Forward Security from Bilinear Pairings:
Signcryption and Threshold Signature

by

CHOW Sze Ming, Sherman

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Computer Science
at
the University of Hong Kong

August 2004

Declaration

I declare that this thesis represents my own work, except where due acknowledgement is made, and that it has not been previously included in a thesis, dissertation or report submitted to this University or to any other institution for a degree, diploma or other qualifications.

Signed _____

CHOW Sze Ming, Sherman

Acknowledgements

I would like to take this opportunity to make my longest acknowledgement in an explicit form in my life so far, for their help in the completion of my MPhil study.

First of all, I would like to express my gratitude to my supervisors, without whom the completion of my master study would not have come to a reality. Thank Dr. Lucas Hui for giving me great flexibility in pursuing the research areas which I am interested in and working at my own pace, which are important for the success of my research. Thank Dr. K.P. Chow for giving me opportunities to be a research assistant in the Center of Information Security and Cryptography, to be his student in a security-related final year project, and to be a research postgraduate. Thank Dr. Yiu Siu Ming for not only teaching me how to work in a research community, but also guiding me in many aspects of life outside research. I am also grateful for his confidence in my aptitude in research and in tutoring the master level security course. Thank Dr. Russell Yiu for giving me support in exploring the area of pairing-based cryptography and the chance to participate in an industrial security-related project.

Thank for my friends who always willing to tea and dine with me, including Go Hiu Wing, Laurel Kong, Pauline Siu, Manfred Ng, Holy Chan, Boris Yiu and Kevin Lam. Without them I will read and write papers with a starving stomach. I would like to thank them in enriching my research life too.

Thank Ivy Tong for helping me to come to the decision of pursuing my MPhil degree and thank Yu Kin Ying for his confidence in my chance of studying in a top university overseas. Thank Richard Lui for his encouragement during my early stage of research, and be my companion in the "Ethical Hacking, Incident Responses and Forensics" course. I also thank for the support of my groupmates including Joe Yau, Eric Chan, Sam Tso, Venus Cheung, Alton Lau and Lydia Woo.

Thank for the help of teaching staff of the course "Database Management Systems", including Dr. Beta Yip, Ho Wai Shing, Eric Lo and Kevin Yip. They helped and taught me to be a tutor.

Thank my ex-roommate Feng Jian Qiang and current roommate Tony Wang, for sharing the office with me at nights.

Thank my secondary school's Mathematics teacher Mrs. Yung, who supported me to self-study Additional Mathematics, by that time I start to study Mathematics seriously.

Thank my friend Douglas Kei, a good partner in the database access control and

encryption project.

Thank Calvin Tang, my classmate in King's College and also my friend.

Thank my parents, who brought me on the earth and this computer world.

This research is supported in part by the Postgraduate Studentships from graduate school, Conference Grant of Committee on Research and Conference Grants, the Areas of Excellence Scheme established under the University Grants Committee of the Hong Kong Special Administrative Region, China (Project No. AoE/E-01/99), a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. HKU/7144/03E), and a grant from the Innovation and Technology Commission of the Hong Kong Special Administrative Region, China (Project No. ITS/170/01). Without these supports, I would not have the chance to experience the atmosphere of international conferences and to meet researchers outside the University of Hong Kong.

Thanks again.

Abstract of Thesis entitled

Forward Security from Bilinear Pairings:
Signcryption and Threshold Signature

submitted by

CHOW Sze Ming, Sherman

for the degree of
Master of Philosophy in Computer Science
at the University of Hong Kong
in August 2004

Given the trend for data to be transferred and transactions to be made on-line, the security of online communications has become a matter of increasing concern. One of the current solutions is public key infrastructure (PKI), in which a trusted authority known as the Certificate Authority (CA) issues a certificate to users. Each certificate contains a public key representing a user in the electronic world and a digital signature by the CA assuring the relationship between the keys and the user. Each user also gets the corresponding private key from the CA, which gives a user the power to digital sign a document and decrypts an encrypted document directed to him.

Despite years of research developing this system, PKI has not been adopted as widely as hoped. One of the major problems is users must first subscribe to PKI in order to receive an encrypted message. Thus has created a "chicken and egg" situation, as potential users are unable to assess the potential value of PKI before subscribing. Many mobile devices have network connectivity nowadays. However, it has so far been difficult to deploy PKI in mobile ad-hoc networks, due to security concerns and potential availability (frequent sudden disconnections). This thesis aimed at providing cryptographic schemes which address these problems.

In ID-based cryptography, the public key can be any string, such as an email address, that can identify the user. This new paradigm is become more popular recently, since it provides a more convenient alternative to PKI. Signcryption scheme is a cryptographic primitive that combines encryption and signing in one step at a lower computational cost, but with higher security than the "sign-then-encrypt" approach. Devising a forward secure ID-based signcryption scheme with public verifiability is an open problem.

Threshold signature schemes lower the chance of key exposure by sharing the key among different entities. They also address the problem of unavailability, in which any subset consisting of a threshold number of shared-key holders can give a valid signature. Combining these properties is particularly useful in an ad-hoc network environment. However, existing schemes are inefficient for ad-hoc networks.

In this study, we utilize bilinear pairing to devise an ID-based signcryption scheme which close the open problem, and a threshold signature scheme which is efficient in key updates and round-optimal in signing. Both schemes are forward-secure, so that the security of the systems will not be completely broken even key exposure occurs.
(400 words)

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Study carried out in this thesis explores cryptographic schemes from bilinear pairings on elliptic curve, which are forward-secure and provide better solutions than the schemes working with traditional public key infrastructure (PKI). The major results are a forward-secure identity-based (ID-based) signcryption scheme and a forward-secure threshold signature scheme.

In traditional PKI, a trusted authority known as the Certificate Authority (CA) issues certificate to users. Each certificate contains a public key representing a user in the electronic world and a digital signature by the CA assuring the relationship between the keys and the user. Can we remove the use of certificate in public key cryptosystems?

In many situations we want to enjoy confidentiality, authenticity and non-repudiation of message simultaneously. Signcryption is a cryptographic primitive that combines encryption and signing in one step. What are the security attributes signcryption schemes should posses? Can we devise a signcryption scheme that can achieve all of the desirable security attributes? Indeed, devising a forward-secure ID-based signcryption scheme with public verifiability was an open problem.

Many mobile devices have network connectivity nowadays. However, it has so far been difficult to deploy PKI in mobile ad-hoc networks (MANET) due to its inherent

dynamic and unpredictable nature. Can we obtain digital certificate under the transient and volatile nature of MANET?

Forward-secure threshold signature schemes can lower the chance and also the disastrous consequence of complete secret exposure. At the same time, they also address the problem of unavailability. However, existing forward-secure threshold signature schemes are not efficient in key updates and require quite a few number of communication rounds among the signers and the signature requester in the signing process. Thus, these schemes are not useful in situation where key update and signing are frequently such as MANET. Can we devise a more efficient forward-secure threshold signature scheme?

This thesis positively answers all the above questions.

1.2 Synopsis

The rest of the thesis is structured in 3 parts.

Part I: This part explains and presents relevant background material. In Chapter 2, we cover various topics in security and cryptography, which include traditional public key cryptography, identity-based cryptography, security issues in mobile ad-hoc network, forward security, signcryption and threshold signature. Chapter 3 firstly covers the frameworks and requirements of forward-secure ID-based signcryption and forward-secure threshold signature, then specific literature surveys on these two classes of schemes are presented.

Part II: This part contains the technical preliminaries that are necessary for understanding the rest of the thesis. In Chapter 4, we give the essences of the formal security analysis in modern cryptography that will be used throughout the thesis, together with two new intractable problems which are the variants of existing well-known problems. Chapter 5 reviews how ID-based encryption, ID-based signature and hierarchical ID-based signature are constructed. This chapter also presents the difference of security notions for schemes in traditional PKI and ID-based schemes, using ID-based signcryp-

tion as an example.

Part III: This part contains the major results of this thesis. Our forward-secure ID-based signcryption scheme is presented in Chapter 6. We analyze our scheme and show our scheme is provably secure in the random oracle model in the same chapter. In Chapter 7 we begin our work on forward-secure threshold signature. Related building blocks will be described before the discussion of our proposed scheme. Security and efficiency analysis of the scheme will also be given in this chapter. Finally, we review the contributions of this thesis and discuss the open problems pertaining to the work carried out in this thesis in the last chapter.

1.3 Chapter Summary

This chapter aimed to provide the motivation and the overall structure of this thesis.

□ End of chapter.

Part I

Background

Chapter 2

Overview

History of *cryptology* is very long. *Cryptology* has two branches: *cryptography* and *cryptanalysis*. Cryptography, which comes from the Greek "kryptos" (secret) and "grapho" (writing), means the art of "secret writing" originally. The Arabs were the first to protect texts by using digits to substitute for letters. We use their word *cipher* (a digit) to refer to the result of secret writing. Cryptanalysis, on the other hand, aimed at looking for ciphers' weaknesses and breaking their security.

The earliest well-known *cryptosystems* (systems for encoding and decoding messages using cryptography), dates back to the first century BC, was devised by Julius Caesar. Cryptography is used to be an *art* for a long time. Most early cryptographers proposed cryptographic algorithms based on their instinct instead of mathematical theory. Claude Shannon's classified article "Communication Theory of Secrecy Systems" [Sha49] turned cryptography to an exact *science* as a division of mathematics. In the history of cryptology up to 1975, encryption (encoding the plaintext into ciphertext) and decryption (the reverse of encryption) algorithm of all the cryptosystems employ the *same key*, this means the principal who encrypts a message and the one who will be receives and decrypts must agree with a key known only by them but no one else beforehand. In 1976, Martin Hellman, a professor at Stanford University, and Whitfield Diffie, a graduate student, introduced the concept of *public key cryptography* [DH76]. The en-

cryption key can be public, while the corresponding key is kept private. Public key cryptography is also known as *asymmetric* cryptography as one cannot derive (mathematically and practically) the private decryption key from the corresponding public key. This concept circulated in the public research community until 1977, when a method for obtaining digital signatures and public-key cryptosystems was proposed in Martin Gardner's column on Mathematical Games in Scientific American [Gar77]. Nowadays this cryptosystem is known as RSA named after its inventors Ron Rivest, Adi Shamir and Len Adleman [RSA78].

Security by obscurity is well-known to be flawed, if not worse than no security at all. Auguste Kerckhoff's principle from the late 1880s [Ker83] states that system designers should assume that the entire design of a security system is known to all attackers, with the exception of cryptographic key secrets. So we do not need to place extra effort to prevent the cryptosystems from reverse engineering for security purpose. A famous example of the danger of *limited* cryptographic algorithm, which the security depends on the secrecy of the implementation details, is A5/1 stream cipher used in GSM. However, the great threat is migrated to the secrecy of the key.

To deal with the *key exposure* problem, there are two main classes of solutions. The first approach is to prevent key exposure altogether; however, it is not always practical. For example, we need some degree of physical security to prevent the devices from physically compromised. A different class of approaches tries to minimize the damage caused by the key exposure, *threshold* cryptography and *forward-secure* cryptography can be viewed as different means of taking this approach.

