_R || ID_A || y_R || ID_B)$ by using the public parameter.

4. Firewalls accept the message m only if $stG = P^*$.

Unsigncryption Phase

Bob

1. Recieves (C, R, s) from Alice.
2. Computes the elleptic curve point $P^* = (R + P_A)$.
3. Computes $Q = (n_B)P^* = (x_Q, y_Q)$.
4. Computes $k = h(x_Q || ID_A || y_Q || ID_B)$.
5. Gets plaintext message $m = D_k(C)$ by using symmetric encryption scheme with shared key k .
6. Computes $t = h(C || x_R || ID_A || y_R || ID_B)$.
7. Accept the message m only if $stG = P^*$.

4.2 Cryptanalysis

In this section, the security of Iqbal et al. scheme [86] is analyzed. The cryptanalysis of the proposed scheme shows that it has many security issues and weaknesses. The security attributes of message authenticity, unforgeability and non-repudiation are compromised. Mallory (an Attacker) generates the signcrypted text from his desired message in such a way that unsigncryption algorithm correctly verifies it. Suppose Mallory intercepts the network traffic between Alice and Bob and wants to establish the trustful connections with them. T The cryptanalysis model for the proposed scheme is described in Figure 4.1. he Main purpose of Mallory is to send his desired message M to Bob. Following steps are performed to generate and sends a signcrypted text of his choice.

Signcryption Phase

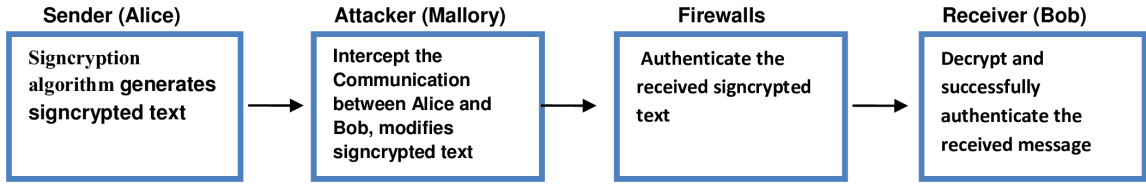


FIGURE 4.1: Cryptanalysis Model

Mallory performs the following operations to transmit a message m' of his choice.

(a) Mallory

1. Chooses a random number $v \in \{1, 2, 3, \dots, q-1\}$.
2. Computes the elliptic curve point $R' = v'G - P_A = (x'_R, y'_R)$.
3. Computes the elliptic curve point $Q' = v'P_B = (x'_Q, y'_Q)$.
4. Computes the secret key as $k' = h(x'_Q || ID_A || y'_Q || ID_B)$.
5. Computes the ciphertext $C' = E_{k'}'(m')$ by using symmetric encryption s with secret key k' .
6. Computes $t' = h(C' || x'_R || ID_A || y'_R || ID_B)$ by using hash function.
7. Computes the signature parameter $s' = t'^{-1}v' \mod q$.
8. Sends (C', R', s') to Bob.

(b) Firewalls Signature Verification Phase

1. Receives the signcrypted text (C', R', s') from sender.
2. Computes the elliptic curve point $P^* = (R' + P_A)$.
3. Computes $t' = h(C' || x'_R || ID_A || y'_R || ID_B)$ by using the public parameters.
4. Firewalls authenticate the message m' by verifying the equation $s't'G = P^*$.

(c) Unsigncryption Phase

Bob

1. Receives the signcrypted text (C', R', s') .
2. Computes the elliptic curve point $P^* = (R' + P_A)$
3. Computes the elliptic curve point as $Q' = (n_B)P^* = (x'_Q, y'_Q)$
4. Computes the secret key $k' = h(x'_Q || ID_A || y'_Q || ID_B)$.
5. Gets the plaintext message $m' = D_{k'}(C')$ by using symmetric encryption with secret key k' .
6. Computes the hash value $t' = h(C' || x'_R || ID_A || y'_R || ID_B)$.
7. Accept the message m' by verifying $s't'G = P^*$.

In this way, Mallory makes a fake signcrypted text of his choice and sends it to Bob. After receiving the signcrypted message (C', R', s') , first the firewalls successfully verifies the signature. On the receiver's end, unsigncryption, algorithm verifies the signcrypted text and then decrypts the message. Bob now believes that the message is sending by authentic person Alice. In this way, Mallory defeats the cryptosystem by sending the signcrypted text of his choice. Further, recall the Man-At-The-End (MATE) attack as described in Section 3.6.7 and note that this scheme has no protection against MATE attack.

4.2.1 Proof of Correctness

This section shows that the above presented cryptanalysis is correct. That is, the same secret key k' is generated by Mallory and Bob. The elliptic curve point Q' , which is used for generation of secret key k' , is same. For instance,

$$\begin{aligned}
 Q' &= (n_B)P^* && \text{(Receiver's end)} \\
 &= (n_B)(R' + P_A) && \text{(Step 2 of Section 4.2(c))} \\
 &= (n_B)(v'G - P_A + P_A) && \text{(Step 2 of Section 4.2(a))} \\
 &= (n_B)(v'G) && \text{(Additive Inverse)} \\
 &= v'P_B && \text{(Key Generation Phase)} \\
 &= Q' && \text{(Sender's end)}
 \end{aligned}$$

After receiving the signcrypted text, unsignryption algorithm correctly verifies the authenticity of the received message.

$$\begin{aligned} s't'G &= (t'^{-1}v')(t'G) \\ &= v'G \\ &= P^* \end{aligned}$$

After this verification, Bob believes that the received message is sending by authentic person Alice. In this way, Mallory defeats the cryptosystem and now able to send his desire message to Bob.

4.3 Modified Signcryption Scheme

Our analysis shows that the claimed security attributes of proposed signcryption scheme of Iqbal et al, [86] are compromised. To overcome this issue, we modifies the existing scheme to ensure the basic properties of security. In proposed scheme, the method to generate common secret key is very weak. In improved scheme, a key generation process is modified in such a way that only authentic sender and receiver can generate a valid common key. In Step (5) of Signcryption Phase (4.1), replace (ID_A, ID_B) to (x_S, y_S) in key generation phase. In improved scheme, only authentic sender can generate the signcrypted text that is verified by unsignryption algorithm. The private and public key generation process is same as described in Key Generation Phase (4.1).

Global parameters Both Bob and Alice agreed on the same parameters as listed in Table 4.1.

First Alice checks the Bob's certificate and verifies his public key P_B . Then she performed following steps to generate the signcrypted text.

Signcryption Phase

Alice

1. Choose a random number $v \in \{1, 2, 3, \dots, q-1\}$.

2. Computes $R = vG = (x_R, y_R)$.
3. Computes $r = (v + n_A) \bmod q$.
4. Computes $Q = rP_B = (x_Q, y_Q)$.
5. Computes $S = (n_A)P_B = (x_S, y_S)$.
6. Computes $k = h(x_Q || x_S || y_Q || y_S)$.
7. Gets the ciphertext $C = E_k(m)$ by using symmetric encryption E_k with secret key k .
8. Computes $t = h(C || x_R || ID_A || y_R || ID_B)$.
9. Computes the signature parameter $s = t^{-1}r \bmod q$.
10. Sends (C, R, s) to Bob.

Firewalls Signature Verification Phase

The improved scheme enables firewalls to authenticate the signcrypted text (C, R, s) without reading the contents of the original message. Firewalls authentication consists of the following steps:

1. Receives (C, R, s) from the Alice.
2. Computes the elliptic curve point $P^* = (R + P_A)$.
3. Computes $t = h(C || x_R || ID_A || y_R || ID_B)$ by using the public parameter.
4. Firewalls accept the message m only if $stG = P^*$.

Unigncrypton Phase

Bob

1. Recieves (C, R, s) from sender.
2. Computes the elleptic curve point as $P^* = (R + P_A)$.

3. Computes $Q = (n_B)P^* = (x_Q, y_Q)$.
4. Computes $S = (n_B)P_A = (x_S, y_S)$.
5. Computes the common shared secret key $k = h(x_Q||x_S||y_Q||y_S)$.
6. Gets the plaintext message $m = E_k(C)$ by using symmetric encryption with secret key k .
7. Computes the signature parameter $t = h(C||x_R||ID_A||y_R||ID_B)$.
8. Accept the message m only if $stG = P^*$.

4.3.1 Proof of Correctness

The modified scheme is correctly verifiable. The same secret key k is generated by both the sender and the receiver. The elliptic curve point Q is used for key generation, which is same in Step (4) of Signcryption Algorithm and Step(3) in Unsigncryption Algorithm.

On the Receiver's end, Bob computes $Q = n_BP^*$ in Step 3 of Unsigncryption Phase. But $P^* = (R + P_A)$ from Step 2 of Unsigncryption Phase. So,

$$Q = n_B(R + P_A)$$

Replace R by vG in Step 2 of Signcryption Phase and P_A by n_AG in key generation phase. So,

$$Q = n_B(vG + n_AG)$$

and the fact that $n_BG = P_B$ from Key Generation Phase and $r = v + n_A$ in Step 3 of Signcryption Phase, the above equation becomes

$$Q = rP_B$$

So the same secret key is generated on both ends.

Further Receiver accept the message m only if the signature is verified by unsign-
 crypton algorithm. In the Verification Phase, Bob computes stG in Step 8 of
 Unsigncrypton Phase. As $s = t^{-1}r$ from Step 9 of Signcrypton Phase, so

$$stG = (t^{-1}r)tG = rG$$

But $r = v + n_A$ in Step 3 of Signcrypton Phase, so

$$stG = (v + n_A)G$$

As $R = vG$ from Step 2 of Signcrypton phase and $P_A = n_A G$ from Key Generation
 Phase, finally get

$$stG = R + P_A = P^*$$

In fact, the verification of above equation provides the authenticity of the received
 message. After the above verification of the signature, Bob accepts the received
 message as valid and authentic.

4.4 Analysis of Modified Scheme

The analysis of the modified scheme is presented in this section. The computa-
 tional cost in signcrypton, unsigncrypton and signature verification phase is same
 as given in [86]. The communication cost of modified scheme is also same as in
 [86]. The security analysis of the modified scheme is described below.

4.4.1 Security Attributes

The security of the improved scheme depends upon ECDLP. The improved scheme
 is secure and provides the security attributes of confidentiality, signature unforge-
 ability, integrity, forward secrecy of message confidentiality, authentication, public
 verification and non-repudiation.

Confidentiality

The security of our improved scheme depends upon elliptic curve discrete logarithm problem (ECDLP), which is computationally infeasible to solve. An adversary will not be able to read the contents of the original message without the secret parameters r, v and n_A . The common shared secret key k is used by both the sender and the receiver for symmetric encryption and decryption. If an adversary wants to compute the secret key $k = h(x_Q || x_S || y_Q || y_S)$ in Step 6 of Signcryption Phase then he has to find

$$\begin{aligned} Q &= rP_B = (x_Q, y_Q) \\ S &= n_AP_B = (x_S, y_S) \end{aligned}$$

in Step 4 and Step 5 of Signcryption Phase. If an attacker wants to find $r = v + n_A$ then he must have the knowledge of both secret parameters r and n_A in Step 3 of Signcryption Algorithm. To find n_A , given $P_A = n_AG$ and G in the key generation process means to solve ECDLP.

Integrity

The improved scheme provides integrity of the data. After receiving the signcrypted text, receiver will verify that the received message M is not tampered in the process of transmission. If an attacker will change the ciphertext C to C' then consequently

$$t = h(C || x_R || ID_A || y_R || ID_B)$$

changes to t' in Step 8 of Signcryption Algorithm. Due to these changes, signature generated in Step 9 of Signcryption algorithm changes from s to s' . Thus, Unsigncryption Algorithm can not verify the signature on the message M and hence rejected the modified message.

Non-repudiation

When dispute occurs between two parties then the receiver of a message will send (C, R, s) to judge or third party for checking the authenticity of the sender and the message M . The judge will be able to verify the authenticity of original message M by using the signature

$$s = t^{-1}r \mod q$$

in Step 9 of Signcryption Algorithm. The secret parameter $r = v + n_A$ in Step 3 of Signcryption Algorithm involves secret key n_A of the sender (Alice). It confirms that the Alice is the original sender of the message she will not be able to deny being the sender of the message.

Forward Secrecy

In the modified scheme, if sender's long term private key is disclosed then the adversary will not be able to get the contents of original message. The secret random number v is involved in Step 1 of Signcryption Phase and is changed each time the message is encrypted. It provides forward secrecy of message confidentiality in the modified scheme.

Public Verification

The improved scheme provides public verifiability property. Anyone with the help of public parameters can be able to verify that signcrypted text is generated by the authentic person. The verifier can get s , G and R from public information and generate elliptic curve point $P^* = (R + P_A)$ by using the public key P_A of the Alice. The verifier of a message will check the equation $stG = P^*$ to prove the authenticity of message.

Unforgeability

The improved scheme also provides unforgeability. An attacker can not forge a valid signature that is generated by authentic sender. If an attacker wants to generate a valid signature s in Step 9 of signcryption process then he must has the secret parameters t and r . The generation of secret parameter

$$r = v + n_A \mod q$$

in Step 3 of Signcryption Algorithm involves secret key n_A of the sender that is not possible to find or generate from a very large key size in a elliptic curve group $E_p(a, b)$. So an attacker cannot generate a valid signature s on message m .

Authentication

The scheme ensures authentication, as it is certificate based. The validity of certificates is verified in signcryption and unsigncryption phases.

The comparison of security attributes of modified scheme with the existing schemes is described in Table 4.2 below.

4.4.2 Attack Analysis

As discussed earlier, the proposed signcryption scheme of iqbal et al. [86] is vulnerable to Man-in-middle attack and Man-at-the-end attack. The improved scheme is secure and provides protection against these attacks. We now discuss the impact of these attacks in our improved signcryption scheme and then discuss the counter measures against these attacks.

Signcryption Scheme	C	I	U	N	P	A	F	F.S
Zheng [20]	yes	yes	yes	yes	no	no	no	no
Gamage et al. [25]	yes	yes	yes	yes	yes	yes	no	yes
Bao and deng [22]	yes	yes	yes	yes	no	no	no	no
Jung et al. [24]	yes	yes	yes	yes	no	no	yes	no
Elkamchochi [105]	yes	yes	yes	yes	no	no	no	no
Zheng and Imai [21]	yes	yes	yes	yes	no	no	no	no
Mohamed [49]	yes	yes	yes	yes	yes	yes	no	yes
Hwang et al. [106]	yes	yes	yes	yes	no	yes	no	no
Zhou [47]	yes	yes	yes	yes	no	yes	no	no
Han et al. [50]	yes	yes	yes	yes	no	yes	no	no
Iqbal et al [86]	yes	no	no	no	yes	no	yes	yes
Our Modified Scheme	yes	yes	yes	yes	yes	yes	yes	yes

C: Confidentiality, I: Integrity, U: Unforgeability, N: Non-repudiation, P: Public Verification, A: Authentication of ciphertext-only, F: Forward Secrecy, F.S: Firewall Suitability.

TABLE 4.2: Comparison of Modified Scheme with Existing Schemes

Man-At-The-End (MATE) Attack

Previously Man-At-The-End (MATE) attack is neglected largely in security analysis by researchers because it is difficult to model, analyze and evaluate predominantly [104]. Since the attacker is human, therefore can utilize all the capabilities of a human mind. Beside the adversary has authorized and unlimited access to the device and this results in all security protections to stand up for an adversary for a specific period of time.

The MATE attack has different forms depending upon the physical scenario of compromised device. At an individual level, altering attack is possible in which adversary altered the integrity of piece of software [107]. In reverse engineering attack, the adversary trace the intellectual property rights from the device software and then disrupts the privacy right of vendor [108]. Similarly, in cloning attack an adversary creates and issues the copies of software by vilating the copyright laws [109]. Sometime an adversary may attack by crafting his own exploit code using the publicly available codes to make it hard to be reconised by an antivirus software [110].