In threshold cryptography, the secret key is shared among many entities in a "special" way. Cryptographic tasks like decryption and signing can only be done if a threshold number (or more) of active participants help by using their respective share of secret key. The exposure of the secret keys of any group of size smaller than the threshold size does not enable the adversary to complete the cryptographic task.

For cryptographic schemes to be forward-secure, the exposure of the secret key cor-

responding to a given time period does not enable any adversary to "break" the scheme's security (in the sense of the corresponding cryptographic scheme) for any prior time period.

Two fundamental services of cryptography are encryption and digital signature. Encryption provides confidentiality of messages as only the intended recipient can get the original message (plaintext) from the encrypted message (ciphertext), while digital signature provides non-repudiation and authentication of the messages.

In this thesis, we propose two forward-secure cryptographic schemes: namely an *ID-based signcryption* scheme and a *threshold signature* scheme, both of them are extended notion of traditional signature (and also encryption for the signcryption case).

2.1 Traditional Public Key Cryptography

To have secure communication over the Internet, basically an unsecured public network, we need the help of an infrastructure which is known as *public key infrastructure (PKI)*. In PKI, a trusted-by-all party called *Certificate Authority (CA)* provides a *digital certificate* to each individual or organization. Each certificate is associated with a *key pair*: a public key and a private key. The public key represents an individual or an organization in the electronic world, while the corresponding private key gives an entity the power to digitally sign a document and decrypt the encrypted document that is directed to him/her.

Since the public key is usually a "random" string that is unrelated to the identity of the user, certificate also includes a digital signature by CA on a user's public key to assure the relationship between the cryptographic keys and the user. When a user (says Alice) wants to send a message to another user (says Bob), she must obtain an authorized certificate that contains the public key of Bob.

Despite many years of effort (including the recent effort of Hong Kong government), PKI has not been adopted as widely or as quickly as hoped. There are many well documented reasons [Gut02] about the difficulty of deploying the technology by the service

providers (such as maintaining a gigantic online certificate directory) and the difficulty of using the technology by users (such as the strict online requirement and difficulty of locating the certificate). There is also privacy issue associated: the certificate must be accessible to the user of the PKI system and hence a vast amount of information about the certificate entities is made available to the world. Another major problem is users must first subscribe to PKI in order to receive an encrypted message. As potential users are unable to assess the potential value of PKI before subscribing, this has created a "chicken and egg" situation.

2.2 Identity-Based Cryptography

In 1984, Adi Shamir introduced the notion of *identity-based (ID-based) cryptography* to solve the certificate management problem (or the public key distribution problem). The distinguishing property of ID-based cryptography is that a user's public key can be any binary string, such as an email address, that can identify the user. Then a trusted party called *Private Key Generator* (PKG, c.f. CA in traditional PKI) generates the associated private key on user's demand, with the help of PKG's master secret key.

Since the public key can be easily derived, PKG does not need to maintain a list of certificates issued. Each user only need to store the PKG's system parameter instead of a database of certificates of other users, hence ID-based cryptography is supposed to provide a more convenient alternative to the traditional PKI.

ID-based cryptography can be easily extended to support access control policy as well. Since the public key is not some random bits but a human / machine readable string. PKG can simply concatenate the extra condition in key usage with the user's identifier. For example, the user holding the private key of "$user@domain.com||expiry : 08/2004$" can no longer use his/her private key after 08/2004 as the verification processes will return fail if he/she claimed some other expiry date instead like 09/2004.

Shamir suggested a concrete ID-based signature scheme; however, ID-based encryption scheme (IBE) was left as an open question [Sha85]. There have been several con-

structions of IBE afterwards, (for example, bandwidth-inefficient scheme in [Coc01]) but none of these proposals are fully satisfactory until the work of Dan Boneh and Matt Franklin in 2001 [BF01]. They proposed the first practical IBE scheme by utilizing bilinear pairings (which will be described in more details in later chapters). Afterwards, bilinear pairings have been used extensively in the design of ID-based schemes (e.g. [CC02, GS02, ML02, Boy03b, Hes03, LQ03, NR03, LBD+04]) and other cryptographic schemes (e.g. [BLS01, BGLS03, CHK03, DFK+03, HWI03, LQ04]).

2.3 Security Issues in Mobile Ad-Hoc Network

Many mobile devices (e.g., laptops, handheld digital devices, personal digital assistants and wearable computers) have network connectivity nowadays, which give rise to wireless network. With advancement in wireless networks in general, recently we have a new network paradigm: mobile ad-hoc network (MANET). A MANET[1] is an autonomous system of mobile nodes (routers and associated hosts) usually connected by short range wireless channel. These nodes can freely and dynamically self-organize into arbitrary and temporary network topologies, allowing people and devices to seamlessly internetwork in areas without any preexisting communication infrastructure.

Mobile wireless nodes are usually less physically secure (compared with traditional computational devices like desktop computers) and communication channels are subject to eavesdropping due to the open medium nature, so there is a great need to maintain MANET's security. There are many security challenges in MANET [PH03]. For availability, we need to have secure routing [PH02]. A few examples of ad-hoc network applications which need to have confidentiality and authenticity in the communication includes ad-hoc group meeting [AG00], surveillance sensor networks [BHBR01], military battlefield [KaG+02], disaster recovery and emergency operations [VOT04] (where the news reporters may be the eavesdropper).

PKI is again one of the solutions to enable the secure communication within ad-hoc

[1] Actually MANET is somewhat synonymous with Mobile Packet Radio Networking, a term coined via during early military research in the 70's and 80's.[CM99]

networks. However, like the case in wired network, it is also non-trivial to deploy PKI in MANET. Due to its inherent dynamic and unpredictable nature, the idea of single CA is not practical.

2.4 Forward Security

Forward security has different senses in different cryptographic schemes. Basically, the exposure of the secret key corresponding to a given time period does not enable any adversary to "break" the scheme's security for any prior time period.

2.4.1 Forward Secure Signature

The notion of forward security for signature and public key encryption was introduced by Ross Anderson in an invited talk given at ACM Conference on Computer and Communications Security in 1997 [And97].

Mihir Bellare and Sara Miner extended the security definition for ordinary signatures [GMR88] to forward-secure digital signature scheme with two concrete constructions in [BM99]. One is a binary certification tree scheme using any ordinary signature scheme, and the other is transformed from Fiat-Shamir [FS87] ordinary signature scheme. Subsequent constructions followed these two approaches too. The first approach treats ordinary signature schemes (e.g. RSA) as black box, and tries to construct forward-secure signature schemes out of them with random number sequence ([Kra00] generates many certificates in advance in a pseudorandom manner) and different tree constructions ([Kra00] employs Merkle tree while [MMM02] modifies the [BM99]'s tree-based scheme and removes the requirement of fixed number of time periods prior to key generation). The second approach [BM99, AR00, IR01] modifies specific signature scheme using various techniques like *repeated squaring*. These schemes have different trade-offs. Based on [OS91], [AR00] shortens the secret and public keys of [BM99]. Both of [BM99, AR00]'s signing and verification times are linear in the number of time periods, subsequent schemes have faster signing and verification time [IR01] (based on [GQ90], but key generation and update time are linear in the number of time

period supported) or faster update time [KR03] (inspired by [Son01]).

Recently, forward-security is built into other signature schemes except the standard one. For examples, forward-secure threshold signature schemes [AMN01, TT01, CLT03], forward-secure group signature [Son01] and forward-secure blind signature scheme [DCK03]. Unfortunately, the later two constructions were shown to be insecure by [Wan04] and [LC04] respectively.

For the applicability of forward-secure signature, [CJMM03] evaluated the practical performance and the feasibility of deploying forward-secure signature in real world applications, while [GDH+04] gave an analysis of the suitability of using existing forward-secure threshold signature scheme in private keys of MANET's users.

2.4.2 Forward Secure Encryption

Using session keys allows different sessions to be independently secure: even if one session key is compromised, the security of any other session will not be affected. In most session key exchange (or establishment) protocols, long term keys are used to establish session keys. The term "(perfect) forward secrecy" was first appeared in 1989 from the session key exchange protocols proposed by Christoph G. Günther [Gün90]. A protocol is said to provide forward secrecy if the compromise of long term key does not compromise past session keys that have been established before the compromise of the long term key.

In the sense of encryption, forward security means a break-in to the system does not compromise the secrecy of previously-encrypted information. A trivial forward secure public key encryption scheme [CHK03] can be obtained from a forward-secure key-exchange protocol: the sender and the receiver first generate a shared session key K together, then the sender encrypts the message using K, finally both parties promptly erase this shared key [AAB+98]. However, this solution is *interactive*.

The concept of forward security is also defined in private-key cryptography [BY03]: the confidentiality of data that have been encrypted using some secret information in the

past is not compromised by loss of the secret at present.

For ID-based signcryption, we borrowed the definition in [BY03] and adopted it in public-key cryptography, where the secret information we consider here is the private signcryption key of the sender.

Non-interactive solution for forward-secure symmetric-key encryption [BY03] and public-key encryption [CHK03] have been studied previously, but not for ID-based signcryption with public verifiability.

2.4.3 Other Key-Evolving Paradigms

There are other key-evolving paradigms except forward-secure schemes. These paradigms differ in configurations (usually involving more than one entity), adversary settings and security properties.

- *Weak Forward-Security*: In the weak forward-security paradigm, there is an additional entity called security mediator (SEM), which holds a share of the user's private key. The user and the SEM must cooperate to sign on a message or decrypt the ciphertext received. It is weaker than the normal sense of forward-security as the forward-security of the scheme is defined with respect to one party's private key share only, i.e. the compromise of only one of the parties' secret still ensure the forward-security of the scheme, but the compromise of both parties' secret implies a total-break of the cryptosystem.

  This weaker notion of forward-security was proposed in [Tsu03], together with the construction of weak forward-secure signature scheme and encryption scheme enabled by mediated RSA [BDTW01].

- *Strong Forward-Security*: The inherent weakness in forward-security paradigm is that the security of the system is in question after the key compromise, until the public key is revoked. The notion of strong forward-security was introduced in [BCKM01] to address this weakness. Their scheme requires the user sending the

updated public key in each time period to CA to get certification. However, notice that keeping the same public key to be used in all time periods is the basic feature of the original definition of forward-secure schemes.

- *(Strong) Key-Insulation*: There is a special entity called the *base* in this paradigm, which is responsible for updating the user's secret key at the start of each time period. With the help of this additional entity, the security against the key-exposure is somehow stronger than that of forward-secure schemes. Even the adversary adaptively obtained the user's secret keys for t distinct periods, the scheme remains secure in any other periods. The base is assumed to be fully trustable, which is different from the weak forward-security paradigm, in which the compromise of only the SEM does not enable the adversary to break the whole scheme. The notion of strong key-insulated cryptography addressed this weakness, where the base may be untrusted.