Although MATE attack is difficult to analyze and model but there are mechanism to protect your device. The techniques to protect against MATE attack are: digital asset protection, software protection, hardware protection and hardware based software protection. For further details on core protection mechanism against this attack we refer to [104].

Man-In-The-Middle Attack

In man-in-middle attack , an adversary intercepts the network traffic between two parties and alter the information in such a way that both parties believes they are communicating with each other. The proposed Signcryption scheme of Iqbal et al. [86] is not secure against man-in-middle attack and an active attacker modifies the signcrypted text that is verified by unsignryption algorithm.

Our modified signcryption scheme overcome this security issue and resist against the man-in-middle attack. An Adversary gets the signcrypted text (C, R, s) from publicly transmitted information but unable to modify the signcrypted text of his choice that is verified by Unsignryption Algorithm. In the modified scheme, the private key of Alice is used for key generation process in Step (5) of Signcryption Algorithm and then used for signature generation in Step (9) of Signcryption Algorithm. If an attacker generates a signcrypted text with any fake key then Unsignryption Algorithm will not verify the signature s in Step (8) of Unsignryption Algorithm and hence the message M will not be accepted.

4.5 Conclusion

In this chapter, the security strenght of Iqbal et al. [86] scheme is analyzed and it is proved that it has many security flaws. In their proposed scheme, one can easily generate the signcrypted text of his choice that is acceptable by Unsignryption Algorithm. Their scheme does not provide message authentication, integrity, non-repudiation and unforgeability as claimed in [86]. The modified scheme ensure

the compromised security attributes of the proposed scheme. As discussed earlier, the improved scheme has the security attributes of authentication, unforgeability, integrity, message confidentiality, non-repudiation, public verification, authentication of ciphertext-only, forward secrecy and firewall suitability. The comparison of the modified signcryption scheme with the existing schemes in the literature is highlighted in Table 4.2. The content presented in this chapter has been published in journal Plos One [88].

Chapter 5

Cryptanalysis and Improvement of Blind Signcryption Scheme based on Elliptic Curve

Blind signcryption schemes are the extension of signcryption schemes. They are used to protect the privacy and identity of the sender from other users, especially in electronic voting and electronic cash payment systems. In [87], Riazullah et al. proposed a blind signcryption scheme based on elliptic curves. The claimed security attributes of their proposed scheme are confidentiality, sender anonymity, message integrity, authentication, unforgeability, signer non-repudiation, forward secrecy, blindness and message untraceability. In this chapter, the cryptanalysis of their proposed blind signcryption scheme [87] is carried out. It is observed that the scheme has many security flaws and fails to provide some of the claimed security attributes. After identifying the security flaws, a modified version of the proposed scheme is introduced. The blind signcryption scheme of Riazullah et al. [87] is described in Section 5.1. The cryptanalysis together with its correction is presented in Section 5.2. The modified version of this scheme is introduced in Section 5.3 and Section 5.4 is devoted for the security analysis of the modified scheme.

5.1 Elliptic Curve based Blind Signcryption Scheme

[87]

In this section, the blind signcryption scheme of Riazullah et al. [87] is described. The scheme is based on elliptic curve cryptography and its security relies on the difficulty of solving ECDLP (Definition 2.4.2). The proposed scheme has three participants:

1. Sender or Requester (Alice) wants to communicate anonymously with receiver.
2. Receiver or Verifier (Bob) authenticate the received message.
3. Signer is the party that signs the received message without reading the content of the original message.

The blind signcryption scheme of Riazullah et al. [87] consists of four phases:

- (a) Pre-request Phase (Global Parameter)
- (b) Key Generation Phase
- (c) Blind Signcryption Phase
- (d) Unsigncryption Phase

The detailed description of these phases is stated below.

(A) Pre-request Phase

In this phase, system publishes the global parameters as given in Table 5.1.

(B) Key Generation Phase

Signer

- Chooses an integer x_s randomly as his secret key such that $x_s < n$.

Variables	Description
q	A prime number greater than 2^{160} .
F_q	A working finite field.
E	Elliptic curve over finite field F_q .
n	A prime number with $n > 2^{160}$.
h	A one-way hash function.
G	A base point of E such that $nG = 0$.
E_k	Symmetric encryption algorithm with private key k .
D_k	Symmetric decryption algorithm with private key k .

TABLE 5.1: Global Parameters of the Scheme [87]

- Computes his public key $Y_s = x_s G$. as elliptic curve point.

Alice

- Chooses an integer x_r randomly as his secret key such that $x_r < n$.
- Computes her public key $Y_r = x_r G$. as elliptic curve point.

Bob

- Chooses an integer x_v randomly as his secret key such that $x_b < n$.
- Computes his public key $Y_b = x_b G$ as elliptic curve point.

(C) Blind Signcryption Phase

Suppose Alice (sender) wants to transmits a message m over a public network to Bob. First Alice blinds the message m and then sends it to signer of a message for signing. After receiving the signed message from signer, Alice unblind the document and then sends this signcrypted text to Bob (Receiver). Following steps are required to generate the blind signcrypted text.

Signer

1. Chooses a random number $a \in \{1, 2, 3, \dots, n-1\}$.
2. Computes $Z = aG \pmod n$.
3. Sends Z to Alice.

Alice

4. Selects random numbers $\alpha, \beta, \gamma \in \{1, 2, 3 \dots, n-1\}$ as a blinding factors.
5. Computes the secret key $(k_1 || k_2) = h(\gamma Y_b \bmod n)$
6. Computes $t = kh_{k_2}(m || k_2)$
7. Computes the ciphertext $c = E_{k_1}(m)$ by using symmetric encryption with secret key k_1 .
8. Computes $X = ((\gamma + \beta)Z + \alpha G) \bmod n$
9. Computes $\bar{t} = (t + \beta) \bmod n$
10. Sends \bar{t} to signer.

Signer

11. Computes $\bar{\ell} = (x_s + \bar{t}a) \bmod n$
12. Sends $\bar{\ell}$ to Alice.

Alice

13. Computes the signature parameter $s = \frac{\gamma}{t + \bar{\ell} + \alpha} \bmod n$ by using random numbers α and γ
14. Sends (c, t, s, X) to Bob.

(D) Unsigncryption Phase

Bob

1. Receives (c, t, s, X) from the Alice
2. Computes $v = x_b s$.
3. Computes the secret key $(k_1 || k_2) = h(v(Y_s + X + tG))$
4. Gets the plaintext message $m = D_{k_1}(c)$ by using symmetric encryption with secret key k_1
5. Computes $r_1 = kh_{k_2}(m || k_2)$
6. If $r_1 = t$ then consider m as a valid message otherwise refuse.

5.2 Cryptanalysis

In this section, the cryptanalysis of the scheme presented above in Section 5.1 is proposed and proved it to be insecure. The analysis showed that the claimed security properties of authentication, message integrity, signer non-repudiation and unforgeability are compromised. Suppose Mallory(Attacker) wishes to transmit a message m' of his choice to Bob. Mallory intercepts the network communication between Alice and Bob and generates a signcrypted text of his choice. Bob received the signcrypted text from Mallory and accepts the received message m' as a valid message. Mallory performs the following procedure to send a fake message m' .

(A) Signcryption Phase

Mallory

1. Chooses a random number $a' \in \{1, 2, 3, \dots, n-1\}$.
2. Computes $Z' = a'G \pmod n$ by using the base point G .
3. Computes random numbers $\alpha', \beta', \gamma' \in \{1, 2, 3, \dots, n-1\}$.
4. Computes $(k'_1 || k'_2) = h(\gamma'Y_b \pmod n)$
5. Computes $t_1 = kh_{k'_2}(m' || k'_2)$
6. Gets the ciphertext message $c' = E_{k'_1}(m')$ by using symmetric encryption with secret key k'_1
7. Computes $X'_1 = ((\gamma' + \beta')Z' + \alpha'G) \pmod n$
8. Computes $t_2 = (\gamma' + \beta') \pmod n$
9. Computes the signature parameter $s' = \frac{\gamma'}{t_1 + t_2a' + \alpha'}$
10. Computes $X_1 = X'_1 - Y_s$
11. Sends (c, t_1, s', X_1) to Bob.

(B) Unsigncryption Phase

Bob

1. Receives the signcrypted text (c, t_1, s', X_1)

2. Computes $v' = x_b s'$. by using his private key.
3. Computes $(k'_1 || k'_2) = h(v'(Y_s + X_1 + t_1 G))$
4. Computes $m' = D_{k'_1}(c')$ by using symmetric decryption with secret key k'_1 .
5. Computes $r'_1 = kh_{k'_2}(m' || k'_2)$
6. Accept m' as a valid and authentic message because $r'_1 = t_1$.

The unsigncryption algorithm authenticates the received message and Bob now believes that the received message m' is valid and sent by authentic person Alice. In this way, Mallory successfully transmits the message m' of his choice to Bob. Due to successful implementation of this attack, the claimed security attributes of confidentiality, message integrity, authentication, unforgeability, signer non-repudiation are compromised.

5.2.1 Proof of Correctness

The Unsigncryption Algorithm verifies the authenticity of received signcrypted text correctly. The same secret key k' is generated by the Mallory and the Bob to get the valid authentic signature.

$$\begin{aligned}
 (k'_1 || k'_2) &= h(v'(Y_s + X_1 + t_1 G)) && \text{(Receiver's End)} \\
 &= h(x_b s'(Y_s + X_1 + t_1 G)) && \text{(Step 2 of Unsigncryption Phase)} \\
 &= h(x_b s'(Y_s + X'_1 - Y_s + t_1 G)) && \text{(Step 10 of Signcryption Phase)} \\
 &= h(x_b s'(((\gamma' + \beta')Z' + \alpha'G) + t_1 G)) && \text{(Step 7 of Signcryption Phase)} \\
 &= h(x_b s'(((\gamma' + \beta')a'G + \alpha'G) + t_1 G)) && \text{(Step 2 of Signcryption Phase)} \\
 &= h(x_b s'((t_2 a' + \alpha') + t_1)G) && \text{(Step 8 of Signcryption Phase)} \\
 &= h(x_b \frac{\gamma'}{t_1 + t_2 a' + \alpha'}(t_2 a' + \alpha' + t_1)G) && \text{(Step 9 of Signcryption Phase)} \\
 &= h(\gamma' Y_b) && \text{(Sender's End)}
 \end{aligned}$$

The shared secret key $(k'_1 || k'_2)$ between Mallory and Bob is same. Bob now correctly verifies the authenticity of the received message m' by using the relation $r'_1 = kh_{k'_2}(m' || k'_2) = t_1$.

5.3 Modified Blind Signcryption scheme

Our analysis shows that proposed signcryption scheme of Riazullah et al. [87] is not secure and fails to provide the claimed security properties of signer's non-repudiation, authentication, unforgeability and message integrity. The process of signature generation and verification of proposed scheme is not up to mark according to desired security requirements. We modify the signature generation process of the proposed scheme in such a way that only the signcrypted text generated by authentic sender is verified by Unsigncryption Algorithm. The modified scheme uses private key of signer for generation of signature so that only authentic sender can generate the valid signature. The modified scheme provides the security attributes of sender anonymity, authentication, message integrity, unforgeability, signer non-repudiation, confidentiality, forward secrecy, blindness and message untraceability. The Global Parameters and Key Generation Phase are same as described in Section 5.1. The blind signcryption and unsigncryption phases are described below.

(A) Blind Signcryption Phase

Signer

1. Chooses a random number $a \in \{1, 2, 3, \dots, n-1\}$.
2. Computes $Z = aG \pmod n$.
3. Sends Z to Alice.

Alice

4. Chooses random numbers $q, \alpha, \beta, \gamma \in \{1, 2, 3, \dots, n-1\}$ as a blinding factors.
5. Computes $(k_1 || k_2) = h(\gamma Y_b \pmod n)$
6. Computes $t = kh_{k_2}(m || k_2)$
7. Computes $X = ((t + \beta)Z + \alpha G) \pmod n$
8. Computes ciphertext $c = E_{k_1}(m)$ by using symmetric encryption with secret key k_1

9. Computes $\bar{t} = (t + \beta) \bmod n$

10. Computes $Q_1 = qY_b$

11. Sends (\bar{t}, Q_1) to signer.

Signer

12. Computes $\bar{\ell} = (x_s + \bar{t}a) \bmod n$

13. Computes $Q_2 = x_s Q_1$

14. Sends $(\bar{\ell}, Q_2)$ to Alice.

Alice

15. Computes $Q_3 = q^{-1}Q_2 = (q_1, q_2)$

16. Computes $s = \frac{\gamma q_2}{t + \bar{\ell} + \alpha} \bmod n$

17. Sends (c, t, s, X) to Bob.

(B) Unsigncryption Phase

Bob

1. Receives (c, t, s, X) from the Alice

2. Computes $Q_3 = x_b Y_s = (q_1, q_2)$

3. Computes $v = x_b q_2^{-1} s$.

4. Computes $(k_1 || k_2) = h(v(Y_s + X + tG))$

5. Computes plaintext $m = D_{k_1}(c)$ by using symmetric encryption with secret key k_1

6. Computes $r_1 = kh_{k_2}(m || k_2)$

7. If $r_1 = t$ then accept m as a valid and authentic message otherwise reject.

5.3.1 Proof of Correctness

The modified signcryption scheme is correctly verifiable. The same secret key is generated on the sender and the receiver's end. From Step 4 of Section 5.3(B),

the receiver of the message generates the secret key $(k_1||k_2) = h(v(Y_s + X + tG))$. But $X = ((t + \beta)Z + \alpha G) \mod n$ from Step 7 of Section 5.3(A), so

$$(k_1||k_2) = h(v(Y_s + ((t + \beta)Z + \alpha G) + tG))$$

Replace $v = x_b q_2^{-1} s$ from Step 3 of Section 5.3(B) to get

$$(k_1||k_2) = h(x_b q_2^{-1} s(Y_s + (t + \beta)aG + \alpha G + tG))$$

But from Step 16 of Section 5.3(A) $s = \frac{\gamma q_2}{t + \bar{\ell} + \alpha} \mod n$, so

$$(k_1||k_2) = h(x_b q_2^{-1} \frac{\gamma q_2}{t + \bar{\ell} + \alpha} (Y_s + (t + \beta)aG + \alpha G + tG))$$

As $\bar{\ell} = (x_s + \bar{t}a) \mod n$ from Step 12 of Section 5.3(A) and $\bar{t} = (t + \beta) \mod n$ from Step 9 of Section 5.3(A), so

$$(k_1||k_2) = h(x_b \frac{\gamma}{t + x_s + (t + \beta)a + \alpha} (x_s + (t + \beta)a + \alpha + t)G)$$

Also from Key Generation Phase $Y_b = x_b G$, so

$$(k_1||k_2) = h(\gamma Y_b)$$

This shows that same secret key is generated on sender's and receiver's end. The modified scheme correctly verifies the authenticity of the received message because the generated value of r_1 is same as the received value of t i.e $r_1 = kh_{k_2}(m||k_2) = t$.

5.4 Analysis of Modified Scheme

In this section, the security and cost analysis of the modified scheme is presented.

5.4.1 Security Attributes

The modified blind signcryption scheme provides the following security attributes.

Confidentiality

The improved blind signcryption scheme provides confidentiality. An attacker can not be able to obtain the original message m without the secret key $(k_1||k_2)$. If an attacker wants to compute

$$(k_1||k_2) = h(\gamma Y_b \mod n)$$

in Step 5 of blind signcryption Phase then he must have the knowledge of secret parameter γ . The secret random number γ is chosen in Step 5 of Signcryption Phase that is only known to Alice.

Integrity

Our improved scheme provides the message integrity. If an attacker wants to change the ciphertext

$$c = E_{k_1}(m)$$

to c' in Step 8 of Signcryption Phase then its corresponding plaintext message changes from m to m' which effects the hash value

$$t = kh_{k_2}(m||k_2)$$

of message m in Step 6 of Signcryption Phase. Due to these changes, the verification process in Step 7 of Unsigncryption Phase unable to verify the integrity of original message.