  Indeed, the first work [DKXY02a] introducing the concept of key-insulation already gave the definition of strong key-insulated cryptography. There was a refinement of this concept in the later work [DKXY02b], which named this notion as strong key-insulated cryptography and proposed the definition of corresponding normal key-insulation.

  A strong key-insulated public key encryption scheme was proposed in [DKXY02a] while a strong key-insulated signature scheme was proposed in [DKXY02b].

- *Intrusion-Resilience*: This paradigm can be considered as the extension of both of the forward-security and the key-insulation paradigms. The configuration of this paradigm is similar to that of the key-insulation paradigm. If the key-exposures occur alternatively between the signer and the base (i.e. either one of the signer or the base is compromised), the scheme remains secure for all unexposed time periods. On the other hand, if the key-exposures occur at both sides, the scheme remains forward-secure (of course except the period when the user's secret has

already been exposed). Similar to the case of strong key-insulated schemes, key exposure of the base is allowed. The base and the user's secret keys are both forward-secure in this paradigm, but not in the key-insulated paradigm.

The notion of intrusion-resilience was proposed in [IR02] together with a concrete construction of intrusion-resilient signature scheme. General constructions of intrusion-resilient signatures have been studied in [Itk03]. Similar to the case of signature, intrusion-resilient public-key encryption was proposed in [DFK+03] and a general construction was proposed in [DFK+04]. The construction in [DFK+03] made use of the bilinear pairings.

2.5 Signcryption

In many situations we want to enjoy confidentiality, authenticity and non-repudiation of message simultaneously. A traditional approach to achieve this objective is to "sign-then-encrypt" the message, or employing special cryptographic schemes otherwise.

2.5.1 Signcryption and Other Related Notions

An example of the encryption schemes that provide more than confidentiality is *authenticated encryption* (e.g. [HMP94, HMP95, LC95]). Authenticated encryption provides data integrity in addition to the confidentiality provided by normal encryption. However, early authenticated encryption schemes provide no non-repudiation.

In 1997, Yuliang Zheng proposed a novel public key cryptographic primitive that combines encryption and signing in one step at a lower computational cost, which is called *signcryption* [Zhe97, Zhe98]. Zheng claimed that the schemes in [Zhe97] provide both data integrity, confidentiality and non-repudiation. Later work in [Zhe98] proposed a variant of [Zhe97] which supports multiple designated receivers. Indeed, non-repudiation for signcryption is not trivial to achieve, the straight-forward construction only enables the intended recipient to verify the authenticity of the message since the signcrypted message is "encrypted". As pointed out by [PM98], the schemes in [Zhe97], which can be viewed as variants of the general authentic message encryption

in [HMP95], cannot achieve the non-repudiation property: the information given by the intended recipient to settle a dispute compromises the privacy of all other signcrypted messages, i.e. confidentiality is compromised to achieve non-repudiation.

Instead of giving authenticity to encryption, another way is to incorporate encryption features into *signature schemes with message recovery*. Examples of signature schemes with message recovery include the discrete logarithm based signature schemes in [NR95], and RSA signature without using hash function. There were some confusions in the cryptographic community about the properties of digital signature with message recovery and authenticated encryption: [MY99] showed that [Che98] is not a signature with message recovery but an authenticated encryption (which has been confirmed by the author of [Che98] in [Che99]). A distinction is made by [Yeu99] with one concrete construction for both classes of schemes. Signature schemes with message recovery provide data integrity and non-repudiation without revealing the recipient's private key, but not confidentiality, while authenticated encryption schemes provide all of confidentiality, data integrity and non-repudiation if the recipient leaks his/her private key.

In resolving a repudiation dispute, if the surrender of private key is not necessary, then a third party must get some other help from the intended recipient. Some authenticated encryption schemes [HMP94, LC95] simply do not have any repudiation settlement procedure, while some other [Zhe97, PM98, HW99, Zhe98] require interaction to settle a repudiation dispute, which is inconvenient and only a limited number of parties can get convinced.

Non-interactive repudiation dispute settlement is achieved in convertible authenticated encryption schemes [LKP00, WH02] or signcryption scheme with public verifiability [BD98]. The recipient can convert the ciphertext into an ordinary signature that can be verified by every party, which provided flexibility in the verifiability. However, message under signcryption must be revealed to the party who want to do the verification in all these constructions [BD98, LKP00, WH02]. This requirement motivated

the work in [GLZ99], in which public verification can be done without accessing the plaintext. Besides, only short messages are supported by [LKP00].

There are some public verifiable signcryption schemes that are compatible with standardized signature schemes. The first work is [YL02], which is based on Korea Certificate-based Digital Signature Algorithm (KCDSA). Another work [SLS03] is based on Digital Signature Algorithm (DSA) [Nat95], a more widely used standardized signature. Their paper also pointed out that [YL02] leaks a small amount of information about the message.

Since convertible authenticated encryption schemes achieved non-repudiation without the surrender of private key and (possibly expensive) interactive protocol, essentially they can be regarded as signcryption schemes with public verifiability. In the rest of the thesis, we stick to the original definitions and whenever the term "authenticated encryption" is used, we refer to a scheme without non-repudiation property.

2.5.2 Provably Secure Signcryption

All signcryption schemes mentioned previously (e.g. [Zhe97, BD98, PM98, HW99]) were proposed with no rigorous treatment of security (the essence of rigorous treatment of security will be given in Chapter 4). It turned out that some signcryption schemes are actually insecure.

The cryptanalysis and the corresponding fix of [Zhe97] in [PM98] was cryptanalyzed and fixed by [HW99]. Some signcryption schemes with public verifiability (e.g. [BD98, WH02]) are actually inherently insecure under a reasonable definition of confidentiality of signcryption (which is called "semantic security", details will be explained in Chapter 4). Publicly verifiable authenticated encryption in [MC03] was shown in [WLH03] that the third party will reject a valid signature produced by their scheme with non-negligible probability. Moreover, it was shown in [WBMC04] that it is forgeable under the condition that the public key registered by the forger is dependent on the victim's public key. (Note that such attack is practically infeasible in ID-based cryp-

tography since there should be a standard in the composition of string used to represent users.) New publicly verifiable authenticated encryption scheme based on Schnorr signature scheme [Sch91] was proposed in [WBMC04] too.

A new notion called group signcryption was proposed in [KM03], a concrete construction based on distributed schemes [MVN99, MV00] was also presented; however, no formal proof of security was provided. It was later shown to be failing to meet the security requirements of coalition-resistance, traceability and unforgeability by [WDKM04]. Analysis in [WDKM04] also showed that the scheme in [HC03], while being the combination of two cryptographic primitives that were widely believed are secure (the ElGamal encryption algorithm [ElG85] and the Schnorr signature scheme [Sch91]), indeed cannot satisfy the requirements of confidentiality.

There are several literatures studied the formal models and security proofs for signcryption schemes [An01, ADR02, BSZ02, MLM03]. Treatment of authentic encryption in the public key setting was provided in [An01]. The model in [ADR02] did not make non-repudiation a requirement of signcryption and did not aim at making signcryption more efficient than traditional "sign-then-encrypt" approach. A variant of [Zhe97] was proposed and proven secure in [BSZ02], but the non-interactive non-repudiation procedure is complex. It made use of zero-knowledge proof and the only suggested method for this is the one used in the proof of Cook's theorem that boolean satisfiability is NP-complete [Coo71]. No details of practical implementation are given. A signcryption scheme based on the intractability of the integer factorization problem was proposed in [SZ00], the scheme has provable unforgeability of ciphertext but no proof on chosen ciphertext security (will be described in Chapter 4).

The first formal model of non-repudiation for signcryption was developed in [MLM03], accompanied by two provably secure signcryption schemes in their model, one is based on RSA and another is based on discrete logarithm problem. Yet, their model addressed the security in a single-user setting instead of the more realistic multi-user setting.

Up to this point, all schemes mentioned do not support ID-based public keys.

2.5.3 Parallel Signcryption

Another way to provide both confidentiality and non-repudiation simultaneously and efficiently is to perform signing and encryption in parallel. In [ADR02], commitment of a message is encrypted and the corresponding de-commitment is signed, the message can be recovered from the commitment with the de-commitment. Such a mechanism decreases the computation time to signcrypt a message to the maximum time required by the encryption and the signing, but there are some computational overheads in the commitment step. Later construction [PP03] employed the secret sharing technique [Sha79] to perform the commitment step: one of the shares is treated as the commitment while the other is the corresponding de-commitment. Apart from the improvement of the efficiency of the commitment step, the scheme achieved a strong notion of security (chosen ciphertext secure and existentially unforgeable, more details in the next section) from weakly secure encryption and signature scheme. Another signcryption scheme employed secret sharing technique is [AI03], but it is actually a simple concatenation of signature and ciphertext.

Recently [DFJW04] proposed another technique to perform parallel signcryption, which achieve optimal exact security, flexible key management, compatibility with PKCS's [Kal98] standard and other properties; but [LQ04] pointed out that their scheme cannot achieve ciphertext anonymity: the recipient of the message needs to know from whom the signcrypted message emanates, or he/she cannot perform unsigncryption. But whether this is a weakness or a useful property depends on the application domain [GLZ99, Boy03b].

2.6 Threshold Cryptography

Threshold cryptography lowers the chance of complete secret exposure by sharing the secret among different entities. No single entity will get hold of the complete secret. Moreover, they also address the problem of unavailability, because any t out of n entities are sufficient to perform the cryptographic task, where t is the pre-determined threshold

size and n is the number of entities sharing the secret (with $t \leq n$). For a (t, n) threshold decryption scheme, the ciphertext is generated by a single sender as usual but it can be decrypted only if the collaborating subgroup is larger than or equal to t. Any $t - 1$ entities learn no information about the message being encrypted. For a (t, n) threshold signature scheme, a valid signature can only be given by the co-operation of t or more entities, but $t - 1$ or fewer entities cannot generate a valid signature.

These schemes are getting more and more popular due to the increasing prevalence of MANET and pervasive computing applications, where ad-hoc groups are very common [BSS02]. One important application of threshold signature is to address the problem of low security level and low availability of a single CA in MANET. Before the generation of signature, the ad-hoc group need to share a common secret first, which may involve the use of *secret sharing* techniques or *distributed key generation*. Moreover, to cope with perpetual leakage, the concept of *proactive security* was introduced.

2.6.1 Secret Sharing

Secret sharing scheme enables a secret to be kept collectively by a group of participants in a way that only a qualified subgroup can reconstruct the secret. In (t, n) threshold secret sharing, the secret is shared (via *distribution* protocol) among a network of n *participants* (one of them may be a trusted *dealer*); any t of them can recreate the secret easily (via *reconstruction* protocol), but any set of fewer members gain no information about the shared secret (in the sense that all possible values are equally likely).

The concept of threshold secret sharing was proposed in [Sha79] and [Bla79], utilizing Lagrange polynomial interpolation and projective spaces respectively.