Unforgeability

The modified blind signcryption scheme provides the unforgeability property. Only the authentic sender can generate the valid signature that is verified by the unsigncryption algorithm. The signature generation process involves secret parameter

$$t = kh_{k_2}(m||k_2)$$

in Step 6 of Signcryption Phase that is only known to authentic sender. If an attacker wants to generate t then he has to compute secret parameter k_2 in Step 5 of Signcryption Phase. But it is computationally infeasible for an attacker to compute $(k_1||k_2) = h(\gamma Y_b \bmod n)$ because the secret number γ is chosen randomly in Step 4 of blind Signcryption Phase.

Blindness

The modified scheme provides the blindness property. The signer of a message cannot able to see the contents of original message during the blind signature process. Alice uses the secret random key k_2 for blinding a message. If signer wants to get the original message m from $t = kh_{k_2}(m||k_2)$ in Step 9 of Signcryption Phase then he must have the knowledge of secret parameter k_2 . The attacker is unable to get these secret parameters because it is only known to Alice.

Forward Secrecy

The proposed scheme provides the additional security attribute of forward secrecy. In modified scheme, if the long term private key of signer is disclosed then no one can get the contents of original message because it involves the random number γ in Step 5 of Signcryption Phase. This random number is changed each time when the message is encrypted.

The comparison of security attributes of modified scheme with the proposed scheme

[87] is described in Table 5.2.

Scheme	Confidentiality	Integrity	Unforgeability	Non-repudiation	Forward secrecy	Blindness	Authentication
Riazullah et al. [87]	yes	no	no	no	yes	yes	no
Modified Scheme	yes	yes	yes	yes	yes	yes	yes

TABLE 5.2: Comparision of Modified Scheme with the Proposed Scheme [87]

5.4.2 Attack Analysis

In this section, the attack analysis of the modified scheme is presented. It has shown that the modified scheme has resistance against the various attacks.

Forgery Attack

In this attack model, an attacker observes the network communication between the participants. The aim of the attacker is to generate the forge signature on the fake message in such a way that the receiver correctly verifies it. The blind signcryption scheme [87] is vulnerable against this attack as presented in Section 5.2 and therefore modified scheme is proposed to overcome the issues. Suppose an attacker wants to generate a signature

$$s = \frac{\gamma q_2}{t + \bar{\ell} + \alpha} \mod n$$

in Step 16 of Signcryption Phase. The process of the generation of signature s involves the parameters $\alpha, \gamma, t, \bar{\ell}$ and q_2 . The attacker must have the knowledge of these parameters in order to forge the signature on his desired message. The secret random numbers α and γ chosen in Step 4 of Signcryption Phase is only known to authentic sender. The generation of

$$t = kh_{k_2}(m||k_2)$$

involves secret key k_2 that is only known to the authentic sender and the receiver. The secret parameter q_2 generated in Step 5 of Signcryption Phase is involved in

signature generation process that is only known to authentic participants Alice and Bob. It is important here to mention that the generation of q_2 involves secret key x_s of signer in modified scheme. Also the parameter

$$\bar{\ell} = (x_s + \bar{t}a) \mod n$$

generated in Step 12 of Signcryption Phase has secret parameters x_s and a . To find x_s , given $Y_s = x_s G$ and G is ECDLP. Without the knowledge of these parameters, the attacker can not generate a valid signature s' that is verified in Unsigncryption Phase. So the modified scheme has resistance against this attack.

Man-In-The-Middle Attack

In Man-In-The-Middle Attack (MITM), the attacker indulge himself in between the communication of the sender and the receiver. The basic aim of the attacker is to establish the common key with each of the participant anonymously for transmission of his desired information. In the modified scheme, suppose an attacker insert himself in between the communication of sender and the receiver and try to make trustful connection with them. Suppose the attacker chooses his private key d_A and compute his public key $U_A = d_A G$. But the attacker will not be able to establish the common shared secret key with his public key because key generation phase involves secret random number γ in Step 5 of Signcryption Phase. The fake random number γ leads to different shared key in Step 4 of Unsigncryption Phase. So an attacker cannot generate a common shared key with any of the participant. So the modified scheme resist against this attack.

5.4.3 Computational Cost

The modified scheme uses elliptic curve cryptography for key generation as well as for blind digital signature generation and verification phases. The main benefit of ECC is its smaller key size as compared to other public key cryptosystem like

ElGamal [9] and RSA [10]. The proposed scheme [87] is vulnerable to the forgery attack and unable to provide the basic security requirements. To counter this attack, the process of generation of blind signature is modified and extra operations are added. So the modified scheme has greater computational cost as compared to proposed scheme of Riazullah et al. [87]. The comparison of major operations involved in blind signcryption scheme [87] and existing schemes are described in its Table 2. The comparison of modified scheme with the blind signcryption scheme [87] is described in Table 5.3.

Scheme	Hash	ECPM	ECPA	EXP	DIV	MUL	ADD	SADD
Riazullah et al. [87]	2	4	2	-	1	4	6	-
Modified Scheme	2	7	2	-	1	4	6	-

TABLE 5.3: Comparison of Major Operations of Modified Scheme with the Scheme Proposed in [87]

Hash: One way hash function, ECPA: Elliptic curve point addition, ECPM: Elliptic curve point multiplication, Div: Modular division, EXP: Modular exponentiation, ADD: Modular addition MUL: Modular multiplication, SADD: Simple addition.

5.5 Conclusion

In this chapter, the cryptanalysis of a recently proposed blind signcryption scheme of Riazullah et al. [87] is proposed. They claimed that their scheme is secure and has less computational cost as compared to other existing blind signcryption schemes. The claimed security attributes of proposed scheme are confidentiality, authentication, sender anonymity, message integrity, unforgeability, signer non-repudiation, forward secrecy, blindness and message untraceability. In this paper, we analyzed and proved that the proposed scheme of Riazullah et al. [87] is not secure and has many security flaws. After our successful cryptanalysis of this scheme the claimed security attributes of message integrity, authentication, unforgeability and signer non-repudiation are compromised. We modify and improve this scheme to achieve the security requirements of confidentiality, sender anonymity, authentication, message integrity, unforgeability, forward secrecy, blindness and

message untraceability. The material presented in this chapter has been published in journal of Electronics Letters [\[89\]](#).

Chapter 6

Cryptanalysis and Improvement of a Blind Multi-Document Signcryption Scheme

Recall that, Blind Signcryption is used to maintain the anonymity and privacy of the sender from other participants in an unsecured public network. It has vast applications for privacy related mechanisms such as electronic voting and electronic auction systems. In this chapter, a recently proposed blind signcryption scheme [4] for multiple digital documents, that is based upon hyperelliptic curve, is analyzed. The cryptanalysis of the scheme [4] shows that the proposed blind signcryption scheme is not secure against the existing attacks. An adversary, with the knowledge of public parameters, can modify the signcrypted text of his choice. The successful cryptanalysis of the proposed scheme shows that it does not provide the security attributes of authentication and message integrity. To overcome the security flaws, a modified version of this scheme is also proposed. The blind signcryption scheme [4] is described in Section 6.1 together with its cryptanalysis is in Section 6.2. The modified scheme is proposed in Section 6.3 and the analysis of the modified scheme is described in Section 6.4.

Variables	Description
q, n	Prime numbers greater than 2^{128}
Kh	Keyed one way hash function
h	One way hash function
H	Hyperelliptic curve over finite field \mathbb{F}_q
D	A divisor, which generates a group, of order n
E_k	Symmetric encryption algorithm with secret key k
D_k	Symmetric decryption algorithm with secret key k

TABLE 6.1: Global Parameters of the Proposed Scheme [4]

6.1 Blind Signcryption [4]

Recently, Fazlullah et al. [4] introduced a new blind signcryption scheme that uses the computations on the hyperelliptic curves. The main benefit of hyperelliptic curve is that it uses smaller keys as compared to other public key cryptosystems like Elliptic curve cryptography [8], RSA [10] and ElGamal [9]. Their scheme combines the role of multiple signatures in a single signature, which reduces the communication and computational cost. The claimed security properties of proposed scheme are message integrity, forward secrecy, message untraceability, sender anonymity, signer non-repudiation, unforgeability, confidentiality, blindness and authentication. The proposed scheme uses a single signature for multiple digital documents to overcome the computational and communication cost. The proposed scheme is described as follows:

(A) Global Parameters The global parameters of the proposed scheme are described in Table 6.1.

(B) Key Generation Phase In this phase, each participant selects and generates their private and public keys.

Signer

- Selects a random number $0 < k_s < n$ as a secret key.
- Computes public key $K_s = k_s D$ as a hyperelliptic curve point.

Alice (Sender)

- Selects a random number $0 < k_r < n$ as a secret key.
- Computes public key $K_r = k_r D$ as a hyperelliptic curve point.

Bob (Receiver)

- Selects a random number $0 < k_v < n$ as a secret key.
- Computes public key $K_v = k_v D$ as a hyperelliptic curve point.

(C) Multi-Document Blind Signcryption Phase Suppose Alice wants to transmits an array \mathbf{m} of multiple messages m_i i.e $\mathbf{m} = (m_i)$ to Bob in a unsecure and public network. For generation of blind signcrypted text, Alice performs the following steps:

Signer

1. Chooses a random integer $u \in \{1, 2, 3, \dots, n-1\}$.
2. Computes $W = uD \pmod n$.
3. Sends W to Alice.

Alice

4. Chooses the random integers $\alpha_1, \alpha_2, \alpha_3 \in \{1, 2, 3, \dots, n-1\}$.
5. Computes $K = h(\alpha_3 K_v \pmod n)$.
6. Splits K into two parts $(k_1 || k_2)$.
7. Computes $t = Kh_{k_2}(\mathbf{m} || k_1)$.
8. Computes the array \mathbf{c} of multiple messages as $\mathbf{c} = (c_i) = E_{k_1}(m_i)$ with symmetric encryption by using secret key k_1 .
9. Computes $Y = ((\alpha_3 + \alpha_2)W + \alpha_1 D) \pmod n$.
10. Computes $\bar{t} = (\alpha_2 + \alpha_3) \pmod n$.
11. Sends \bar{t} to signer.

Signer

12. Computes $\bar{s} = (k_s + \bar{t}u) \pmod n$.

13. Sends \bar{s} to Alice.

Alice

14. Computes $s = \frac{\alpha_3}{t + \bar{s} + \alpha_1} \mod n$.

15. Sends (\mathbf{c}, t, s, Y) to Bob.

(D) Multi-Document Unsigncryption Phase Bob receives and decrypts the vector \mathbf{c} of encrypted messages c_i and then verifies the authenticity of received signcrypted text as follows:

Bob

1. Receives (\mathbf{c}, t, s, Y) from Alice.
2. Computes $y = k_v s$.
3. Computes $K = h(y(K_s + Y + tD) \mod n)$.
4. Splits K into two parts $(k_1 || k_2)$.
5. Obtain the array of plaintext messages as $\mathbf{m} = (m_i) = D_{k_1}(c_i)$ with symmetric decryption by using secret key k_1 .
6. Computes $t_1 = Kh_{k_2}(\mathbf{m} || k_1)$.
7. Consider \mathbf{m} as a valid array of plaintext messages if $t_1 = t$, otherwise reject.

Proof of Correctness

Bob and Alice both generate the same shared secret key and then use it to verify the digital signature.

$$\begin{aligned}
 (k_1 || k_2) &= h(y(K_s + Y + tD)) && \text{(Receiver's end)} \\
 &= h(k_v s(k_s D + ((\alpha_3 + \alpha_2)W + \alpha_1 D) + tD)) && \text{(Step 2 of Section 6.1(D))} \\
 &= h(k_v s(k_s D + ((\alpha_3 + \alpha_2)uD + \alpha_1 D) + tD)) && \text{(Step 2 of Section 6.1(C))} \\
 &= h(k_v s(k_s + \bar{t}u + \alpha_1 + t)D) && \text{(Step 10 of Section 6.1(C))} \\
 &= h\left(k_v \frac{\alpha_3}{\bar{s} + \alpha_1 + t} (\bar{s} + \alpha_1 + t) D\right) && \text{(Modular Inverse)} \\
 &= h(\alpha_3 K_v) && \text{(Sender's end)}
 \end{aligned}$$

The same key $(k_1||k_2)$ generated on the sender and the receiver end and then used in the encryption and decryption process. Also the same key is used for the generation and verification of the digital signature i.e $t_1 = t = Kh_{k_2}(\mathbf{m}||k_1)$.

6.2 Cryptanalysis

In present section, the security analysis of the above mentioned blind signcryption scheme of Fazlullah et al. [4] is presented and proved the succesful implimnation of the forgery attack. The cryptananyis of the scheme [4] shows that it is not secure and does not provide the claimed security properties of authentication and message integrity. In particular, the scheme does not have resistance against the forgery attack (For detail see 3.6.5). Suppose Mallory (Attacker) intercepts the network communication between Alice (Sender) and Bob (Receiver) and wants to send a vector of messages m'_i of his choice to Bob. The structure of cryptanalysis of the proposed blind signcryption scheme [4] is described in Figure 6.1. Following steps are required to generate and transmit a signcrypted text of his choice:

(A) Signcryption Phase

Mallory

1. Selects a random number $u' \in \{1, 2, 3, \dots, n-1\}$.
2. Computes $W' = u'D \pmod n$.
3. Chooses random numbers $\alpha'_1, \alpha'_2, \alpha'_3 \in \{1, 2, 3, \dots, n-1\}$.
4. Computes $K' = h(\alpha'_3 K_v \pmod n)$.
5. Splits K' into two parts $(k'_1||k'_2)$.
6. Computes $t' = Kh_{k'_2}(m'_i||k'_1)$.
7. Computes the vector \mathbf{c} of ciphertext messages $c'_i = E_{k'_1}(m'_i)$ with symmetric encryption by using secret key k'_1 .
8. Computes $Y' = ((\alpha'_3 + \alpha'_2)W' + \alpha'_1 D) \pmod n$.
9. Computes $\bar{t}' = (\alpha'_3 + \alpha'_2) \pmod n$.

10. Computes $s' = \frac{\alpha'_3}{t' + \bar{t}'u' + \alpha'_1}$.
11. Computes $Y'_1 = Y' - K_s$.
12. Sends (c'_i, t', s', Y'_1) to Bob.

(B) Unsignryption Phase

Bob

1. Receives (c'_i, t', s', Y'_1) .
2. Computes $y' = k_v s'$.
3. Computes $K' = h(y'(K_s + Y'_1 + t'D) \bmod n)$.
4. Splits K' into two parts $(k'_1 || k'_2)$.
5. Obtains the vector of plaintext messages $m'_i = D'_{k'_1}(c'_i)$ with symmetric decryption by using the secret key k'_1 .
6. Computes $t'_1 = Kh_{k'_2}(m'_i || k'_1)$.
7. Consider m'_i as a valid vector of plaintext messages after verifying the relation $t'_1 = t'$.