After these seminal works, other constructions of threshold secret sharing appeared. A modification of [Bla79] was proposed in [Sim91] based on affine spaces. Construction from congruence class based on Chinese Remainder Theorem was proposed in [AB83], the formal security proof of the scheme was later presented in [QPV02]. A few

schemes based on the technique of multiplicative secret sharing [DCB95, BBDW96, WLXZ00] were also proposed: [DCB95] proposed a multiplicative non-abelian sharing schemes while [BBDW96] worked on an abelian group and provided a better bound on the share size expansion, [WLXZ00] further improved the scheme in [BBDW96] with the notion of multiple perfect hash families.

There are generalization of secret sharing scheme [ISN87, BL90], which enable the secret to be reconstructed from authorized subsets with different cardinalities, instead of groups of the same size. Some work [WWW02] aimed at redistribution of secret from (t, n) access structure to a (t', n') one. Apart from these generalizations, there were many research works that extended the capabilities of secret sharing schemes; for examples, removing the requirement of trusted dealer, (publicly) verifiable secret sharing and proactive secret sharing.

2.6.2 Distributed Key Generation

The original motivation of secret sharing is for enabling the mutually suspicious entities with conflicting interests to cooperate with each other. However, in the basic version of secret sharing (e.g. [Sha79]), participants can neither verify the validity of their shares obtained in the distribution protocol nor verify the validity of their shares constructed by other in the reconstruction protocol. Hence a trusted dealer is assumed (actually, the trusted dealer also knows the secret).

Torben Pryds Pedersen designed the first scheme of distributed key generation (instead of centralized key generation by trusted dealer) in [Ped91], which is a non-interactive scheme assuming the existence of broadcast and private channels. This work is simple and efficient, but it was later proven by [GJKR99] that the key generated is not uniformly distributed in the key space (two malicious members can bias the last bit of the public key with probability of $\frac{3}{4}$ instead of $\frac{1}{2}$). [CGJ+99] further improved the solution proposed in [GJKR99] to withstand adaptive attacks, where the adversary chooses which participants to corrupt at any time and based on any information he sees during

the protocol.

In [JL00], protocols ([CGJ+99, FMY99a, FMY99b]) which are secure against adaptive attacks were found to be insufficient to support cryptosystem secure against adaptive chosen ciphertext attack and signature scheme secure against adaptive chosen message attack (these two notions will be described at Chapter 4). [JL00] proposed two new models of security for adaptive attacks: the first one dealt also with concurrent adversaries whereas the second presents erasure-free adaptive security with persistently inconsistent players.

All previously mentioned schemes (except one of the schemes in [JL00] which uses inefficient non-committing encryption) assumed the existence of private channels. Private channels are usually implemented by the establishment of a secret key between each pair of participants, which involves an extra round (and hence extra cost) before the execution of the actual protocol. Moreover, the use of private channel make it difficult to detect whether the faulty participant is the sender or the receiver. So publicly verifiable encryption scheme (PVE) [CD99] was used by subsequent proposal in [FS01] to detect whether the sender has sent faculty parts in a "private" channel. The result in [FS01] is a one round scheme which generates a discrete logarithm key with public channels only, utilizing publicly verifiable secret sharing (PVSS) scheme and PVE.

Recently, [ZI03] generalized the round-optimal distributed key generation protocol in [FS01] by using any arbitrary homomorphic encryption other than Paillier cryptosystem [Pai99] used in [FS01]. With this generalization, the security of the protocol only relies on a single class of mathematical assumptions instead of involving the composite degree residuosity assumption relied by [Pai99].

2.6.3 Proactive Security

Throughout the lifetime of a threshold cryptosystem, more and more parties may get compromised, or the shares holding may get corrupted or lost. To prevent the secret to be lost forever when there are less than t correct shares remain, an obvious solution

is to reconstruct the secret before this happens. However, the threshold nature of the scheme is destroyed forever. Rafail Ostrovsky and Moti Yung introduced the concept of proactive security for secret sharing in [OY91]: at the beginning of each predefined time period, a share *renewal* protocol is executed so that all compromised or lost shares are regenerated, but the secret stays the same.

Amir Herzberg *et al.* [HJKY95] introduced robustness into this notion by using the verifiable secret sharing (VSS) scheme in [Ped92].

2.6.4 Threshold Signature

Any threshold cryptosystems can be realized by secure multi-party computation, but these multi-party protocols [Yao82, GMW87] were designed to compute a single arithmetic or boolean function, which only provided inefficient constructions of threshold signature scheme. Followed by the first efficient threshold cryptosystem introduced by Yvo Desmedt in [Des88], many threshold schemes were devised, Some are based on standard or well-known signature schemes. To name a few examples, threshold version of Digital Signature Standard (DSS) [Nat00] includes [GJKR96b, CGJ+99, FMY99a, JL00], threshold RSA signatures includes [DF92, DDFY94, GJKR96a, FGY96, BF97, FGMY97, PS98, Rab98, CGJ+99, FMY99a, FMY99b, Gil99, Sho00], (some of these work are not specific to threshold RSA signature: for examples, function sharing, threshold public key cryptosystem, two-party and distributed RSA key generation) while [LHL95] and [LCT03] are the threshold version of ElGamal [ElG85] and Guillou-Quisquater (GQ) [GQ90] signature respectively.

2.7 Chapter Summary

This chapter starts by a very brief history of cryptology. The idea of traditional public key cryptography and its shortcomings are discussed. A new paradigm of public key cryptography which is known as ID-based cryptography is introduced. Security issues in emerging mobile ad-hoc network are discussed briefly. We also gives literature surveys on forward-secure cryptography schemes (which includes signature

and encryption), other key-evolving paradigms (which includes weak forward-security, strong forward-security, key-insulation, strong key-insulation and intrusion-resilience), signcryption schemes with other related notions (which includes authenticated encryption and signature schemes with message recovery.) and threshold signature schemes with related techniques (which includes secret sharing, distributed key generation and proactive security).

□ **End of chapter.**

Chapter 3

Literature Survey

Enough background on the existing works in the area of signcryption and threshold signatures are given in the last chapter, now we move to two more specific literature surveys: ID-based signcryption and forward-secure threshold signature. For the better understanding of these two notions, the frameworks and the requirements of forward-secure ID-based signcryption and forward-secure threshold signature will be given before the respective survey.

3.1 ID-based Signcryption

3.1.1 Framework

An identity-based (ID-based) signcryption scheme consists of five algorithms: `Setup`, `Extract`, `Signcrypt`, `Unsigncrypt` and `TP_Verify` (if public verifiability is satisfied). In essence, `Setup` generates common public parameters and master secret depending on the security level parameter; `Extract` generates the private key(s) for each user according to the user's public identity; `Signcrypt` produces the ciphertext from a sender to a designated recipient; `Unsigncrypt` recovers the original message after checking its integrity and origin; `TP_Verify` enables any third party to verify the integrity and origin of the message. The functions of these algorithms are described as follows.

- `Setup`: On an unary string input 1^k where k is a security parameter, it produces

the common public parameters $params$, which include a description of a finite message space together with a description of a finite ciphertext space; and the master secret s, which is kept secret by the Private Key Generator (PKG)

- `Extract`: On an arbitrary string input ID, it computes the private signcryption key S_{ID} and the private decryption key D_{ID}, corresponding to $(params, s)$. Note that in our framework, the signcryption key and the decryption key are not necessary the same.

- `Signcrypt`: On input (m, S_{ID_A}, ID_B), it outputs a signcrypted ciphertext σ, corresponding to $(params, s)$.

- `Unsigncrypt`: On input (σ, ID_A, D_{ID_B}), it outputs the original message m and ephemeral data $temp$ for public verification (if the scheme provides public verifiability), or the symbol $\perp$ if σ is not accepted as a valid ciphertext, corresponding to $(params, s)$.

- `TP_Verify`: On input $(\sigma, ID_A, m, temp)$, it outputs $\top$ for "true" or $\perp$ for "false", depending on whether σ is a valid ciphertext of message m signcrypted by ID_A or not, corresponding to $(params, s)$.

These algorithms must satisfy the standard consistency constraint of ID-based signcryption, i.e. if $\sigma = \texttt{Signcrypt}(m, S_{ID_A}, ID_B)$, then we must have $(m, temp) = \texttt{Unsigncrypt}(\sigma, ID_A, D_{ID_B})$ and $\top = \texttt{TP\_Verify}(\sigma, ID_A, m, temp)$.

The framework of ID-based signcryption is illustrated in Figure 3.1.

3.1.2 Requirements

An ID-based signcryption scheme should provide the following properties.

1. *Forward Security (FwSec)*: Following the definition from [LQ03] and [BY03], an ID-based signcryption scheme provides forward secure *encryption* if knowing the private key of the *sender* cannot *reveal* the messages he or she signcrypted before.

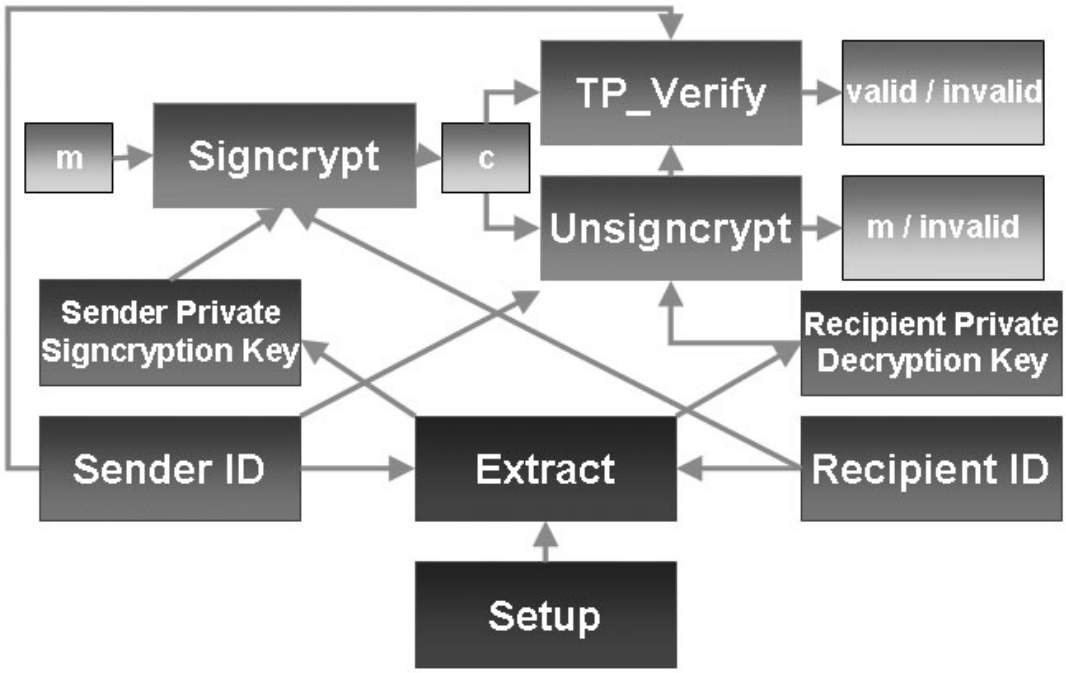

Figure 3.1: *Framework of ID-based Signcryption*

2. *Public Verifiability (PubVer)*: An ID-based signcryption scheme is publicly verifiable if given a message m, a signcrypted message σ, and possibly some additional information provided by the recipient, a third party can verify that σ is a valid signature of the sender for m, without knowing the recipient's private key.

   Note that we adopt a different definition from that in [JJR+03], we refer the public verifiability defined in [JJR+03] as *public ciphertext authenticity*, which we will revisit later.

3. *Provable Security (ProvSec)*: An ID-based signcryption scheme is said to be provably secure if it satisfies the property of *indistinguishability against adaptive chosen-ciphertext-and-identity attacks* (also known as semantical security) and is secure against *an existential forgery for adaptive chosen-message-and-identity attacks*.

   Depending on the applications, an ID-based signcryption scheme may need to sat-

isfy additional requirements. In this thesis, we consider the following additional requirements.

4. *Public Ciphertext Authenticity (PubCAuth)*: An ID-based signcryption scheme is said to provide public ciphertext authenticity if any third party can verify the validity and the origin of the ciphertext without knowing the content of the message and getting any help from the intended recipient.

   This requirement is useful in applications such as authentication of encrypted messages by firewalls [GLZ99] in which the origin of the received signcrypted message must be verified by a third party before the message is accepted in the system.

5. *Recipient Anonymity (ReAnon)*: An ID-based signcryption scheme is said to provide recipient anonymity if any third party cannot learn from the ciphertext that who can unsigncrypt the ciphertext successfully. Again, the recipient anonymity should be preserved even the private key of the *sender* is compromised.

   This requirement is important for scheme with public ciphertext authenticity, or any third party will have the full knowledge of where is the ciphertext originated and who is the intended recipient. For scheme with forward security, the recipient is the only one (except the trusted PKG) that can unsigncrypt a ciphertext. If the adversary can know who is the intended recipient easily, the recipient may face the threat of being compromised or forced to unsigncrypt the ciphertext.

3.1.3 Existing Work

Before our proposed ID-based signcryption scheme, *none* of the previous ID-based signcryption schemes can satisfy all the above requirements. John Charles Malone-Lee gave the first ID-based signcryption scheme [ML02]. His scheme provides forward security and public verifiability. However, the scheme is not semantically secure. As pointed out by [LQ03], this scheme is the result of a combination of a simplified ver-

sion of Boneh and Franklin's ID-based encryption [BF01] with a variant of Florian Hess's ID-based signature [Hes03]. Roughly speaking, the signcrypted message is a concatenation of a signature and a ciphertext. In other words, the signature of the message is visible in the signcrypted message, so the scheme cannot be semantically secure [SLS03]. The scheme proposed by Ryuichi Sakai and Masao Kasahara [SK03] is semantically insecure too. Possibly their scheme only provides forward security and receipt anonymity.

On the other hand, Divya Nalla and K. Chandrasekhar Reddy's ID-based signcryption scheme [NR03] cannot provide public verifiability as well as public ciphertext authenticity since the verifications can only be done with the knowledge of recipient's private key. Benoît Libert and Jean-Jacques Quisquater proposed three ID-based signcryption schemes [LQ03]. None of them can satisfy the requirements for public verifiability and forward security at the same time.

Xavier Boyen's multipurpose ID-based signcryption scheme [Boy03b] is the only existing scheme that provides public verifiability and forward security and is also provably secure. However, this scheme aimed at providing ciphertext unlinkability and anonymity. So, a third party cannot verify the origin of the ciphertext, thus the scheme does not satisfy the requirement of public ciphertext authenticity. We remark that Boyen's scheme is very useful in applications that require unlinkability and anonymity.

3.2 Forward-Secure Threshold Signature

3.2.1 Framework

We adopt a standard framework of a forward-secure threshold signature scheme [AMN01, TT01, CLT03]. For a (n, t) forward-secure threshold signature scheme, there are n signers each with a secret key share, where any t out of n signers can function together to generate signatures. A (n, t) forward-secure threshold signature scheme consists of four components: key generation (`KeyGen`), distributed signing (`Sign`), distributed key evolution (`Update`), and verification (`Verify`). The functions of these al-

gorithms are formalized as below.

- `KeyGen`: On input of the total number of signers n, the threshold number of signers t, the total number of time periods T and an unary string input 1^k where k is a security parameter, it produces the public parameters $params$, which include the public key PK, a description of a finite message space together with a description of a finite signature space. Each signer i also gets $SK_0^{(i)}$ as the share of the secret key value SK_0 for period 0.

- `Sign`: On input of (i, j, m, $SK_j^{(i)}$), where m denotes the message to be signed and $SK_j^{(i)}$ denotes the i^{th} share of the secret key SK_j for period j ($1 \leq j \leq T$), it outputs the partial signature $\sigma^{(i)}$. A third party or any signer are able to construct the final signature σ given the set of t partial signatures $\{\sigma^{(i)}\}$.

- `Update`: On input of (i, j, $SK_j^{(i)}$), where $SK_j^{(i)}$ denotes the i^{th} share of the secret key SK_j for period j ($1 \leq j \leq T$), it outputs $SK_{j+1}^{(i)}$ and deletes $SK_j^{(i)}$. As a result, the system's secret key is implicitly evolved to SK_{j+1}. For the forward-security, we require `Update` to be a *one-way function* (OWF), i.e. it is efficient to calculate the output form a given input, but not the reverse. The key evolution paradigm is illustrated in Figure 3.2, where "Private Key (i)" denotes the private key at the time period i.

- `Verify`: On input (σ, j, PK, m), it outputs $\top$ for "true" or $\perp$ for "false", depending on whether σ is a valid signature of message m signed by the corresponding secret key of PK at j^{th} time period or not.

3.2.2 Requirements

Apart from the standard requirements on the consistency and the unforgeability, a practical forward-secure threshold signature schemes should satisfy the following list of efficiency requirements, especially when used for situations like MANET.

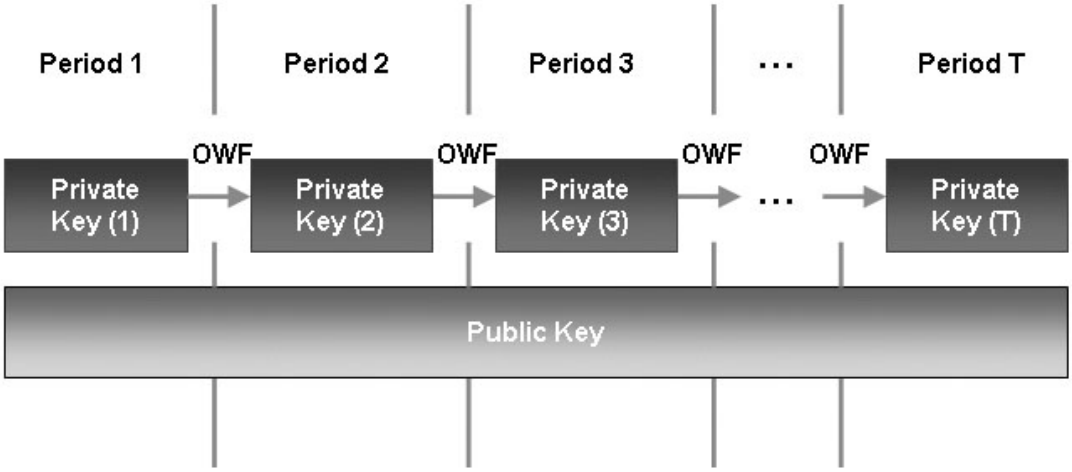

Figure 3.2: *Key Evolution in Forward-Security Paradigm*

1. *Small Signature Size*: In MANET, bandwidth is limited and each packet are probably transmitted via multi-hops, it is desirable to have a small signature size.

2. *Small Storage Size (of Public Key and Private Key)*: Mobile devices typically have a small storage capacity, the scheme's private key size should be kept as small as possible. Since CA(s) needs to keep a repository of certificates (each containing a public key) and each device usually keep a number of certificates for efficiency purpose, the public key size should be small too.

3. *Computationally Efficient Operations (Key Generation, Key Update, Signature Generation and Verification)*: Mobile devices are typically with less computational power, the operations of the scheme should be as efficient as possible, especially for those frequently used operations like key update and signature generation.

4. *Round-Complexity of Operations (Key Generation, Key Update and Signature Generation)*: In MANET, the network condition is volatile and unreliable, the operations of the scheme should requires as few interaction as possible. Optimally, the communication should requires only a single round.

3.2.3 Existing Work

Threshold cryptography has been well studied, however, the results in forward-secure threshold signature schemes are far more limited. There have been several proposals of forward-secure threshold signature schemes.

Michel Abdalla, Sara Miner and Chanathip Namprempre [AMN01] gave a threshold version of [BM99], and proposed two threshold schemes. One is based on multiplicative secret sharing and the other on polynomial secret sharing. The multiplicative scheme introduced requires all n signers to be present to sign messages and perform key update, hence it can only tolerate mobile eavesdropping adversaries and is *not* a truly threshold signature scheme. The polynomial scheme can tolerate mobile halting adversaries and only requires $(2n + 1)/3$ signers to perform signing and key update. However, the signing protocol needs $O(\lg(l))$ communication rounds, where l is the length of the hash output. So this scheme is not useful in situation such as an ad-hoc network.

Wen-Guey Tzeng and Zhi-Jia Tzeng [TT01] gave a threshold version of [AR00], which improves on [BM99] in key sizes. The threshold scheme proposed is both robust and efficient. However, for the polynomial-based construction, the evolution of secret key needs $O(l)$ rounds of communication. In the multiplicative construction, communication round performance is improved but it still requires all n signers to perform signing and key update, (i.e. *not* a truly threshold signature scheme) or at least t out of n signers to reconstruct the secret of the other unavailable signers *explicitly*. As a consequence of reconstruction, the scheme is no longer a threshold one, or the regeneration and redistribution of the new secret must be done after every signing or update operation. This makes the scheme rather *impractical*.

Cheng-Kang Chu and Li-Shan Liu and Wen-Guey Tzeng [CLT03] proposed a better

forward secure threshold signature scheme by integrating [IR01] and [LCT03]. However, the scheme inherits the inefficiency of [IR01]: without any optimization, the cost of key update is $O(T)$ where T is the number of time periods; even optimization techniques in [IR01] were applied, the cost of key update is $O(\log^5 T)$ while secret storage required is increased to $O(\log T)$. Moreover, the multiple uses of VSS protocol make the scheme requires a few number of communication rounds.

3.3 Chapter Summary

In this chapter, we have given the framework of forward-secure identity-based signcryption and forward-secure threshold signature. In particular, we give a new paradigm for identity-based signcryption in which the private signcryption key and the private decryption key are separated.