6.2.1 Proof of Correctness

Bob correctly verifies the authenticity of the received data because the same shared secret key is used for generation and verification of digital signature by Bob and Mallory.

$$\begin{aligned}
 (k'_1 || k'_2) &= h(y'(K_s + Y'_1 + t'D)) && \text{(Receiver's End)} \\
 &= h(k_v s'(K_s + Y' - K_s + t'D)) && \text{(Step 11 of Signcryption Phase)} \\
 &= h(k_v s'(Y' + t'D)) && \text{(Additive Inverse)} \\
 &= h(k_v s'(((\alpha'_3 + \alpha'_2)W' + \alpha'_1 D) + t'D)) && \text{(Step 8 of Signcryption Phase)} \\
 &= h(k_v s'((\alpha'_3 + \alpha'_2)u'D + \alpha'_1 D) + t'D)) && \text{(Step 2 of Signcryption Phase)} \\
 &= h(k_v s'(\bar{t}'u' + \alpha'_1 + t')D) && \text{(Step 9 of Signcryption Phase)} \\
 &= h(k_v \frac{\alpha'_3}{t' + \bar{t}'u' + \alpha'_1} (\bar{t}'u' + \alpha'_1 + t')D) && \text{(Step 10 of Signcryption Phase)} \\
 &= h(\alpha'_3 K_v) && \text{(Sender's End)}
 \end{aligned}$$

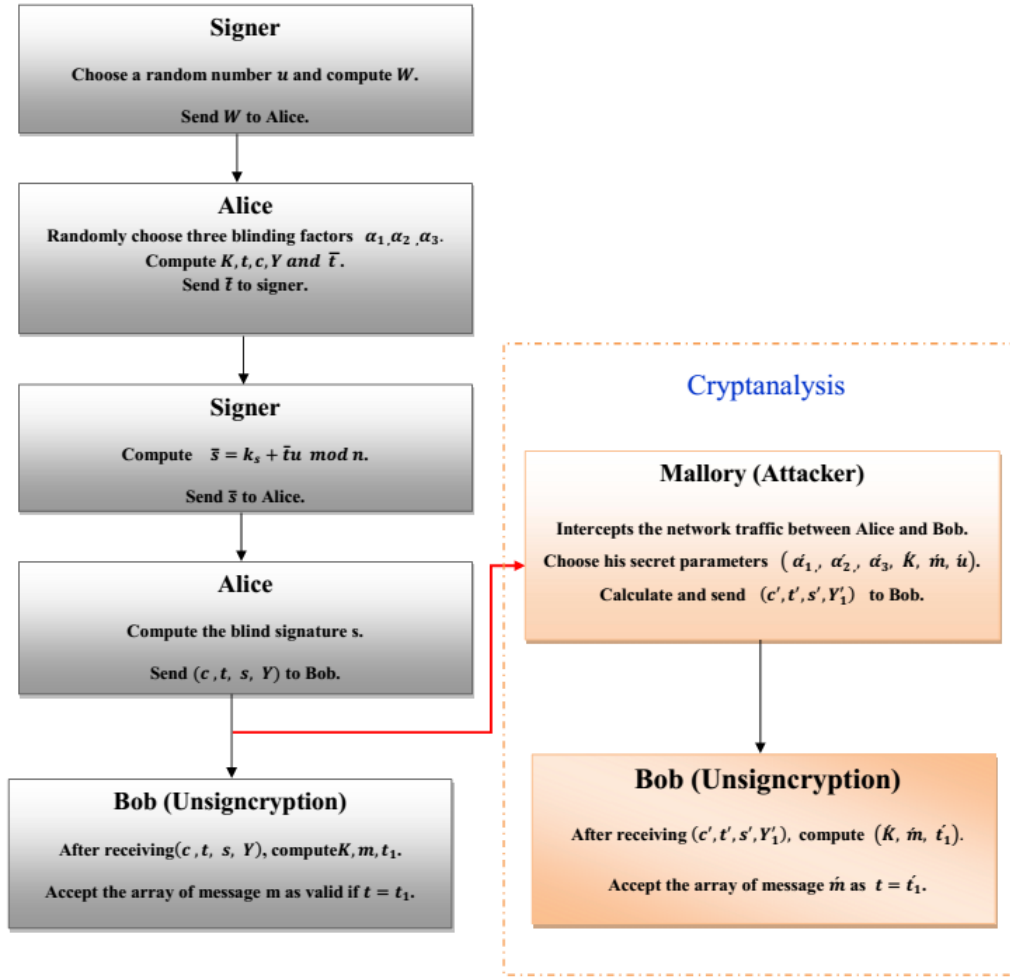


FIGURE 6.1: Structure of Blind Signcryption Scheme [4] and its Cryptanalysis

Now the fake digital signature $t'_1 = Kh_{k'_2}(m'_i || k'_1) = t'$ generated on Mallory's end is correctly verified on Bob's end i.e $t'_1 = t' = Kh_{k'_2}(m'_i || k'_1)$

6.2.2 Security Flaws

In this section, the security analysis of the signcryption scheme [4] is provided. After applying the forgery attack, our analysis shows that the blind signcryption scheme [4] does not provide the security attributes of integrity and authentication. The proposed scheme [4] provides the security attribute of unforgeability in the sense that an adversary can not obtain a valid signature on the original multiple

document message \mathbf{m} . But a valid signature can be generated on a fake message \mathbf{m}' .

(a) Integrity

The scheme [4] cannot provide integrity of multiple messages. The authors [4] claimed that, if an adversary will change the original message \mathbf{m} to \mathbf{m}' then it will be detected automatically on the unsignryption process. Due to our successful cryptanalysis, an adversary will be able to replace the original message with his desired message \mathbf{m}' in Step 6 and Step 7 of signcryption algorithm. The unsign-encryption algorithm fails to detect the modification in the original message \mathbf{m} , as the signature will be successfully verified in Step 7 at the receiver's end.

(b) Authentication

The proposed scheme does not provide authentication. Our analysis shows that an adversary can send a signcrypted text of his choice to a receiver. The unsign-encryption algorithm uses the public key of the signer for verification of the digital signature in Step 3 and Step 7 of unsignryption algorithm. It verifies the signature on the received message and hence believes that the message is signed by the authentic signer and then sent by the authentic sender.

6.3 Modified Blind Signcryption Scheme

In previous section, we have seen that the scheme [4] is not secure against forgery attack. To counter such attack, an improved version of their scheme is proposed. The proposed modified scheme involves the private key of signer in different phases of signature generation process and public key in the signature verification process. In modified scheme, only authentic sender can generate a digital signature that can be verified by the unsignacryption algorithm. The improved blind signcryption scheme is described below:

(A) Generation Phases

In this phase, each participant selects and generates their private and public keys.

Signer

- Selects a random number $0 < k_s < n$ as a secret key.
- Computes public key $K_s = k_s D$ as a hyperelliptic curve point.

Alice (Sender)

- Selects a random number $0 < k_r < n$ as a secret key.
- Computes public key $K_r = k_r D$ as a hyperelliptic curve point.

Bob (Receiver)

- Selects a random number $0 < k_v < n$ as a secret key.
- Computes public key $K_v = k_v D$ as a hyperelliptic curve point.

(B) Multi-Document Signcryption**Signer**

1. Chooses a random integer $u \in \{1, 2, 3, \dots, n-1\}$.
2. Computes $W = uD \pmod n$.
3. Sends W to Alice.

Alice

4. Chooses random numbers $\alpha_1, \alpha_2, \alpha_3, \alpha_4 \in \{1, 2, 3, \dots, n-1\}$.
5. Computes $Y = ((\alpha_3 + \alpha_2)W + \alpha_1 D) \pmod n$.
6. Computes $\bar{t} = (\alpha_3 + \alpha_2) \pmod n$.
7. Computes $P_1 = \alpha_4 K_v$.
8. Sends (\bar{t}, P_1) to signer.

Signer

9. Computes $\bar{s} = (k_s + \bar{t}u) \pmod n$.

10. Computes $P_2 = k_s P_1$.
11. Sends (\bar{s}, P_2) to Alice.

Alice

12. Computes $P_3 = \alpha_4^{-1} P_2 = (P_1, P_2)$.
13. Computes $K = h(\alpha_3 K_v \bmod n)$.
14. Splits K into two parts $(k_1 || k_2)$.
15. Computes $t = K h_{k_2}(\mathbf{m} || P_1)$.
16. Computes the array \mathbf{c} of multiple messages as $\mathbf{c} = (c_i) = E_{k_1}(m_i)$ with symmetric encryption by using secret key k_1 .
17. Computes $s = \frac{\alpha_3}{t + \bar{s} + \alpha_1} \bmod n$.
18. Sends (\mathbf{c}, t, s, Y) to Bob.

(C) Multi-Document Unsignryption

Bob

1. Receives (\mathbf{c}, t, s, Y) from the Alice.
2. Computes $y = k_v s$.
3. Computes $K = h(y(K_s + Y + tD) \bmod n)$.
4. Splits K into two parts $(k_1 || k_2)$.
5. Obtain the array of plaintext messages as $\mathbf{m} = (m_i) = D_{k_1}(c_i)$ with symmetric decryption by using secret key k_1 .
6. Computes $P_3 = k_v K_s = (P_1, P_2)$.
7. Computes $t_1 = k h_{k_2}(\mathbf{m} || P_1)$.
8. if $t_1 = t$ then consider m_i as a valid vector of plaintext messages otherwise reject.

6.3.1 Proof of Correctness

Alice and Bob generate the same shared secret key and use it to generate and verify the digital signature.

$$\begin{aligned}
(k_1||k_2) &= h(y(K_s + Y + tD)) && \text{(Receiver's End)} \\
&= h(k_v s(K_s + Y + tD)) && \text{(Step 2 of Unsignryption Phase)} \\
&= h(k_v s(k_s D + Y + tD)) && \text{(Key Generation Phase)} \\
&= h(k_v s(k_s D + ((\alpha_3 + \alpha_2)W + \alpha_1 D) + tD)) && \text{(Step 5 of Signcryption Phase)} \\
&= h(k_v s(k_s D + ((\alpha_3 + \alpha_2)uD + \alpha_1 D) + tD)) && \text{(Step 2 of Signcryption Phase)} \\
&= h(k_v s(k_s + \bar{t}u + \alpha_1 + t)D) && \text{(Step 6 of Signcryption Phase)} \\
&= h\left(k_v \frac{\alpha_3}{\bar{s} + \alpha_1 + t} (\bar{s} + \alpha_1 + t) D\right) && \text{(Step 17 of Signcryption Phase)} \\
&= h(\alpha_3 K_v) && \text{(Sender's End)}
\end{aligned}$$

Due to the generation of same shared secret key $(k_1||k_2)$, both Alice and Bob will generate the same signature parameter $t_1 = kh_{k_2}(\mathbf{m}||P_1) = t$. After this verification, Bob will believe that message is sending by authentic person Alice and signed by authentic Signer. This ensures the correctness of the improved scheme.

6.4 Analysis of Modified Scheme

This section presents the security analysis of the modified blind signcryption scheme.

6.4.1 Security Attributes

The security of modified and improved scheme relies on the hardness of hyperelliptic curve discrete logarithm problem (HECDLP). In security perspective hyperelliptic curve gives the same level of security as compared to elliptic curve and reduces the storage cost, transmission cost and computational power [4]. “HECC with 80 bits key size produces the same level of security when compared to ECC with 160 bits key size and RSA cryptosystem with 1024 bits key size.” [111]. Unlike the original scheme (Section 3), the proposed modified scheme also provides the main security attributes of confidentiality and message integrity.

Confidentiality

The improved blind signcryption scheme provides confidentiality. If an adversary wants to get the original message then he must have the common shared secret key k_1 . The shared key generation process involves the private key k_v of receiver in Step 2 of Unsigncryption Algorithm. Given D and $K_v = k_v D$ to find k_v is a hyperelliptic curve discrete logarithm problem (HECDLP) which is computationally infeasible [93, 97].

Data Integrity

The improved scheme provides data integrity. Our scheme uses the hash value of the multiple messages and after this it is transmitted over a public network. In the improved scheme, if an attacker wants to change \mathbf{m} to \mathbf{m}' then \mathbf{c} changes to \mathbf{c}' . Consequently, the hash value will be changed in Step 7 and then message will be rejected in Step 8.

Unforgeability

The modified scheme also provides the security attribute of unforgeability. In the improved scheme, only the authentic sender can generate a valid signature that is verified by the Unsigncryption Algorithm. If an attacker wants to generate the forge signature s' then it requires secret parameters α_1 and α_3 in Step 17 of Signcryption Algorithm. In Step 6 of Signcryption Algorithm, solving the equation

$$\bar{t} = (\alpha_3 + \alpha_2) \mod n$$

in order to recover the random number α_3 with two unknown variables is infeasible. If an adversary wants to find private key k_v of receiver then he has to solve HECDLP.

Forward Secrecy

In the improved scheme, if the private key of signer is disclosed then an adversary will not be able to decrypt and get the contents of any of the original messages. The improved scheme uses secret random numbers $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ in the process of signcryption. The shared secret key K generated in Step 13 of signcryption algorithm requires the random number α_3 and using random numbers from a high quality source will imply freshness for the secret key in every session. Thus our scheme provides the forward secrecy property.

Blindness

The security attribute of blindness is also guaranteed by this signcryption scheme. The signer of a message cannot see the contents of original message during the blind signature process. For blinding the original message, Alice uses the secret random numbers α_3, α_4 and P_1 in Step 7, 13 and 15 of blind signcryption process. If signer wants to see the contents of the original message then he has to solve HECDLP for finding these secret random numbers in Step 7 of Signcryption Algorithm.

Authentication

The modified blind signcryption scheme provides the security property of authentication. The signer uses his private key k_s in Step 9 and Step 10 of Signcryption Algorithm for generation of digital signature s . The receiver uses the public key of signer K_s to generate the common shared secret key K in Step 3 of Unsigncryption Algorithm. This shared secret key is used to verify the identity of signer in Step 8 of Unsigncryption Algorithm.

6.4.2 Resistance against the Attack

The proposed blind signcryption scheme of Fazlullah et al. [4] is vulnerable against the forgery attack as discussed in Section 6.2. To counter the attack, the modified scheme is proposed in Section 6.3. Suppose Mallory wants to generate the digital signature on his desired message. The generation of digital signature

$$s = \frac{\alpha_3}{t + \bar{s} + \alpha_1} \mod n$$

involves the parameters α_1, α_3, t and \bar{s} . If Mallory wants to forge the digital signature then he must have the knowledge of these parameters. The secret random numbers α_1 and α_3 are chosen in Step 4 of Signcryption Phase that is only known to authentic sender Alice. The generation of secret parameter $\bar{s} = (k_s + \bar{t}u) \mod n$ involves the secret key k_s of signer. To find k_s , given $K_s = k_s D$ and D is a Hyperelliptic curve discrete logarithm problem that is computationally infeasible to solve. The generation of $t = Kh_{k_2}(\mathbf{m}||P_1)$ involves secret key k_2 and secret number P_1 in Step 15 of Signcryption Phase that are only known to Alice. Without these parameters, the Mallory cannot generate a valid signature s that is verified by unsigncryption algorithm in Step 8 of Unsigncryption Phase.

6.5 Conclusion

Recently, Fazlullah et al. [4] introduced a new blind signcryption scheme that uses the computations on a hyperelliptic curve. The proposed cryptanalysis shows that their scheme is not secure and unable to provide the basic security requirements of blind signcryption. Due to the successful cryptanalysis, an adversary can mount a forgery attack by sending a signcrypted text of his own choice and unsigncryption algorithm will verify it correctly. Thus their scheme does not provide some of the claimed security properties of authentication and message integrity. To overcome the security issues, we propose a modified and improved form of this scheme. We analyze the security of the modified scheme in detail. Our analysis indicates

that the improved scheme should be able to provide all the security attributes required i.e blindness, unforgeability, data integrity, authentication, confidentiality and forward secrecy. The material presented in this chapter has been published in journal of Cryptologia [\[90\]](#).

Chapter 7

A Multi Recipient Aggregate Signcryption Scheme based on Elliptic curve

Recall that, a cryptographic technique “signcryption” combines the role of digital signature and encryption in a single logical step. This helps in reducing the computational cost associated with the traditional approach of signature-then-encryption technique. In last two decades, several signcryption schemes were proposed for single and multiple recipients, each having its own drawbacks and benefits. In this chapter, we introduced an efficient signcryption scheme that uses the elliptic curve cryptography and is capable of transmitting data to single as well as multiple recipients. The scheme consists of different versions and is suitable for transmitting the single and multiple documents to single or multiple recipients. The proposed scheme generates single signature by aggregating the multiple signatures for authentication and verification of data. This reduces the communicational and computational cost associated with the signature generation and verification process. 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. Various versions of the scheme are described

in Section 7.1 and the detailed security and cost analysis are described in Section 7.2.