We have also given a list of requirement for forward-secure identity-based signcryption and a list of requirement for practical forward-secure threshold signature in MANET, which existing schemes cannot satisfy all of them simultaneously.

□ End of chapter.

Part II

Technical Preliminaries

Chapter 4

Provable Security

Rather than defined in a rigorous manner, the security notions for cryptosystems that their designers wished to meet were intuitive at the early stage of cryptography studies. Cryptosystems designed in this way were "insecurity-prone", cryptanalysis often appear after the publication of the designs, modifications were made to prevent specific attacks and later the scheme was found to be still insecure (the scheme is still vulnerable to the attack or another line of attack is opened). It is widely accepted that this approach is doomed to be flawed, which gives rise to a sub-discipline of cryptography: provable security.

Provable security stemmed from the pioneering work of Shafi Goldwasser and Silvio Micali [GM84] in probabilistic public key encryption. The approach is to design the cryptosystem based on some *atomic primitives*: computationally problem that are assumed to be intractable. Security definitions and adversary models are precisely specified to capture what it means for the cryptosystem to be "secure" or what it means to "break" the cryptosystem.

A security proof is constructed via a reduction from the hardness of breaking the underlying atomic primitive to the hardness of breaking the cryptosystem, similar to the way one proves the NP-completeness of a problem by reducing from boolean satisfiability. Such proof assures us the only way to defeat the cryptosystem in the prescribed

model is to break the atomic primitive. The implication is: as long as the atomic primitive is sound (i.e. the underlying problem has no reasonably-efficient solution), the prescribed cryptosystem is secure under the chosen definition of security and adversary model.

Security proof is essential to assert the level of the security a cryptosystem provides, a provably secure cryptosystem provides "fit-for-application" security that simple "textbook cryptography" cannot provide. One example is RSA, possibly the best known cryptosystem. Informally speaking, RSA [RSA78] is insecure [Mao03] under "lunch-time" attack [KY00], an attack aimed at decrypting a certain ciphertext by querying the decryption mechanism for some other predefined ciphertext in a short duration of time says lunch time; but RSA optimal asymmetric encryption padding (RSA-OAEP) [FOPS01] is provably secure against an even stronger mode of attack (ciphertext is not necessarily predefined, it can be prepared after the interaction with the decryption mechanism started).

4.1 Computational Assumptions

Cryptography is the study of human's stupidity (which can be considered as a rephrase of the statement by Adam Young and Moti Yung: "Modern cryptography is made possible by the failures in modern algorithmics." [YY04]). Security of modern cryptosystems are often based on the intractability of some computational problems, e.g. if you can factor, then you can break RSA, but the integer factorization problem is presumed to be intractable.

There are many computational problems in number theory, to name a few, the e^{th} roots problem, the (computational and decisional) composite residuosity problem (e.g. relied by Paillier cryptosystem [Pai99]), the quadratic residuosity problem (e.g. relied by the ID-based encryption scheme in [Coc01]), the Phi-hiding problem [CMS99], etc. In this section, problems related to the security of our proposed schemes are discussed, i.e. problems related to discrete logarithm and bilinear pairing.

4.1.1 Bilinear Pairing

Bilinear pairing (see [BF01] for implementation details) is a mathematical structure that is recently applied extensively in cryptography. It gives rise to many cryptographic schemes that are yet to be (efficiently) constructed using other cryptographic primitives, e.g. aggregate signature [BGLS03] and short signature [BLS01]. One of the most distinguishing cryptographic scheme enabled by bilinear pairing is ID-based encryption [BF01], which solved the open problem proposed by [Sha85] in 1984. Currently, the research of pairing-based cryptosystems still continues at a furious rate. We describe the key properties of bilinear pairing below.

Let $(\mathbb{G}_1, +)$ and $(\mathbb{G}_2, \cdot)$ be two cyclic groups of prime order q. The bilinear pairing is given as $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, which satisfies the following properties:

1. *Bilinearity*: For all $P, Q, R \in \mathbb{G}_1$, $\hat{e}(P + Q, R) = \hat{e}(P, R)\hat{e}(Q, R)$, and $\hat{e}(P, Q + R) = \hat{e}(P, Q)\hat{e}(P, R)$.

2. *Non-Degeneracy*: There exists $P, Q \in \mathbb{G}_1$ such that $\hat{e}(P, Q) \neq 1$.

3. *Computability*: There exists an efficient algorithm to compute $\hat{e}(P, Q)$ $\forall P, Q \in \mathbb{G}_1$.

4.1.2 Discrete Logarithm Problems

There are many famous discrete logarithm based cryptosystems, like ElGamal encryption and signature [ElG85], Schnorr signature [Sch91], Cramer-Shoup encryption [CS98], Pointcheval-Stern signature [PS00] and Digital Signature Algorithm (DSA) [Nat00].

Definition 1. *Let p be a prime such that $p - 1$ has a large prime divisor, given a generator g of a group $\mathbb{Z}_p^*$ and a value $g^a \in \mathbb{Z}_p^*$ where $a < p-1$, the Discrete Logarithm problem is to compute a.*

In this thesis, we consider the variant of discrete logarithm problem which is known as elliptic curve discrete logarithm (ECDLP), since all the current admissible pairings (Weil pairing and Tate pairing [BF01, Jou02, GHS02]) are realized on elliptic curve.

Definition 2. *Let E be an elliptic curve defined over a finite field $K = \mathbb{F}_{2^N}$. Given a generator $P \in E(K)$ where the order of P is q (i.e. $\mathrm{ord}(P) = q$), and Q generated by P (i.e. $Q \in < P >$), the Elliptic Curve Discrete Logarithm Problem (ECDLP) is to find the integer $a \in \{0, q-1\}$ such that $Q = aP$.*

4.1.3 Diffie-Hellman Problems

In the seminal paper of public key cryptography [DH76], the first protocol that allows two entities to establish a session key over an untrusted network with passive eavesdroppers is proposed, which is now known as Diffie-Hellman key exchange. Diffie-Hellman key exchange is based on the assumed intractability of solving the following problem: the computational Diffie-Hellman problem.

Definition 3. *Given a generator P of a group $\mathbb{G}$ and a 2-tuple $(aP, bP) \in \mathbb{G}^2$, the Computational Diffie-Hellman problem (CDH problem) is to compute abP.*

Some cryptosystems' security (e.g. Cramer-Shoup cryptosystem [CS98]) is based on the intractability of the decisional variant of the CDH problem.

Definition 4. *Given a generator P of a group $\mathbb{G}$ and a 3-tuple $(aP, bP, cP) \in \mathbb{G}^3$, the Decisional Diffie-Hellman problem (DDH problem) is to decide whether $c = ab$.*

There is a "gap" between computational problems and decisional problems. The gap problems can be considered as a dual to the class of the decisional problems [OP01]. Informally, a gap problem is to solve the computational problem with the help of the an oracle that can solve the related decisional problem.

Definition 5. *The Gap Diffie-Hellman problem (GDH problem) is to solve the CDH problem in a group $\mathbb{G}$ with the help of the oracle that solves the DDH problem in $\mathbb{G}$.*

4.1.4 Bilinear Diffie-Hellman Problems

The existence of bilinear pairings leads to the definition of the following problems:

Definition 6. *Given two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of the same prime order q, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator P of $\mathbb{G}_1$, the Decisional Bilinear Diffie-Hellman problem (DBDHP) in $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$ is to decide whether $h = \hat{e}(P, P)^{abc}$ given $(P, aP, bP, cP) \in {\mathbb{G}_1}^4$ and an element $h \in \mathbb{G}_2$.*

Definition 7. *Given two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of the same prime order q, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator P of $\mathbb{G}_1$, the Computational Bilinear Diffie-Hellman problem (CBDHP) in $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$ is to compute $h = \hat{e}(P, P)^{abc}$ given $(P, aP, bP, cP) \in {\mathbb{G}_1}^4$.*

4.1.5 Modified Bilinear Diffie-Hellman Problems

Both DBDHP and CBDHP are assumed to be hard and no known algorithm can solve any of them efficiently. In this thesis, we consider variants of DBDHP and CBDHP, in which $c^{-1}P$ is also given as input. We refer these variants as MDBDHP (Modified DBDHP) and MCBDHP (Modified CBDHP).

Definition 8. *Given two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of the same prime order q, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator P of $\mathbb{G}_1$, the Modified Decisional Bilinear Diffie-Hellman problem (MDBDHP) in $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$ is to decide whether $h = \hat{e}(P, P)^{abc}$ given $(P, aP, bP, cP, c^{-1}P) \in {\mathbb{G}_1}^5$ and an element $h \in \mathbb{G}_2$.*

Definition 9. *Given two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of the same prime order q, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator P of $\mathbb{G}_1$, the Modified Computational Bilinear Diffie-Hellman problem (MCBDHP) in $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$ is to compute $h = \hat{e}(P, P)^{abc}$ given $(P, aP, bP, cP, c^{-1}P) \in {\mathbb{G}_1}^5$.*

Obviously DBDHP and CBDHP are harder than MDBDHP and MCBDHP, respectively. However, no known existing efficient algorithm can solve MDBDHP and MCBDHP, to the best of our knowledge. Indeed, our actual scheme only needs to publish (P, aP, bP, cP). In [BDZ03], it is shown that inverse computational Diffie-Hellman problem (On input P and cP, outputs $c^{-1}P$) is equivalent to computational Diffie-Hellman problem. So it is computationally infeasible to derive $c^{-1}P$ from the public

system parameters.

We believe that MDBDHP and MCBDHP are interesting in their own rights as their intractabilities may give raise to new cryptosystems like our proposed scheme, in which the private signcryption key and the private decryption key are separated.

4.2 Formal Security Notions

In the realm of provable security, we need to define the *adversarial goals*, schemes only secure against a less ambitious goal is of less security level. We will describe different notions of security of encryption and signature one by one.

On the other hand, what powers an adversary may have, i.e. the *adversarial models*, must be defined clearly too. A trivial method for breaking the security of most cryptographic schemes is to try every possible key from the keys pace. A secure cryptographic scheme should not be breakable in a way more effectively than the above *brute-force attack*. We consider other types of cryptanalytic attacks below, ordered from the weakest threat model to the strongest threat model.

4.2.1 Adversarial Goals in Attacking Encryption Schemes

The most fundamental security property of an encryption scheme is confidentiality.

- *All-or-Nothing Secrecy*: For a given plaintext-ciphertext pair under an encryption algorithm, the adversary either reveal the whole private decryption key, or nothing. And for a given ciphertext from an encryption algorithm, the adversary either determines the whole plaintext block correctly, or nothing. The meaning of "nothing" is that the adversary does not gain any other knowledge before or after its attempt to attack.

  This is the weakest notion of confidentiality. In real applications, we may have some *apriori information* which is known by the attacker. Examples include predefined format of electronic file (e.g. "%PDF" in Adobe PDF file, "%!PS" in PostScript file), small domain (which are small in comparison with the domain

size in cryptographic sense, e.g. key spaces) of values in the file (e.g. the decision of buy/sell, the user's password). These information may assist in achieving the adversary's goal. Most "textbook encryption algorithm" can achieve this notion of security and it was simply considered as "notion of insecurity" in [Mao03].

- *Indistinguishability*: In contrast to all-or-nothing secrecy, this notion of confidentiality captures the impossibility of extracting any information from a ciphertext about the plaintext.

  This notion of confidentiality was proposed together with the probabilistic encryption scheme in [GM84]. Note that the essential property of the probabilistic algorithm is that even the same plaintext is encrypted under an encryption key twice, the two resultant ciphertexts will be different with an overwhelming probability. Probabilistic encryption algorithm challenges the adversary by the following "game", which also defines what is meant by indistinguishability. At the start of the game, the adversary first prepares two distinct message of equal length and sends them to the encryption oracle, then the challenger tosses a fair coin and encrypts either one of the plaintexts according to the face of the coin appeared. The resultant ciphertext is then presented to the adversary. If the adversary is unable to guess the face of the coin seen by the challenger (i.e. which plaintext is encrypted) with probability significantly greater than $\frac{1}{2}$, then the encryption scheme is considered to be indistinguishable.

  This security notion is named as *semantic security* in [GM84]: whatever is efficiently computable about the plaintext given the ciphertext, is also efficiently computable without the ciphertext [Mao03]. This level of confidentiality is indeed essential, since many message contain certain "non-secret partial information". For example, consider the ciphertext encrypting the name of the candidate/option chosen in an e-voting event, there is no need for the adversary to decrypt the ciphertext if the adversary is fully capable to distinguish the ciphertexts. Most di-

rect applications of one-way trapdoor function (e.g. RSA) are very weak in hiding such kind of semantic information [GM84].

- *Non-Malleability*: Indistinguishability sounds secure, but it only guarantees the security of the scheme against a passive adversary, i.e. the adversary only possesses the power to eavesdrop the ciphertext but not modifying it. Non-malleability [DDN91] lifted the security notion of encryption scheme by ensuring the integrity of the ciphertext in the sense that the corresponding plaintext is modified in a manner controlled by the adversary.

  Consider the use of encryption in sending the value of a bid to another party, if the encryption scheme is malleable, then the adversary gain advantage by modifying others' bid to an unreasonable value.

  Indistinguishability and non-malleability capture different requirements of encryption scheme, however, they are indeed related to each other. It has been shown in [BDPR98] that non-malleability implies indistinguishability and indistinguishability implies non-malleability under adaptive chosen-ciphertext attack (CCA2), an attack model which we will cover shortly. We only need to design an encryption scheme that is indistinguishable under CCA2 to achieve non-malleability under CCA2.

4.2.2 Attacks on Encryption

Assuming both of the encryption and decryption algorithm are publicly known, the adversary power is characterized by the resource he/she can access in the attack, i.e. the information he/she holds and the ways he/she can interactive with the encryption and decryption oracles in the case of encryption.

- *Ciphertext-Only Attack*: The adversary is only given some ciphertexts, and the adversary's goal is to reveal information of one or more plaintext from these ciphertexts. Encryption scheme succumbed to this type of attack is simply considered

as insecure.

- *Known-Plaintext Attack*: In this attack the adversary is given a few examples of plaintexts and the corresponding ciphertexts, but this set cannot be chosen by the adversary. It may be assumed that all ciphertexts were produced using the same key and the adversary knew this fact too. The goal of this attack is to reveal information given by the decryption result of one or more ciphertexts that the adversary has not yet seen.

- *(Non-Adaptive) Chosen-Plaintext Attack*: The adversary can ask the encryption oracle to encrypt some chosen plaintexts and see the encryption result. However, these plaintexts must be submitted in one single batch. The adversary cannot query for the encryption oracle twice and choose what plaintext to be encrypted in response to the encryption oracle. Again it may be assumed that all ciphertexts were produced using the same key and the adversary knew this fact too. The goal of this attack is to determine information given by the decryption result(s) of one or more ciphertexts that is/are not included any encryption request.

- *Adaptive Chosen-Plaintext Attack*: This threat model is the enhanced version of chosen-plaintext attack, in which the adversary is permitted to present the encryption requests adaptively, i.e. the adversary can formulate the plaintext to submit after obtained some previous queries' result.

  The probabilistic encryption scheme in the pioneering work in provable security [GM84] was shown to be secure against this class of attack.

  The above two notions are mainly for assessing the security of private key encryption schemes. All adversaries have the power to mount chosen-plaintext attack on any public key encryption scheme.

- *(Non-Adaptive) Chosen-Ciphertext Attack*: This threat model is similar to known-plaintext attack, but decryption oracle is given in addition to the encryption oracle.

Notice that only temporary access of the decryption oracle is given to the adversary, and not the secret key.

This attack is also known as a "lunch-time" attack: the scenario that an employee probes the decryption device when left alone in the office during a short period of time such as lunch-time [KY00, Mao03].

The first public-key encryption scheme that is provable secure against this class of attack is [NY90], based on the quadratic residuosity intractability.

- *Adaptive Chosen-Ciphertext Attack*: As pointed out by [RS92], "lunch-time" attack is bounded by an artificial constraint, hence they proposed a stronger notion: the adversary can adaptively choose the ciphertexts to be decrypted based on previously received plaintexts.

  An encryption scheme based on the non-interactive zero-knowledge proof of knowledge was proposed in [RS92] which is secure against adaptive chosen-ciphertext attack. Several email encryption protocols were shown to be insecure against this class of attack in [KS00].

4.2.3 Adversarial Goals in Attacking Signature Schemes

For a signature scheme to be secure, it should be unforgeable. Just like the case of confidentiality in encryption, there are various levels of unforgeability in signature schemes.

- *Total-Break*: The adversary can get the private key of another user (and hence universal forgery is also possible).

- *Universal Forgery*: The adversary can forge signatures on messages of his/her choice.

- *Selective Forgery*: The adversary can forge a signature on a particular class of messages.

- *Existential Forgery*: The adversary can forge a signature for at least one message, but the choice of the message is not in control, and hence the message whose signature is obtained may be random-looking or nonsensical.

The severities of these forgeries are in descending order, e.g. an existentially unforgeable signature scheme is stronger than a selectively unforgeable scheme.

4.2.4 Attacks on Signature

The attacks of signature scheme are somewhat different from those of encryption. In encryption scheme, there are two oracle services provided: encryption and decryption (although the encryption oracle is only useful in the symmetric setting). In signature scheme the verification algorithm is inherently accessible to all parties (including adversary), so in encryption's scenario there are message attacks and ciphertext attacks but only message attacks for signature schemes.

- *Key-Only Attack*: The only thing that is given access to the adversary is the public verification key.

- *Known-Message Attack*: Message attack is an attack in which the adversary is given access to the signatures of messages created using one's private signing key. In a known-message attack, the adversary is given a few examples of messages and their corresponding signatures, but these messages are not chosen by the adversary. It may be assumed that the signing oracle always uses the same key and the adversary knew this fact too.

- *Generic Chosen-Message Attack*: In this attack, the adversary can ask for the signing oracle to sign a set of messages prepared in a single batch, before knowing the public key of the user under attack. The attack is generic since the set of messages prepared are independent on the public key of the user under attack.

- *Directed Chosen-Message Attack*: It is the variation of generic chosen-message attack, in which the message to be submitted the signing oracle is prepared after

the knowledge of the public key of the user under attack, i.e. the attack is directed at a particular user.

- *Adaptive Chosen-Message Attack*: It is the strongest form of attacks for signature schemes, where the adversary is able to request for the signatures of adaptively chosen messages, based on the signatures that are obtained from the signing oracle.

4.3 Random Oracle Model

Random oracle is a popular tool for proving the security of cryptosystems. The model for such a security proof is called *random oracle model* (ROM) [BR93]. In reality, the random oracle is usually instantiated by cryptographic hash function, which can emulate the imaginary random oracle's behaviour to a certain degree.

4.3.1 Cryptographic Hash Function

A *hash function* is a deterministic function which maps a string of arbitrary length to a string of fixed length called the *hashed value*. Hash function is used in many areas of computer science, e.g. a data structure called hash table utilizing the hash function is used in many algorithms. Similar to hash function, cryptographic hash functions are used universally in cryptography: digital signatures (e.g. [Bel00]), public-key cryptosystems with fit-for-application security (e.g. [BR95]), and the random sequence generators used in key agreement, authentication protocols (see [Boy03a]), non-interactive proof of knowledge protocols (e.g. Fiat-Shamir heuristic in [FS87]), and e-commerce protocols like micro-payment aggregation via gambling (e.g. [Whe97, MR01]).

A hash function is necessarily many-to-one due to the pigeon principle, as the domain of the hash function is larger than its range. Collisions in hash function is possible: consider two messages m_0 and m_1 which are two arbitrary elements of the domain of a hash function $H(\cdot)$, $\exists m, m' \; s.t. \; H(m) = H(m'), m \neq m'$. The existence of collision requires cryptographic hash function to satisfy the following additional properties.

1. *Mixing-Transformation*: on any input m, the output hashed value $H(m)$ is computationally indistinguishable from a uniform binary string in the interval $[0, 2^{|H|})$, where $|H|$ denotes the output length of H.

2. *Preimage Resistant (One-Way, or Hard to Invert)*: given y from the range H, it is hard to find m such that $H(m) = y$.

3. *Second-Preimage Resistant*: given m from the domain of H, it is hard to find $m' \neq m$ such that $H(m) = H(m')$.

4. *Collision Resistant (or Collision-Free)*: it is hard to find a pair of distinct messages m, m' such that $H(m) = H(m')$.

A cryptographic hash function should also have practical efficiency like a normal hash function. In our proposed schemes, cryptographic hash function is used. More inquisitive readers may find [Pre99] for various definitions of hash functions, some generic constructions and attacks of hash functions.

4.3.2 Random Oracle Paradigm

Random oracle is a complete idealization of the functionality of a hash function. Similar to the hash function, random oracle also gives deterministic output; but the output of random oracle is uniform in the output space, while the mixing-transformation property of a hash function only requires the output of the function to be *computationally indistinguishable* from the uniform distribution in the range of the function. Below is the formal definition of a random oracle.

Definition 10. *A random oracle R is a function from $\{0,1\}^*$ to $\{0,1\}^\infty$ such that for a given query s to R, each and every output bit of $R(s)$ is chosen uniformly at random and independent of every bit in s.*

There exists no computing mechanism that can provide the functionalities of random oracle in all of the existing computational models. Actually, the random oracle 's properties of determinism and uniform output implies the entropy of its output is greater than

that of its input, which is contradictory to Shannon's entropy theory [Sha48]. Hence random oracle is just a theoretical construction but does not exist in reality. (Please refer to [Mao03] for a more detailed descriptions of Shannon's entropy theory and computational indistinguishability.)