7.1 Proposed Aggregate Signcryption Scheme

In this section, a new signcryption scheme that uses the elliptic curve for digital signature and encryption is proposed. In the proposed scheme, Alice (Sender) is capable of sending the same or different messages to a single or multiple recipients. For every message, a distinct signature is generated and then combined together to get a single signature (aggregate signature). It reduces the length of certificate chains as compared to multiple signatures for multiple documents. The analysis of the proposed scheme shows that it provides the security attributes of unforgeability, integrity, forward secrecy, confidentiality, authentication and non-repudiation. The proposed signcryption scheme has four versions.

- **Version-1** Signcryption Scheme for Single Message

The proposed scheme uses the elliptic curve cryptography for both the generation/verification of the digital signature and the encryption/decryption of data. In this Version, the sender is able to transmits the single message to single receiver. The more description of this Version followed by its correctness is highlighted in Section 7.1.1 .

- **Version-2** Aggregate Signcryption Scheme for Multiple Messages

In this version, the sender is able to transmits the multiple messages to single recipient. The single signature is generated from multiple signatures and then used it to verify the authenticity of the data. Section 7.1.2 is reserved for the detailed description of this Version.

- **Version-3** Multi-Recipient Signcryption Scheme for Single Message

The proposed scheme is capable of sending the same message to multiple recipients. In this Version, there is a requirement of single signature in such

a way that each receiver will verify the authenticity of the received message. For more detail about this Version, Section 7.1.3 is reserved.

- **Version-4** Multi-Recipient Aggregate Signcryption Scheme for Multiple Message

The scheme is capable of sending the multiple messages to multiple recipients. In this Version, the multiple signatures generated from multiple messages are combine together to get the single aggregate signature. This verification of single signature provides the authenticity of data to each recipient. For more detail about this version, Section 7.1.4 is reserved.

7.1.1 Signcryption Scheme for Single Message (Version-1)

Suppose Alice wants to transmits a message M to Bob through public network. Following four phases are required for generation and verification of the signcrypted text.

- (A) **Global Parameters** In this phase, the global parameters for all communicating participant of the scheme are fixed. These parameters are described in Table ??.

p	Large prime number greater then 2^{160}
\mathbb{F}_p	Working finite field of the scheme
$E_p(a, b)$	Elliptic curve defined over a field \mathbb{F}_p
h	One way hash function
G	A base point of E of order $n > 2^{160}$

TABLE 7.1: Global Parameters of the Proposed Scheme

- (B) **Key Generation Phase**

Alice (Sender)

- Randomly chooses her secret key $d_A < n$.
- Computes her public key $U_A = d_A G$ as a point on the elliptic curve.

Bob (Receiver)

- Randomly chooses his secret key $d_B < n$.
- Computes her public key $U_B = d_b G$ as a point on the elliptic curve.

(C) Signcryption Phase Let Alice wants to transmit a message M to Bob through a unsecured public network. First Alice converts the original message M into the elliptic curve points by using the mapping from alphanumeric characters to elliptic curve points [13]. Alice involves her own private key d_A and public key U_B of Bob to encrypt the message. To generate the signcrypted text, the **sender** has to perform the following steps.

1. Verify the public key U_B of Bob by using his certificate.
2. Chooses a random number $r < n$.
3. Computes the elliptic curve point as $R = rG = (r_1, r_2)$
4. Uses Bob's public key U_B to compute another elliptic curve point

$$A = rU_B = (k, \ell)$$

5. Uses her private key d_A to compute the ciphertext as the pair of encrypted points as

$$C = \{(d_A R), (M + d_A A)\}$$

6. For generating the signature s , she uses the value of ℓ and k from A to compute C' and C'' as follow.

$$C' = \{(d_A R), \ell(M + d_A A)\} = [(p'_1, p'_2), (p'_3, p'_4)]$$

$$C'' = \{((p'_1 + k)\ell, (p'_2 + k)\ell), ((p'_3 + k)\ell, (p'_4 + k)\ell)\} = \{(p_1, p_2), (p_3, p_4)\}$$

7. Using C'' , compute an integer d by adding x and y components of points in C'' that is

$$d = \sum_{j=1}^4 p_j = p_1 + p_2 + p_3 + p_4$$

Note that, each point of C'' is used for computation of d .

8. Using the hash function h , she computes the signature $s = h(d||k)$.
9. Sends (C, R, s) to Bob.

(D) Unsigncryption Phase After receiving (C, R, s) , first Bob verifies the authenticity of the received message and then decryption is performed to recover the original message M . For this, the **receiver** has to perform the following steps.

1. Verifies the public key U_A of Bob by using his certificate.
2. Uses his private key d_B to compute an elliptic curve point as,

$$A = d_B R = (k, \ell)$$

3. For verification of digital signature s , he uses the values of ℓ and k from A to compute

$$C' = \{(d_A R), \ell(M + d_A A)\} = [(p'_1, p'_2), (p'_3, p'_4)]$$

4. Using C' , computes an integer y by adding x, y components of all elliptic curve points in C' that is

$$y = \sum_{j=1}^4 p'_j = p'_1 + p'_2 + p'_3 + p'_4$$

Each point of C' is used for computation of y .

5. Using the value of y , computes $d' = (y + 4k)\ell$.
6. Use the hash function h for computing the signature parameters as $s' = h(d' || k)$.
7. If $s = s'$, then accept ciphertext message C as valid and original message otherwise reject.

8. Use his private key d_B to compute

$$C_1 = \{(d_A d_B R), (M + d_A A)\} = \{(d_A A), (M + d_A A)\}$$

9. To get the original message M , the first part of ciphertext C_1 is subtracted from second part to get the original message M as

$$M = (M + d_A A) - (d_A A)$$

Proof of Correctness

The proposed scheme is correctly verifiable. Any intended receiver can decrypt the original message with his/her private key. For instance, after receiving the ciphertext C , the Bob multiplies the first point $d_A R$ of ciphertext with d_B in Step 8 of Unsigncryption Phase 7.1(D) to get $d_A d_B R = d_A A$. After this, the first part of the ciphertext $d_A A$ is subtracted from the second part $M + d_A A$ to get the original plaintext message M in Step 9 of Unsigncryption Phase 7.1(D) that is,

$$(M + d_A A) - (d_A A) = M$$

Further, $s' = h(d' || k) = h(d || k) = s$ if and only if $d = d'$. Note that s is the signature generated by the signcryption algorithm whereas s' is the signature computed by the unsigncryption algorithm. From Step 6 and 7 of Signcryption Phase and

Step 4 and 5 of Unsignryption Phase, it follows that

$$\begin{aligned}
 d &= \sum_{j=1}^4 p_j \\
 &= \sum_{j=1}^4 (p'_j + k)\ell \\
 &= \sum_{j=1}^4 (p'_j \ell + k\ell) \\
 &= \sum_{j=1}^4 p'_j \ell + \sum_{j=1}^4 k\ell \\
 &= y\ell + 4k\ell \\
 &= d'
 \end{aligned}$$

7.1.2 Aggregate Signcryption Scheme for Multiple Messages (Version-2)

Suppose Alice wants to transmit a vector of message M_i to Bob through a unsecured channel. First Alice converts the vector of original message M_i into elliptic curve points by using the mapping from alphanumeric characters to elliptic curve points [13]. The global parameter and key generation Steps are same as described in Version-1. Rest of the scheme is described in following two phases.

(A) Signcryption Phase For the transmission of multiple messages M_i , the message M is replaced with the vector of messages M_i in the Step 5 of Signcryption Phase (Version-1). It will change the Step 5 and Step 6 of Signcryption Phase and consequently generate c_i, c'_i and c''_i . The above mentioned changes will effect d in the Step 7 of signcryption Phase and each point of C''_i is used for computation of d . The single signature s is generated from the vector of message M_i in Step 8 of Signcryption Phase. For generation of the signcrypted text, the **sender** has to perform the following steps.

1. Verify the public key U_B of Bob by using his certificate.

2. Chooses a random number $r < n$.
3. Computes the elliptic curve point as

$$R = rG = (r_1, r_2)$$

4. Uses Bob's public key U_B to compute another elliptic curve point as

$$A = rU_B = (k, \ell)$$

5. Uses her private key d_A to calculate the ciphertext as the pair of encrypted points as

$$C_i = \{(d_A R), (M_i + d_A A)\}$$

6. For generating the signature s , she uses the value of ℓ and k from A to compute C'_i and C''_i as follow.

$$\begin{aligned} C'_i &= \{(d_A R), \ell(M_i + d_A A)\} = [(p'_{i1}, p'_{i2}), (p'_{i3}, p'_{i4})] \\ C''_i &= \{((p'_{i1} + k)\ell, (p'_{i2} + k)\ell), ((p'_{i3} + k)\ell, (p'_{i4} + k)\ell)\} \\ C'''_i &= \{(p_{i1}, p_{i2}), (p_{i3}, p_{i4})\} \end{aligned}$$

7. Using C'''_i , compute an integer d by adding the components of all points in C'''_i for each $i \in \{1, 2, 3, \dots, N\}$ that is

$$d_i = \sum_{j=1}^{j=4} p_{ij} = p_{i1} + p_{i2} + p_{i3} + p_{i4}$$

Each point of C'''_i is used for computation of d_i .

8. Computes $d = \sum_{i=1}^{i=N} d_i$ by using each d_i .
9. Using the hash function h , she computes the signature $s = h(d||k)$.
10. Send (C_i, R, s) to recipients.

(B) Unsigncryption Phase After receiving (C_i, R, s) , first each recipient verifies the authenticity of the received message and then decryption is performed to recover the vector of original message M_i . For this, the **receiver** has to perform the following steps.

1. Verify the public key U_A of Bob by using his certificate.
2. Uses his private key d_B to calculate an elliptic curve point

$$A = d_B R = (k, l)$$

3. For verification of digital signature s , he uses the value of ℓ and k from A to compute

$$C'_i = \{(d_A R), l(M_i + d_A A)\} = \{(p'_{i1}, p'_{i2}), (p'_{i3}, p'_{i4})\}.$$

4. Using C'_i , computes y_i by adding components of all points on C'_i that is

$$y_i = \sum_{j=1}^{j=4} p'_{ij} = p'_{i1} + p'_{i2} + p'_{i3} + p'_{i4}$$

Each point of C'_i is used for computation of y_i for every $i \in \{1, 2, 3, \dots, N\}$.

5. Computes the sum of all points of y_i as $y = \sum_{i=1}^{j=N} y_i$
6. Using the value of y , computes $d' = (y + 4k)l$.
7. Use the hash function h for computing the signature parameters as $s' = h(d' || k)$.
8. If $s = s'$, then accept the ciphertext message C_i as valid and original message M otherwise reject.
9. Use his private key d_B to compute

$$C_{i1} = \{(d_A d_B R), (M + d_A A)\} = \{(d_A A), (M + d_A A)\}.$$

10. For decryption of the vector of messages M_i , subtracting the first part of C_{i1} from second part to gets the original messages M as

$$(M_i + d_A A) - (d_A A) = M_i$$

The correctness of this version of signcryption scheme is identical as described in Version-1 Section 7.1.2.

7.1.3 Multi-Recipient Signcryption Scheme for Single Message (Version-3)

In this section, a new multiple recipients signcryption scheme that uses the computations of elliptic curve is introduced. Suppose Alice wants to send a message M to N recipients $\{r_1, r_2, r_3, \dots, r_N\}$ in a secure and efficient way. The global parameters are same as described in Version-1. Following phases are required for transmission of message M .

(A) Key Generation Phase

Sender

- Randomly chooses her secret key $d_A < n$.
- Computes her public key $U_A = d_A G$ as a point on the elliptic curve.

Receiver

- Randomly chooses his secret key $d_i < n$.
- Computes his public key $U_i = d_i G$ as a point on the elliptic curve.

(B) Signcryption Phase Suppose Alice wants to send a message M to recipients $\{r_1, r_2, r_3 \dots r_N\}$. First Alice converts the message M into an elliptic curve point [13] and then used his own private key d_A for encryption of a message. To generate the signcrypted text, the **sender** has to perform the following steps.

1. Verify the public key U_i of receiver r_i by using their certificates.
2. Chooses a random number $r < n$.
3. Computes the elliptic curve point as $X = rU_A = (k, \ell)$.
4. Uses the receiver's public key U_i to computes the elliptic curve points

$$A_i = d_A U_i = (k_i, \ell_i)$$

5. Uses her private key d_A to compute the ciphertext as the pair of encrypted points for N recipients as

$$C = \{(d_A G), (M + kU_A)\}$$

6. For generating the signature parameters s , she uses the values of ℓ and k from X to compute C' and C'' as follow.

$$\begin{aligned} C' &= \{(d_A G), \ell(M + kU_A)\} = \{(p'_1, p'_2), (p'_3, p'_4)\} \\ C'' &= \{((p'_1 + k)\ell, (p'_2 + k)\ell), ((p'_3 + k)\ell, (p'_4 + k)\ell)\} \\ C''' &= \{(p_1, p_2), (p_3, p_4)\}. \end{aligned}$$

7. Using C'' , compute an integer d by adding the components of all points on C''' that is

$$d = \sum_{j=1}^t p_j$$

Here t represents the total number of p_j involved in addition. Each point of ciphertext is used for computation of d .

8. Uses the hash function h for computing the signature parameters as $s = h(d||k)$.
9. Uses the values of ℓ_i and k_i in A_i to computes $z_i = \frac{(k_i - r)}{\ell_i} \mod n$.
10. Alice sends (C, z_i, s) to receiver r_i .

(C) Unsigncryption Phase After receiving (C, z_i, s) , receiver r_i first verifies the authenticity of received message and then decryption is performed. For this, the **receiver** has to perform the following steps.

1. Verify the public key U_A of Alice by using her certificate.
2. Use sender's public key U_A to calculate the elliptic curve point

$$A_i = d_i U_A = (k_i, \ell_i)$$

3. Computes the elliptic curve point as $X = (k_i - z_i \ell_i) U_A = (k, \ell)$
4. For verification of digital signature s , he uses the values of ℓ and k from X to compute

$$C' = \{(d_A G), \ell(M + kU_A)\} = \{(p'_1, p'_2), (p'_3, p'_4)\}$$

5. Using C' , computes an integer y by adding components of all points on C' that is

$$y = \sum_{j=1}^t p'_j$$

Here t represents the total number of p'_j involved in addition. Each point of ciphertext is used for computation of y .

6. Using the value of y , computes $d' = (y + kt)l$.
7. Use the hash function h for computing the signature parameters as $s' = h(d' || k)$.
8. If $s = s'$ then accept the ciphertext message C as valid and original message otherwise reject.
9. Use the secret parameter k from X to compute

$$C = \{(kU_A), (M + kU_A)\}$$

10. For decryption of a message, first part of ciphertext is subtracted from second part to get the plaintext message M as

$$(M + kU_A) - (kU_A) = M$$

Proof of Correctness

The proposed scheme is correctly verifiable. Any intended receiver can decrypt the original message with his/her private key. For instance, after receiving the ciphertext C , Bob multiplies the first part of ciphertext U_A with k and obtain kU_A . After this, the first point kU_A of the ciphertext message C is subtracted from the second part $M + kU_A$ in Step 10 of Unsigncryption Phase 7.1.3(C) to get the original plaintext message M that is,

$$\{(M + kU_A) - (kU_A) = M\}$$

Further, $s' = h(d' || k) = h(d || k) = s$. $s = s'$ if and only if $d = d'$. Note that s is the signature generated by the signcryption algorithm whereas s' is the signature computed by the unsigncryption algorithm. From Step 6 and 7 of Signcryption Phase and Step 5 and 6 of Unsigncryption Phase, it follows that

$$\begin{aligned} d &= \sum_{j=1}^t p_j \\ &= \sum_{j=1}^t (p'_j + k)\ell \\ &= \sum_{j=1}^t (p'_j \ell + k\ell) \\ &= \sum_{j=1}^t p'_j \ell + \sum_{j=1}^t k\ell \\ &= y\ell + tk\ell \\ &= d' \end{aligned}$$

7.1.4 Multi-Recipient Aggregate Signcryption Scheme for Multiple Messages (Version-4)

Suppose Alice wants to transmit the multiple messages M_i to N recipients through public network. For the transmission of multiple messages M_i , replace the message M with the vector of messages M_i in the Step 5 of Signcryption Phase of Version-3. It will change the Step 5 and Step 6 of Signcryption Phase and consequently generate c_i, c'_i and c''_i . The above mentioned changes will effect d in the Step 7 of Signcryption Phase. Each point of C''_i is used for computation of d . The single signature s is generated from the vector of message M_i in Step 8 of Signcryption Phase in Version-3. The Key Generation Phase is same as described in Version-3. The rest of the scheme is described in detail as below:

(A) Signcryption Phase Suppose Alice wants to send the vector of message M_i to recipients $\{r_1, r_2, r_3, \dots, r_N\}$. First Alice converts the messages M_i into the elliptic curve points [13] and used his own private key d_A for encrypting the vector of messages M_i . To generate the signcrypted text, the **sender** has to perform the following steps.

1. Verify the public key U_i of receiver r_i by using their certificates.
2. Chooses a random number $r < n$.
3. Computes the elliptic curve point as

$$X = rU_A = (k, \ell)$$

4. Using the recipient's public key U_i to computes the elliptic curve points

$$A_i = d_A U_i = (k_i, \ell_i)$$

5. Uses her private key d_A to computes the vector of ciphertext messages C_i as the pair of encrypted points for N recipients as

$$C_i = \{(d_A G), (M_i + kU_A)\}$$

6. For generating the signature parameters s , she uses the value of ℓ and k from X to compute C'_i and C''_i as follows.

$$\begin{aligned} C'_i &= \{(d_A G), \ell(M_i + kU_A)\} = [(p'_{i1}, p'_{i2}), (p'_{i3}, p'_{i4})] \\ C''_i &= \{((p'_{i1} + k)\ell, (p'_{i2} + k)\ell), ((p'_{i3} + k)\ell, (p'_{i4} + k)\ell)\} \\ C''_i &= \{(p_{i1}, p_{i2}), (p_{i3}, p_{i4})\}. \end{aligned}$$

7. Using C''_i , computes an integer d by adding the x, y components of all elliptic curve points on C''_i that is

$$d_i = \sum_{j=1}^{j=4} p_{ij} = p_{i1} + p_{i2} + p_{i3} + p_{i4}$$

Each point of C''_i is used for computation of d_i , here $i \in \{1, 2, 3, \dots, N\}$

8. Computes $d = \sum_{i=1}^{i=N} d_i$ by using each value of d_i .
9. Uses the hash function h for computing the signature parameters as $s = h(d||k)$.
10. Uses the values of ℓ_i and k_i from A_i to computes $z_i = \frac{(k_i - r)}{\ell_i} \mod n$.
11. Alice sends (C_i, z_i, s) to N recipients.

(B) Unsigncryption Phase After receiving (C_i, z_i, s) , receiver r_i first verifies the authenticity of received vector of messages and then decryption is performed. For this, the **receiver** has to perform the following steps.

1. Verify the public key U_A of Alice by using her certificate.
2. Uses the sender's public key U_A to compute the elliptic curve point

$$A_i = d_i U_A = (k_i, \ell_i)$$

3. Each recipient computes the elliptic curve point as

$$X = (k_i - z_i \ell_i) U_A = (k, \ell)$$

4. For verification of digital signature s , uses the value of ℓ from X to compute

$$C'_i = \{(d_A G), \ell(M_i + kU_A)\} = \{(p'_{i1}, p'_{i2}), (p'_{i3}, p'_{i4})\}$$

5. Using C'_i , computes an integer y by adding components of all points on C'_i that is

$$y_i = \sum_{j=1}^{j=4} p'_{ij} = p'_{i1} + p'_{i2} + p'_{i3} + p'_{i4}$$

Each point of C'_i is used for computation of y_i for every $i \in \{1, 2, 3, \dots, N\}$

6. Computes $y = \sum_{i=1}^{i=N} y_i$ by using the values of y_i .
7. Uses the value of y , computes $d' = (y + kt)l$.
8. Uses the hash function h for computing the signature parameters as $s' = h(d' || k)$.
9. If $s = s'$ then accept the vector of ciphertext messages C_i otherwise reject.
10. Use the secret parameter k in first part of the ciphertext message as

$$C_i = \{(kU_A), (M_i + kU_A)\}$$

11. For decryption, first point of ciphertext is subtracted from second point to get the vector of plaintext messages M_i as

$$M_i = (M_i + kU_A) - (kU_A)$$

The correctness of this version of signcryption scheme is followed from the correctness of Version-3 in Section 7.1.4.

7.2 Analysis of the Proposed Scheme

In this section, the security and cost analysis of the proposed scheme is presented. For instance, the analysis is performed for Version-1 whereas the analysis of rest of the versions are performed on similar basis.

7.2.1 Security Attributes

The proposed scheme fulfills the following security attributes.

Confidentiality

The security of our scheme relies on ECDLP that is secure in the present time. An adversary will not be able to read the contents of the original message without the secret parameters d_A, d_B and A . Recall that from Step 4 of Version-1 of Signcryption Phase 7.1.1(C), without the secret random number r an attacker will not be able to find

$$A = rU_B.$$

To find r from $R = rG$ in Step(3) of Signcryption Phase 7.1.1(C), the attacker has to solve ECDLP but it is computationally infeasible in the given global setting.

Integrity

The proposed scheme provides integrity. After receiving the signcrypted text, receiver will verify that the received message M is not tempered in the process of transmission. If an attacker will change the ciphertext c to c' then consequently $d' = (y + kt)\ell$ changes to d'' in Step 5 of Unsigncryption Phase 7.1.1(D). Due to these changes, signature

$$s' = h(d' || k)$$

generated in Step 6 of Unsigncryption Phase 7.1.1(D) will not be verified. So if the ciphertext c is changed then the receiver will know that the message is tempered during the transmission.

Unforgeability

The proposed scheme provides unforgeability. The adversary cannot generate a valid signature on his desired message M without the secret key of the sender. Suppose an adversary takes any message of his choice M' and generates a signcrypted text (c', s', R') of his choice. But he will not be able to generate the valid signature $s = h(d||k)$ in Step 8 of Signacryption Phase 7.1.1(C) without the knowledge of the secret parameters d and k . Consequently, the Unsigncryption Phase 7.1.1(D) will not verify the signature in Step 7.

Non-repudiation

When dispute occurs between two parties then receiver can send (c, s, R) to judge for the authenticity of message M . The judge will verify the authenticity of original message M by using the signature $s = h(d||k)$ in Step(6) of Unsigncryption Phase 7.1.1(D). The secret key K and random number r is involved in the generation of signature in Step (8) of Signcryption Phase 7.1.1(C), which is only known to authentic sender. So, Alice will not be able to deny being the sender of the message.

Forward Secrecy

Forward secrecy is the additional security requirement of a signcryption scheme. In the proposed scheme, if sender's private key d_A is disclosed then an adversary cannot be able to recover any message from the previous signcrypted text because it involves secret random number r . Note that the knowledge of r requires to solve ECDLP in Step (3) of Signcryption Phase 7.1.1(C). Moreover, the scheme requires

the change of random number r each time a message M is to be signcrypted in Step 2 of Signcryption phase 7.1.1(C). This ensures the forward secrecy capability of the proposed scheme.

Table 7.2 displays the comparison of security attributes with the signcryption schemes provided in [20–22, 24, 25, 47, 49, 50, 86, 105].

Signcryption Schemes	Confidentiality	Integrity	Unforgeability	Non-repudiation	Authentication	Forward secrecy
Zheng [20]	yes	yes	yes	yes	no	no
Jung et al. [24]	yes	yes	yes	yes	no	yes
Iqbal et al. [86]	yes	no	no	no	no	yes
Gamage et al. [25]	yes	yes	yes	yes	yes	no
Elkamchochi [105]	yes	yes	yes	yes	no	no
Bao and deng [22]	yes	yes	yes	yes	no	no
Zheng and Imai [21]	yes	yes	yes	yes	no	no
Han et al. [50]	yes	yes	yes	yes	yes	no
Zhou [47]	yes	yes	yes	yes	yes	no
Mohamed [49]	yes	yes	yes	yes	yes	no
Proposed Scheme (Version-1)	yes	yes	yes	yes	yes	yes

TABLE 7.2: Comparison of Signcryption Scheme with Existing Schemes [86]

7.2.2 Computational Cost

The proposed scheme uses elliptic curve for both digital signature and encryption. The main benefits of ECC is its smaller key size with the equal level of security as compared to ElGamal [9] and RSA [10]. The small key size benefit of ECC is that it reduces the storage requirements. In the proposed scheme, signature generation involves simple arithmetic computations and only one computation of hash function is involved. The comparison of number of major operations involved in the proposed scheme (Version-1) and the existing schemes are given in Table 7.3.

In [112], using the Controller Infineons SLE66CUX640P, a single operation of elliptic curve point multiplication (ECPM) requires 83 ms, whereas a single modular exponentiation requires 220 ms. The comparison of computational cost of major operations involved in our proposed scheme and existing schemes are described in Table 7.4. Moreover, most of the existing signcryption schemes require a symmetric encryption function during the encryption phase of signcryption and hence

Scheme	Hash	ECPM	ECPA	EXP	DIV	MUL	ADD	SADD
Zheng[20]	4	-	-	3	1	2	1	-
Jung et al.[24]	4	-	-	5	1	1	1	-
Bao and Deng[22]	6	-	-	5	1	1	1	-
Gamage et al.[25]	4	-	-	5	1	1	1	-
Iqbal et al. [86]	6	6	2	-	1	3	1	-
Elkamchochi et al. [105]	6	-	-	3	1	4	1	-
Zheng and Lmai[21]	4	3	1	-	1	3	1	-
Han et al [50]	4	5	1	-	2	4	3	-
Zhou [47]	6	6	7	-	1	4	2	-
Mohamed [49]	6	6	1	-	1	-	1	-
Proposed Scheme(Version-1)	2	8	2	-	-	-	-	6

TABLE 7.3: Comparison of Major Operations involved in the Proposed Scheme and Existing Schemes

Hash: One way hash function, ECPA: Elliptic curve point addition, ECPM: Elliptic curve point multiplication, Div: Modular division, EXP: Modular exponentiation, ADD: Modular addition MUL: Modular multiplication, SADD: Simple addition.

have to bear extra computational cost. The last column shows the extra cost associated with the encryption phase. This comparison of the computational cost of our scheme with the existing schemes is graphically highlighted in Figure 7.1.

Scheme	Computational Time(ms)	Extra Cost(Encryption)
Zheng[20]	$3 \times 220 = 660$	Yes
Jung et al.[24]	$5 \times 220 = 1100$	Yes
Bao and Deng[22]	$5 \times 220 = 1100$	Yes
Gamage et al.[25]	$5 \times 220 = 1100$	Yes
Iqbal et al [86]	$6 \times 83 = 498$	Yes
Elkamchochi [105]	$3 \times 220 = 660$	Yes
Zheng and Lmai[21]	$3 \times 83 = 249$	Yes
Han et al [50]	$5 \times 83 = 415$	Yes
Zhou [47]	$6 \times 83 = 498$	Yes
Mohamed [49]	$6 \times 83 = 498$	Yes
Proposed Scheme(Version-1)	$8 \times 83 = 684$	NO

TABLE 7.4: Average Computational Time (in ms) of Major Operations involved in Proposed Scheme and existing schemes

7.2.3 Attack Analysis

In this section, Version-1 of the signcryption scheme proposed in 7.1.1 is analyzed and is found to be resistant against various known attacks.

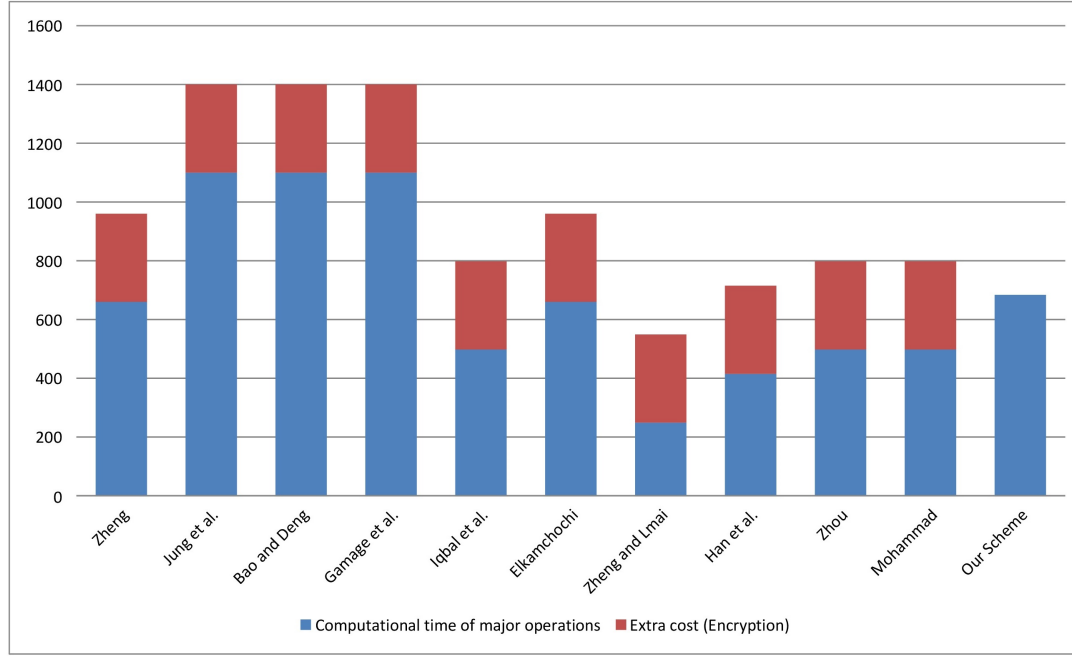


FIGURE 7.1: Comparison of computational Time (in ms) of Major Operations of Proposed Scheme with Existing Schemes

Chosen Plaintext Attack

This type of attack is applicable when an attacker chooses any message of his choice and gets its corresponding ciphertext. The attacker analyzes the relationship between the plaintext and its corresponding ciphertext to guess the secret key. This type of attack is powerful as the attacker can input any message to guess the secret key from the resulting ciphertext. In the proposed scheme, an attacker gets a plaintext M and ciphertext messages $C = [(d_A R), (M + d_A A)]$ and tries to guess the secret key d_A . Given M and C to find a d_A , the attacker has to solve ECDLP which is computationally not possible in the setting of the parameters of the scheme. Hence the scheme resists against this attack.

Ciphertext only Attack

In this attack model, an attacker gets ciphertext message from publicly available information and tries to generate original plaintext message or the secret key. Later on, he gets all the plaintext messages from ciphertext, if the secret key is exposed.

In the proposed scheme, if an attacker gets ciphertext message $C = [(d_A R), (M + d_A A)]$ then, he tries to obtain the secret key d_A or the plaintext message M . Given ciphertext message C and publicly transmitted parameter R to obtain d_A , again, he has to solve ECDLP which is computationally infeasible. Consequently, he will not be able to obtain the original plaintext message M without the knowledge of secret key d_A .

Chosen Ciphertext Attack

In chosen ciphertext attack, an attacker can choose various ciphertext messages of his choice and can get their corresponding plaintext messages. The basic aim of the attacker is to recover the secret key or gets the secret parameters involved in the communication. In the proposed scheme, an attacker chooses a ciphertext C of his choice and obtain its corresponding plaintext message M . Given $C = [(d_A R), (M + d_A A)]$ and M , finding the secret key d_A is not possible because it involves another secret parameter A . If an attacker wants to find $A = rU_B$ in Step 4 of Signcryption Phase then he must have secret random number r . But given $R = rG$ and G , finding r in Step (3) of Signcryption Phase means again to solve ECDLP, which is computationally infeasible with the given parameters of the scheme. So our scheme resists against this attack.

Forgery Attack

In this attack model, an adversary intercepts the network communication between the sender and the receiver. The aim of the attacker is to forge the digital signature in such a way that Unsigncryption Phase 7.1.1(D) correctly verifies it. In the proposed scheme, suppose an adversary intercepts the network traffic between the sender and the receiver. The attacker modifies and generate the signcrypted text (c', s', R') of his choice and sends to the receiver. But the Unsigncryption Algorithm 7.1.1(D) cannot verify the authenticity of the received message. In fact, the generation of signature requires secret parameters, r in Step 2, A in Step

4 of Signcryption Phase 7.1.1(C) and d_A in Key Generation Phase 7.1.1(B), which are not known to an adversary. Thus, without using these secret parameters, the fake signcrypted text cannot be verified by Unsigncryption Algorithm 7.1.1(D). Hence, one cannot mount the forgery attack on the proposed scheme.

Man-In-The- Middle Attack

In this attack, an adversary indulge himself in between the communication of sender and receiver. The aim of the attacker is to establish the common shared secret key with the participants for transmission of his desired message. For protection against this type of attack, a strong authentication protocol is used in communication. The attack model is described in Figure 7.2.

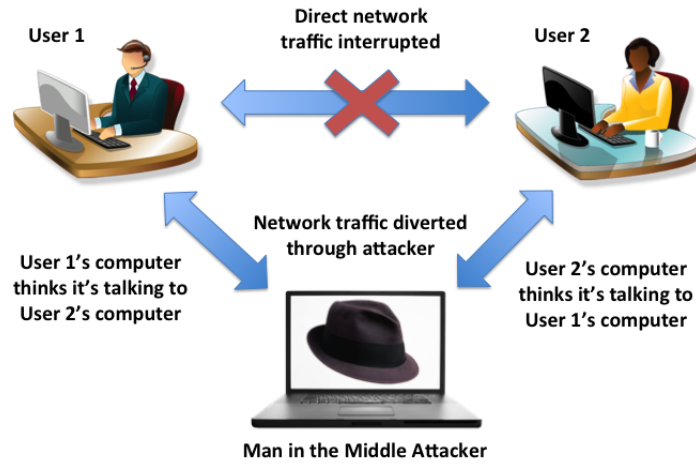


FIGURE 7.2: Man in the Middle Attack

In the proposed scheme, suppose an adversary wants to exploit the process of shared secret key generation. For this purpose, he selects his private key d_M and compute his public key $U_M = d_M G$ as an elliptic curve point. After intercepting the network communication between the sender and the receiver, he wants to make a separate and trustful connections with both sender and the receiver. Firstly, the adversary tries to make a shared secret key with his public key U_M . But he will not be able to establish a correct shared secret key with any of the sender or receiver. The generation of secret key involves secret random number r in Step

4 of Signcryption Phase 7.1.1(C), which is only known to an authentic sender. The selection of a fake random number r in Step 2 of Signcryption Phase 7.1.1(C) leads to different shared key in Unsigncryption Phase 7.1.1(D). This shows that an adversary cannot be able to read the contents of the original message with his generated key. Therefore, the proposed scheme has a resistance against this attack.

7.3 Conclusion

In this chapter, a new signcryption scheme with its different versions are proposed. All four versions are based on elliptic curve and their security depends upon ECDLP. In Version-1, a signcryption scheme is proposed with the facility of sending the single message to the single recipient. It uses the elliptic curve for both the generation/verification of digital signature and for the process of encryption/decryption of messages. The generation of signature requires lesser hash value computations as compared to existing signcryption schemes. The scheme is more efficient as compared to the existing scheme because there is no extra cost associated in the encryption phase of the scheme. The scheme offers the security attributes of integrity, message confidentiality, forward secrecy, unforgeability, authentication and non-repudiation. The security analysis of the proposed scheme shows that it has resistance against various known attacks. The Version-2 of signcryption scheme has the facility of sending the multiple messages to single recipient. It uses the single aggregate signature for the verification of the multiple messages. In Version-3, the single message is transmitted to multiple recipients. It generates the single signature for all the recipients. Version-4 is reserved for transmitting the multiple messages to multiple recipient. The scheme is capable of generating a single signature form multiple messages and recipients. The analysis of the Version-1 is performed in Section 7.2 and the analysis of the rest of the versions are similar to Version-1. The content presented in this chapter has been published in the journal *Wireless Personal Communications* [91].

Chapter 8

Conclusion and Future Work

Recall that, in 1997, Zheng [20] introduced a new cryptographic technique called Signcryption. It simultaneously provides the functionalities of encryption and digital signature in a single logical step. Due to this facility, it reduces the computational as well as communicational cost and, is therefore more efficient as compared to Signature-then-encryption technique. Zheng [20] analysis shows that signcryption scheme reduces 50% computational overheads and 85% communicational cost as compared to traditionally used signature-then-encryption scheme with maintaining the same level of security. After the first signcryption scheme [20], various signcryption schemes were introduced in last two decades. In this research, the security analysis of some existing signcryption schemes has been investigated. Here are the concluding remarks on the entire research work presented in this thesis.

1. In [86], a firewall based signcryption scheme that uses elliptic curve was proposed. Firewall is a security system that monitors the network traffic based on some rules. It is an extra layer of security for authentication schemes. The authors claimed that the scheme [86] provides the security attributes of integrity, message confidentiality, signature unforgeability, public verifiability, non-repudiation, and forward secrecy. The detailed cryptanalysis of this scheme [86] has been carried out in Chapter 4. The analysis shows that it has security flaws and therefore it is not secure against the existing attacks.

Due to the successful cryptanalysis of the scheme, it cannot provide the claimed security attributes of non-repudiation, unforgeability, integrity and authentication. To fix the security flaws, a modified version of this scheme is also proposed in Chapter 4. The modified version is tested against the existing attacks and found it to be secure.

2. Blind signcryption schemes are the extension of signcryption schemes and used in the situation when the sender and the signer of a message are two different entities. It provides the security attributes of anonymity and untraceability in addition to the properties provided by any signcryption scheme. During this work, various blind signcryption schemes [3, 62–65, 68, 71–74, 87] were studied for the purpose of their security features. The detailed security analysis of the blind signcryption scheme presented in [87] is carried out in Chapter 5. The claimed security attributes of the scheme [87] are confidentiality, authentication, sender anonymity, message integrity, unforgeability, signer non-repudiation, forward secrecy, blindness and message untraceability. The cryptanalysis of the scheme in [87] shows that it has security flaws and issues. An attacker can generate a valid signature on his desired message that is acceptable by the unsigncryption algorithm. Due to successful cryptanalysis, the claimed security properties of authentication, message integrity, signer non-repudiation and unforgeability are compromised. To overcome the security flaws, a modified and improved version of the scheme is proposed. The analysis of improved scheme shows that it provides the basic security requirements of blind signcryption scheme.
3. In Chapter 6, the analysis of multi-document blind signcryption scheme [4] is carried out. The proposed scheme [4] used the computations of hyper-elliptic curve and capable of sending the multiple documents on receiver's end. The cryptanalysis of the scheme [4] shows that it is not secure and unable to provide the claimed security attributes of authentication and message integrity. The modified version of this scheme is then introduced to counter the attack. The analysis of the improved scheme shows that it is secure against the existing attacks and provides the security attributes of

blindness, unforgeability, data integrity, authentication, confidentiality and forward secrecy.

4. For multiple digital documents, signing each document separately requires extra computational and communication cost in the process of digital signature. To overcome this issue, the aggregate signature is generated from multiple digital documents instead of multiple signatures and consequently it reduces the length of certificate chain. A new aggregate signcryption scheme with various versions are proposed in Chapter 7. The scheme is based upon elliptic curve and consequently its security depends upon elliptic curve discrete logarithm problem. The proposed scheme is capable of sending, a single message to single recipient (Version-1), multiple messages to single recipient (Version-2), single message to multiple recipients (Version-3) and multiple messages to multiple recipient (Version-4). The security and cost analysis of the proposed scheme is also presented in Chapter 7. The analysis shows that the proposed scheme is secure against the existing attacks and is more efficient as compared to existing schemes. It provides the security attributes of non-repudiation, message confidentiality, forward secrecy, integrity, authentication and unforgeability.

The work presented in Chapter 4, 5, 6 and 7 is published in different international journals.

8.1 Future Work

In this thesis, the security of different signcryption schemes are analyzed and after finding the security flaws their modified and improved versions are also introduced. The improved schemes presented in Chapter 4,5 and 6 of this thesis will be best suited for resource constrained devices like smart devices, sensors, and small computers etc. More precisely, as a future work, some of the possible directions are stated below.

- The aggregate signcryption scheme presented in Chapter 7 can be extended to a generalized signcryption scheme for the possible facility of signcryption mode, signature only mode and encryption only mode.
- The improved blind signature scheme based on elliptic curve presented in Chapter 5 may be further extended and implemented for electronic voting systems and electronic cash payment systems.
- For the secure and authenticated medical image transmission system, the signcryption scheme for firewalls in Chapter 4 has a room for further extension.
- The scheme proposed in Chapter 5 can be extended to ID based signcryption schemes in the setting of elliptic or hyperelliptic curve.
- The proposed schemes may be extended for signcryption schemes that use the algebraic structures so that they provide resistance against quantum computers.
- One can also work for the security analysis of other signcryption schemes by mounting the cryptographic attacks presented in this research.

Bibliography

- [1] Simmons and Gustavus. *Cryptology*. Encyclopedia Britannica, 2016.
- [2] Shivangi Goyal. A survey on the applications of cryptography. *International Journal of Science and Technology*, 2(3), 2012.
- [3] David Chaum. Blind signatures for untraceable payments. In *Advances in cryptology*, pages 199–203. Springer, 1983.
- [4] Jawaaid Iqbal Arif Iqbal Umar Fazlullah, Noor Ul Amin and Muhammad Shahid. Secure and efficient protocol for transmission of multi digital documents using blind signcryption. In *2014 Conference on Information Assurance and Cyber Security (CIACS)*, pages 51–54. IEEE, 2014.
- [5] Don Coppersmith. The data encryption standard (des) and its strength against attacks. *IBM journal of research and development*, 38(3):243–250, 1994.
- [6] Don Coppersmith, Don B Johnson, and Stephen M Matyas. A proposed mode for triple-des encryption. *IBM Journal of Research and Development*, 40(2):253–262, 1996.
- [7] Joan Daemen and Vincent Rijmen. Aes proposal: Rijndael. 1999.
- [8] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of computation*, 48(177):203–209, 1987.
- [9] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, 31(4):469–472, 1985.

- [10] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.
- [11] Patrick Longa and Ali Miri. Fast and flexible elliptic curve point arithmetic over prime fields. *IEEE Transactions on computers*, 57(3):289–302, 2008.
- [12] Daniel V Bailey and Christof Paar. Efficient arithmetic in finite field extensions with application in elliptic curve cryptography. *Journal of cryptology*, 14(3):153–176, 2001.
- [13] Srinivasa Rao and Pallam Setty. Efficient mapping methods for elliptic curve cryptosystems. *International Journal of Engineering Science and Technology*, 2(8):3651–3656, 2010.
- [14] Brian King. Mapping an arbitrary message to an elliptic curve when defined over $\text{gf}(2^n)$. *IJ Network Security*, 8(2):169–176, 2009.
- [15] Mohammed Rafiq Namiq, Wrya K Kadir, and Aram M Ahmed. A new cryptosystem for encryption and decryption using elliptic curves in cryptography over finite fields. *Journal of Theoretical & Applied Information Technology*, 96(1), 2018.
- [16] J Athena, V Sumathy, and K Kumar. An identity attribute-based encryption using elliptic curve digital signature for patient health record maintenance. *International Journal of Communication Systems*, 31(2):e3439, 2018.
- [17] Debiao He, Huaqun Wang, Lina Wang, Jian Shen, and Xianzhao Yang. Efficient certificateless anonymous multi-receiver encryption scheme for mobile devices. *Soft Computing*, 21(22):6801–6810, 2017.
- [18] Subhranil Som. Encryption technique using elliptic curve cryptography through compression and artificial intelligence. In *Cyber Security*, pages 447–457. Springer, 2018.
- [19] Derya Avci. A novel meaningful secret image sharing method based on arabic letters. *Kuwait Journal of Science*, 43(4), 2016.

- [20] Yuliang Zheng. Digital signcryption or how to achieve cost (signature & encryption) less then cost (signature) and cost (encryption). In *Annual international cryptology conference*, pages 165–179. Springer, 1997.
- [21] Yuliang Zheng and Hideki Imai. How to construct efficient signcryption schemes on elliptic curves. *Information processing letters*, 68(5):227–233, 1998.
- [22] Feng Bao and Robert Deng. A signcryption scheme with signature directly verifiable by public key. In *International Workshop on Public Key Cryptography*, pages 55–59. Springer, 1998.
- [23] Ramratan Ahirwal, Anjali Jain, and Jain. Signcryption scheme that utilizes elliptic curve for both encryption and signature generation. *International Journal of Computer Applications*, 62(9), 2013.
- [24] Hee Yun Jung, Dong Hoon Lee, Jong In Lim, and Ki Sik Chang. Signcryption schemes with forward secrecy. *Proceeding of Information Security Application-WISA*, 1:403–475, 2001.
- [25] Chandana Gamage, Jussipekka Leiwo, and Yuliang Zheng. Encrypted message authentication by firewalls. In *International Workshop on Public Key Cryptography*, pages 69–81. Springer, 1999.
- [26] Mohsen Toorani and Ali Beheshti. Cryptanalysis of an elliptic curve-based signcryption scheme. *arXiv preprint arXiv:1004.3521*, 2010.
- [27] Nada Fadhil Mohammed, Suhad Ahmed Ali, and Majid Jabbar Jawad. Biometric-based medical watermarking system for verifying privacy and source authentication. *Kuwait Journal of Science*, 47(3), 2020.
- [28] Baojun Huang and Hang Tu. Strongly secure certificateless one-pass authenticated key agreement scheme. *Kuwait Journal of Science*, 42(1), 2015.

- [29] Xiong Li, Maged Hamada Ibrahim, Saru Kumari, Arun Kumar Sangaiah, Vidushi Gupta, and Kim-Kwang Raymond Choo. Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks. *Computer Networks*, 129:429–443, 2017.
- [30] Xiong Li, Jianwei Niu, Md Zakirul Alam Bhuiyan, Fan Wu, Marimuthu Karuppiah, and Saru Kumari. A robust ecc-based provable secure authentication protocol with privacy preserving for industrial internet of things. *IEEE Transactions on Industrial Informatics*, 14(8):3599–3609, 2017.
- [31] Kelly Grindrod, Hassan Khan, Urs Hengartner, Stephanie Ong, Alexander G Logan, Daniel Vogel, Robert Gebotys, and Jilan Yang. Evaluating authentication options for mobile health applications in younger and older adults. *PloS one*, 13(1):e0189048, 2018.
- [32] Xiong Li, Jieyao Peng, Jianwei Niu, Fan Wu, Junguo Liao, and Kim-Kwang Raymond Choo. A robust and energy efficient authentication protocol for industrial internet of things. *IEEE Internet of Things Journal*, 5(3):1606–1615, 2017.
- [33] Benoit Libert and Jean-Jacques Quisquater. A new identity based signcryption scheme from pairings. In *Proceedings 2003 IEEE Information Theory Workshop (Cat. No. 03EX674)*, pages 155–158. IEEE, 2003.
- [34] Xiuying Yu and Dake He. A new efficient blind signcryption. *Wuhan University Journal of Natural Sciences*, 13(6):662–664, 2008.
- [35] Jianchang Lai, Yi Mu, and Fuchun Guo. Efficient identity-based online and offline encryption and signcryption with short ciphertext. *International Journal of Information Security*, 16(3):299–311, 2017.
- [36] Anuj Kumar Singh and BDK Patro. Performance comparison of signcryption schemes, a step towards designing lightweight cryptographic mechanism. *International Journal of Engineering and Technology (IJET)*, 2017.

- [37] Anuj Kumar Singh and BDK Patro. Elliptic curve signcryption based security protocol for rfid. *KSII Transactions on Internet & Information Systems*, 14(1), 2020.
- [38] Yang Ming and Yumin Wang. Proxy signcryption scheme in the standard model. *Security and Communication Networks*, 8(8):1431–1446, 2015.
- [39] Sunder Lal and Tej Singh. New id based multi-proxy multi-signcryption scheme from pairings. *arXiv preprint cs/0701044*, 2007.
- [40] Huifang Yu and Zhicang Wang. Construction of certificateless proxy signcryption scheme from cmgs. *IEEE Access*, 7:141910–141919, 2019.
- [41] Bo Zhang, Zhongtian Jia, and Chuan Zhao. An efficient certificateless generalized signcryption scheme. *Security and Communication Networks*, 2018, 2018.
- [42] Hui Guo and Lunzhi Deng. An identity based proxy signcryption scheme without pairings. *International Journal of Network Security*, 22(4):561–568, 2020.
- [43] Xuan Wu Zhou. Improved signcryption schemes based on hyper-elliptic curves cryptosystem. In *Applied Mechanics and Materials*, volume 20, pages 546–552. Trans Tech Publ, 2010.
- [44] Manoj Kumar and Pratik Gupta. An efficient and authentication signcryption scheme based on elliptic curves. *MATEMATIKA: Malaysian Journal of Industrial and Applied Mathematics*, 35(1):1–11, 2019.
- [45] Shehzad Ashraf Ch, Muhammad Sher, Anwar Ghani, Husnain Naqvi, Azeem Irshad, et al. An efficient signcryption scheme with forward secrecy and public verifiability based on hyper elliptic curve cryptography. *Multimedia Tools and Applications*, 74(5):1711–1723, 2015.
- [46] S Prasanna Ganesan. An authentication protocol for mobile devices using hyperelliptic curve cryptography. *International J. of Recent Trends in Engineering and Technology*, 3(2):2–4, 2010.

- [47] Xuanwu Zhou. Improved signcryption scheme with public verifiability. In *2009 Pacific-Asia Conference on Knowledge Engineering and Software Engineering*, pages 178–181. IEEE, 2009.
- [48] Sharmila Deva Selvi, Sree Vivek, Rahul Srinivasan, and Chandrasekaran Pandu Rangan. An efficient identity-based signcryption scheme for multiple receivers. In *International Workshop on Security*, pages 71–88. Springer, 2009.
- [49] Elsayed Mohamed and Hassan Elkamchouchi. Elliptic curve signcryption with encrypted message authentication and forward secrecy. *International Journal of Computer Science and Network Security*, 9(1):395–398, 2009.
- [50] Yiliang Han and Xiaoyuan Yang. Ecgsc: Elliptic curve based generalized signcryption scheme. *IACR Cryptol. ePrint Arch.*, 2006:126, 2006.
- [51] Liaojun Pang, Huixian Li, Lu Gao, and Yumin Wang. Completely anonymous multi-recipient signcryption scheme with public verification. *PloS one*, 8(5):e63562, 2013.
- [52] Fahad Ahmed, Asif Masood, and Firdous Kausar. An efficient multi recipient signcryption scheme offering non repudiation. In *2010 10th IEEE International Conference on Computer and Information Technology*, pages 1577–1581. IEEE, 2010.
- [53] Cai-Xue Zhou. An improved multi-receiver generalized signcryption scheme. *International Journal of Network Security*, 17(3):340–350, 2015.
- [54] Pengfei Su, Yong Xie, and Ping Liu. Anonymous and efficient certificateless multirecipient signcryption scheme for ecological data sharing. *Journal of Sensors*, 2020, 2020.
- [55] S Sharmila Deva Selvi, S Sree Vivek, and C Pandu Rangan. Breaking and fixing of an identity based multi-signcryption scheme. In *International Conference on Provable Security*, pages 61–75. Springer, 2009.

- [56] Dan Boneh, Craig Gentry, Ben Lynn, and Hovav Shacham. Aggregate and verifiably encrypted signatures from bilinear maps. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 416–432. Springer, 2003.
- [57] Shi-Jinn Horng, Shiang-Feng Tzeng, Po-Hsian Huang, Xian Wang, Tianrui Li, and Muhammad Khurram Khan. An efficient certificateless aggregate signature with conditional privacy-preserving for vehicular sensor networks. *Information Sciences*, 317:48–66, 2015.
- [58] G Swapna and P Vasudeva Reddy. Efficient identity based aggregate sign-cryption scheme using bilinear pairings over elliptic curves. In *Journal of Physics: Conference Series*, volume 1344, page 012010. IOP Publishing, 2019.
- [59] Eman Abouelkeir and Shamia El-sherbiny. A pairing free identity based aggregate sign-cryption scheme. *IET Information Security*, 2020.
- [60] Hao Wang, Zhen ., Zhe Liu, and Duncan S Wong. Identity-based aggregate sign-cryption in the standard model from multilinear maps. *Frontiers of Computer Science*, 10(4):741–754, 2016.
- [61] Han Yiliang and Chen Fei. The multilinear maps based certificateless aggregate sign-cryption scheme. In *2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, pages 92–99. IEEE, 2015.
- [62] Hui-Feng Huang and Chin-Chen Chang. An untraceable electronic cash system using fair blind signatures. In *2006 IEEE International Conference on e-Business Engineering (ICEBE’06)*, pages 39–46. IEEE, 2006.
- [63] Morteza Nikooghadam and Ali Zakerolhosseini. An efficient blind signature scheme based on the elliptic curve discrete logarithm problem. *ISecure-The ISC International Journal of Information Security*, 1(2):125–131, 2009.

- [64] David Pointcheval and Jacques Stern. New blind signatures equivalent to factorization. In *Proceedings of the 4th ACM conference on Computer and communications security*, pages 92–99, 1997.
- [65] Dhanashree and Agrawal. Implementation of blind digital signature using ecc. *Int. J. Comput. Sci. Netw*, 2(9):316–319, 2012.
- [66] Amit Awasthi and Sunder Lal. An efficient scheme for sensitive message transmission using blind signcryption. *arXiv preprint cs/0504095*, 2005.
- [67] Olivier Delos and Jean-Jacques Quisquater. Efficient multi-signature schemes for cooperating entities. In *Workshop on Algebraic Coding*, pages 63–74. Springer, 1993.
- [68] Kalyan Chakraborty and Jay Mehta. A stamped blind signature scheme based on elliptic curve discrete logarithm problem. *IJ Network Security*, 14(6):316–319, 2012.
- [69] Chien-Hua Tsai and Pin-Chang Su. An ecc-based blind signcryption scheme for multiple digital documents. *Security and Communication Networks*, 2017, 2017.
- [70] Yu Hui-fang, Tian Li-qin, and Wang Zhi-cang. Self-certified proxy blind signcryption scheme from pairings. In *2011 International Conference on Network Computing and Information Security*, volume 1, pages 316–320. IEEE, 2011.
- [71] Masahiro Mambo, Keisuke Usuda, and Eiji Okamoto. Proxy signatures for delegating signing operation. In *Proceedings of the 3rd ACM conference on Computer and communications security*, pages 48–57, 1996.
- [72] Pin-Chang Su and Chien-Hua Tsai. New proxy blind signcryption scheme for secure multiple digital messages transmission based on elliptic curve cryptography. *THIS*, 11(11):5537–5555, 2017.

- [73] Salome James, Gayathri, and Pradesh Vasudeva Reddy. Pairing free identity-based blind signature scheme with message recovery. *Cryptography*, 2(4):29, 2018.
- [74] Sultan Ullah, Muhammad Junaid, Farwa Habib, et al. A novel proxy blind signcryption scheme based on hyper elliptic curve. In *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pages 1964–1968. IEEE, 2016.
- [75] Lin Cheng and Qiaoyan Wen. An improved certificateless signcryption in the standard model. *IJ Network Security*, 17(5):597–606, 2015.
- [76] Zhenhua Liu, Yupu Hu, Xiangsong Zhang, and Hua Ma. Certificateless signcryption scheme in the standard model. *Information Sciences*, 180(3):452–464, 2010.
- [77] Bin Hu, MENGJIE BAO, and NA DONG. Improvement of user authentication protocol with anonymity for wireless communications. *Kuwait Journal of Science*, 41(1), 2014.
- [78] Chun-Ta Li and Cheng-Chi Lee. A novel user authentication and privacy preserving scheme with smart cards for wireless communications. *Mathematical and Computer Modelling*, 55(1-2):35–44, 2012.
- [79] Abdul Waheed, Jawaid Iqbal, Nizamud Din, S Ul, A Iqbal, and N Ul. Improved cryptanalysis of provable certificateless generalized signcryption. *Int. J. Adv. Comput. Sci. Appl*, 10(4):1–7, 2019.
- [80] Vani Rajasekar, Premalatha Jayapaul, and Sathya Krishnamoorthi. Cryptanalysis and enhancement of multi factor remote user authentication scheme based on signcryption. *Advances in Mathematics of Communications*, page 0, 2019.
- [81] Philemon Kasyoka, Michael Kimwele, and Shem Mbandu Angolo. Cryptanalysis of a pairing-free certificateless signcryption scheme. *ICT Express*, 2020.

- [82] Xi-Jun Lin, Lin Sun, Zhen Yan, Xiaoshuai Zhang, and Haipeng Qu. On the security of a certificateless signcryption with known session-specific temporary information security in the standard model. *The Computer Journal*, 63(8):1259–1262, 2020.
- [83] Tarunpreet Bhatia and Verma. Cryptanalysis and improvement of certificateless proxy signcryption scheme for e-prescription system in mobile cloud computing. *Annals of Telecommunications*, 72(9-10):563–576, 2017.
- [84] Han Shen, Jianhua Chen, Jian Shen, and Debiao He. Cryptanalysis of a certificateless aggregate signature scheme with efficient verification. *Security and Communication Networks*, 9(13):2217–2221, 2016.
- [85] Xing Wang and Hai-feng Qian. Attacks against two identity-based signcryption schemes. In *2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing*, volume 1, pages 24–27. IEEE, 2010.
- [86] Waseem Iqbal, Mehreen Afzal, and Farhan Ahmad. An efficient elliptic curve based signcryption scheme for firewalls. In *2013 2nd National Conference on Information Assurance (NCIA)*, pages 67–72. IEEE, 2013.
- [87] Riaz Ullah, Arif Iqbal Umar, Noor ul Amin, et al. Blind signcryption scheme based on elliptic curves. In *2014 Conference on Information Assurance and Cyber Security (CIACS)*, pages 51–54. IEEE, 2014.
- [88] Malik Zia and Rashid Ali. Cryptanalysis and improvement of an elliptic curve based signcryption scheme for firewalls. *PloS one*, 13(12):e0208857, 2018.
- [89] M Zia and Rashid Ali. Cryptanalysis and improvement of blind signcryption scheme based on elliptic curve. *Electronics Letters*, 55(8):457–459, 2019.
- [90] Malik Zia and Rashid Ali. Cryptanalysis and improvement of a blind multi-document signcryption scheme. *Cryptologia*, pages 1–15, 2020.

- [91] Malik Zia and Rashid Ali. A multi recipient aggregate signcryption scheme based on elliptic curve. *Wireless Personal Communications*, 115(2):1465–1480, 2020.
- [92] William Stallings. *Cryptography and network security, 4/E*. Pearson Education India, 2006.
- [93] Henri Cohen, Gerhard Frey, Roberto Avanzi, Christophe Doche, Tanja Lange, Kim Nguyen, and Frederik Vercauteren. *Handbook of elliptic and hyperelliptic curve cryptography*. CRC press, 2005.
- [94] Pierrick Gaudry. Integer factorization and discrete logarithm problems. *Les cours du CIRM*, 4(1):1–20, 2014.
- [95] Henri Cohen, Gerhard Frey, Roberto Avanzi, Christophe Doche, Tanja Lange, Kim Nguyen, and Frederik Vercauteren. *Handbook of elliptic and hyperelliptic curve cryptography*. CRC press, 2005.
- [96] Helmut Hasse. Zur theorie der abstrakten elliptischen funktionenkörper iii. die struktur des meromorphismenrings. die riemannsche vermutung. *Journal für die reine und angewandte Mathematik*, 1936(175):193–208, 1936.
- [97] Jan Pelzl, Thomas J Wollinger, Jorge Guajardo, and Christof Paar. Hyperelliptic curve cryptosystems: Closing the performance gap to elliptic curves (update). *IACR Cryptol. ePrint Arch.*, 2003:26, 2003.
- [98] Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE transactions on Information Theory*, 22(6):644–654, 1976.
- [99] David G Cantor. Computing in the jacobian of a hyperelliptic curve. *Mathematics of computation*, 48(177):95–101, 1987.
- [100] Don Coppersmith. The data encryption standard (des) and its strength against attacks. *IBM journal of research and development*, 38(3):243–250, 1994.

- [101] Joan Daemen and Vincent Rijmen. Aes proposal: Rijndael, aes algorithm submission, september 3, 1999. URL <http://www.nist.gov/CryptoToolKit>, pages 37–38, 1999.
- [102] Elaine Barker, William Barker, William Burr, William Polk, and Miles Smid. Nist special publication 800-57. *NIST Special publication*, 800(57):1–142, 2007.
- [103] Hannes Tschofenig, Manuel Pegourie-Gonnard, and IoT Business Unit. Nist lightweight cryptography workshop 2015 session vii: Implementations & performance.
- [104] Adnan Akhunzada, Mehdi Sookhak, Nor Badrul Anuar, Abdullah Gani, Ejaz Ahmed, Muhammad Shiraz, Steven Furnell, Amir Hayat, and Muhammad Khurram Khan. Man-at-the-end attacks: Analysis, taxonomy, human aspects, motivation and future directions. *Journal of Network and Computer Applications*, 48:44–57, 2015.
- [105] HM Elkamchouchi, Nasr, and Roayat Ismail. A new efficient publicly verifiable signcryption scheme and its multiple recipients variant for firewalls implementation. In *2009 National Radio Science Conference*, pages 1–9. IEEE, 2009.
- [106] Ren-Junn Hwang, Chih-Hua Lai, and Feng-Fu Su. An efficient signcryption scheme with forward secrecy based on elliptic curve. *Applied Mathematics and computation*, 167(2):870–881, 2005.
- [107] Paolo Falcarin, Christian Collberg, Mikhail Atallah, and Mariusz Jakubowski. Guest editors’ introduction: Software protection. *IEEE Software*, 28(2):24–27, 2011.
- [108] Ming Tang, Zhenlong Qiu, Weijie Li, Weijin Sun, Xiaobo Hu, and Huanguo Zhang. Power analysis based reverse engineering on the secret round function of block ciphers. *Concurrency and Computation: Practice and Experience*, 26(8):1531–1545, 2014.

-
- [109] Zifei Shan, Haowen Cao, Jason Lv, Cong Yan, and Annie Liu. Enhancing and identifying cloning attacks in online social networks. In *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication*, pages 1–6, 2013.
 - [110] Gustav Svensson. Auditing the human factor as a part of setting up an information security management system, 2013.
 - [111] Sasikaladevi and Malathi. Privacy preserving light weight authentication protocol (leap) for wban by exploring genus-2 hec. *Multimedia Tools and Applications*, 78(13):18037–18054, 2019.
 - [112] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM conference on Computer and communications security*, pages 62–73, 1993.