Despite of the disparity between the theory and reality, Mihir Bellare and Philip Rogaway made use of the random oracle for proving the security of cryptosystems [BR93]. The security proof in random oracle model (ROM) is to assume all parties including adversaries have access to a random oracle simulated by a special simulator; then prove the cryptosystem is secure in this model, and finally replace the random oracle with a cryptographic hash function, e.g. SHA [Uni95, Uni01], in the actual implementation of the cryptosystem.

The random oracle is simulated by returning a new randomly chosen value except for repeat queries, in which the same response is returned as the query result. By this simulation the simulated oracle has the properties such as preimage resistance and collision resistance of a hash function.

Due to the good approximation of the random oracle behavior from cryptographic hash functions, it is reasoned that if any weaknesses in this actual implementation of cryptographic scheme is found, it must come from the weakness in the hash function used to instantiate the random oracle, and not from the weakness in the cryptographic design.

The choice of cryptographic hash function is important. Bellare and Rogaway warned that using MD5 [Riv92] in a simple way is not a suitable replacement for the random oracle. In [CGH04], Ran Canetti *et. al.* showed a failure of random oracle methodology by presenting a cryptosystem that is provably secure in ROM but totally insecure when the random oracle is instantiated by any hash function. However, their construction is rather artificial. So proving the security of cryptosystems in ROM is still considered to be a good engineering principle. Indeed, schemes analysed in this model also enjoyed widespread acceptance with standards bodies, e.g. RSA-OAEP [BR95]

and probabilistic signature scheme (PSS) [BR96] with RSA.

4.3.3 Forking Lemma

The use of forking lemma [PS00] to prove the unforgeability of signature schemes is very popular in recent years. This lemma is used for proving the security of a class of signature schemes that covers many signature schemes, (e.g. Fiat-Shamir signature [FS87] and Schnorr signature [Sch91]), in the random oracle model.

The forking lemma is applicable to signature that makes use of the hash function and produces signature in the form of a triplet (h, σ_1, σ_2): suppose the hash function is $H(\cdot)$, h is obtained from hashing the message m together with part of the signature σ_1, i.e. $h = H(m, \sigma_1)$, while the remaining part of the signature σ_2 is dependent on all of σ_1, the message m and the hash value h. Moreover, each invocation of the signing algorithm gives a new signature that is independent of any other signature produced, even the signature is made on the same message (like probabilistic encryption scheme).

Recall that a security proof shows a reduction from the breaking of a certain difficult problem to the breaking of the cryptosystem. The reduction is usually obtained from the game played between two parties: the challenger and the forger, in which the challenger simulated the cryptosystem and interacts with the forger to solve the underlying hard problem. The challenger is given a random instance of the difficult problem, then it embeds this instance of problem to the simulation of the hash function (modeled as the random oracle), the key generation and the signing oracle of the scheme. If the forger can launch an successful attack on this simulation of the scheme, then the challenger can solve the given instance of hard problem.

Suppose the forgery made is (h, σ_1, σ_2), there is only a negligible probability that the signature will verify but the forger did not make any hash oracle query of (m, σ_1). Such a query is called the *critical query*. For the challenger to solve the hard problem, the forking lemma uses the "oracle replay attack". The forger algorithm is executed at least twice, but each time it is interacted with a different simulation of the random

oracle. The simulations of the random oracle are the same until the forger present the critical query. At this point a new random response is given to the adversary, which causes a *fork* in the execution of the adversary.

The forger's algorithm is able to output a valid forgery as long as the random oracle answers are of the correct distribution. The forger does not have any additional functionality like detecting different responses to the same query were returned by the challenger in multiple runs. Even an apparently different random oracle is used to answer the forgery's hash query, the forger will still give a valid forgery on the same message m, just like the case in the first forgery. Suppose the forgery made in the oracle replay is $(h' = H'(m, \sigma_1), \sigma_1, \sigma_2')$ where $h' \neq h$ (by construction) and $\sigma_2' \neq \sigma_2$ (since σ_2 is dependent on h), then the adversary can probably use these two valid forgeries to solve the underlying problem. This point will become more clear in the proof of the existential unforgeability of our ID-based signcryption scheme.

4.4 Chapter Summary

In the formal security analysis of modern cryptography, we need to describe the intractable computational problems the scheme based on, define the framework of the scheme, state the adversarial goal, specify the capabilities of the adversary, and most importantly, an formal reductionist security proof in the model assumed. All of these have been gone through in this chapter.

We have also described the properties of the important underlying primitive of our proposed schemes: bilinear pairings. The basic idea of the use of forking lemma in proving the security of the scheme has also been discussed.

Furthermore, we have proposed two new intractable problems which are the variants of existing well-known problems.

□ End of chapter.

Chapter 5

ID-Based Cryptosystems

Motivated by the certificate management problem encountered by traditional PKI, ID-based cryptosystem are devised. In this chapter, we first see how ID-based encryption scheme and ID-based signature scheme are constructed. This also helps the discussion of the traditional "sign-then-encrypt" and "encrypt-then-sign" approaches in Chapter 6. ID-based cryptographic schemes can be used to derive other cryptographic primitives as well. We review the construction of a hierarchical ID-based signature scheme, which is an important building block for our scheme in Chapter 7.

5.1 ID-Based Encryption

We first see how ID-based encryption (IBE) scheme is constructed.

5.1.1 High Level Idea

In Boneh and Franklin's IBE scheme [BF01], to encrypt a message, the sender uses the bilinear pairing to combine the identity of the receiver, the private key generator (PKG)'s public key and a random short term private key into a session key used to mask the message. The receiver can recreate the same session key by using bilinear pairing to combine his private key and the short term public key sent with the ciphertext.

The Weil and Tate pairing on elliptic curves are the only known ways to build secure bilinear pairing ([Jou02]). The bilinear pairing referred below can be substituted by any

one of them (with a certain modification as stated in [BF01]).

5.1.2 BasicIdent

To explain the basic ideas underlying IBE we describe the following simple scheme, called `BasicIdent`. We present the scheme by describing the four algorithms, `Setup`, `Extract`, `Encrypt` and `Decrypt`. In summary, `Setup` is executed by PKG and generate publicly distributed system parameters and a master key, `Extract` extracts private keys corresponding to a given ID (an arbitrary string), `Encrypt` encrypts a message using a given ID, and `Decrypt` decrypts a ciphertext given a private key. We let k be the security parameter given to the setup algorithm and $\mathcal{IG}$ be some randomized BDH parameter generator which runs in time polynomial in k.

`Setup:` Given a security parameter k,

1. Run $\mathcal{IG}$ on input k to generate cyclic groups $\mathbb{G}_1$, $\mathbb{G}_2$ of prime order q together with a bilinear pairing $\hat{\mathrm{e}}$: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ corresponding to this security parameter (say q could be a k-bit prime). Pick a random generator $P \in \mathbb{G}_1$.

2. Pick a random $s \in \mathbb{Z}_p^*$ and compute $P_{pub} = sP$.

3. Pick cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1^*$, $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^n$.

The plaintext space is $\mathcal{M} = \{0,1\}^n$ and the ciphertext space is $\mathcal{C} = \mathbb{G}_1^* \times \{0,1\}^n$. The master-key is s. The public system parameters are

$$\texttt{params} = < \mathbb{G}_1, \mathbb{G}_2, \hat{e}(\cdot,\cdot), p, n, P, P_{pub}, H_1(\cdot), H_2(\cdot) > .$$

`Extract:` Given a string ID $\in \{0,1\}^*$, the master-key s and system parameters `params`, Compute $Q_{\mathrm{ID}} = H_1(\mathrm{ID}) \in \mathbb{G}_1^*$ and $d_{\mathrm{ID}} = sQ_{\mathrm{ID}}$ and return d_{ID}.

`Encrypt:` Given a plaintext $m \in \mathcal{M}$, a public key ID and public parameters `params`,

1. Compute $Q_{\mathrm{ID}} = H_1(\mathrm{ID})$,

2. Pick a random $r \in \mathrm{Z}_p^*$

   (random short term private key)

3. Compute $g = \hat{e}(P_{pub}, Q_{\mathrm{ID}})$,

4. Set the ciphertext to $C =< rP, m \oplus H_2(g^r) >$.

   (rP can be viewed as the short term public key corresponding to r and g^r can be viewed as the session key)

`Decrypt:` Given a ciphertext $< U, V > \in \mathcal{C}$, a private key d_{ID} and system parameters `params`,

1. Compute $g' = \hat{e}(U, d_{\mathrm{ID}})$,

2. Compute $m = V \oplus H_2(g')$,

5.1.3 IBE with Chosen Ciphertext Security

Boneh and Franklin's paper used a technique due to Fujisaki and Okamoto [FO99] (hereafter we refer this technique as "FO's conversion") to convert the `BasicIdent` scheme of the previous section into a chosen ciphertext secure IBE system in the random oracle model. Here is the description of the scheme in full details:

`Setup:` As in the `BasicIdent` scheme. In addition, pick a hash function $G_1 : \{0,1\}^n \times \{0,1\}^n \rightarrow \mathbb{Z}_p^*$, and a hash function $G_2 : \{0,1\}^n \rightarrow \{0,1\}^n$ for some integer $n > 0$. These two hash functions are required by FO's conversion.
The plaintext space is still $\mathcal{M} = \{0,1\}^n$ but the ciphertext space becomes $\mathcal{C} = \mathbb{G}_1^* \times \{0,1\}^n \times \{0,1\}^n$. The public system parameters now becomes

$$\texttt{params} =< \mathbb{G}_1, \mathbb{G}_2, \hat{e}(\cdot,\cdot), p, n, P, P_{pub}, H_1(\cdot), H_2(\cdot), G_1(\cdot), G_2(\cdot) > .$$

`Extract:` As in the `BasicIdent` scheme.

`Encrypt:` Given a plaintext $m \in \mathcal{M}$, a public key ID and public parameters `params`,

1. Compute $Q_{\mathrm{ID}} = H_1(\mathrm{ID})$,
2. Pick a random $\sigma \in \{0,1\}^n$ and compute $r = G_1(\sigma, m)$
   (a step in FO's conversion)
3. Compute $g = \hat{e}(P_{pub}, Q_{\mathrm{ID}})$,
4. Set the ciphertext to $C = < rP, \sigma \oplus H_2(g^r), m \oplus G_2(\sigma) >$
   (another step in FO's conversion)

`Decrypt:` Given a ciphertext $< U, V, W > \in \mathcal{C}$, a private key d_{ID} and system parameters `params`,

1. Compute $g' = \hat{e}(U, d_{\mathrm{ID}})$,
2. Compute $\sigma = V \oplus H_2(g')$
   (asymmetric decryption in FO's conversion)
3. Compute $m = W \oplus G_2(\sigma)$
   (symmetric decryption in FO's conversion)
4. Compute $r = G_1(\sigma, m)$. If $U \neq rP$, reject the ciphertext, else return m
   (testing procedure in FO's conversion)

We refer the reader to [BF01] for the formal security proof of the above construction.

5.2 ID-Based Signature

An ID-based signature scheme consists of four algorithms: `Setup`, `Extract`, `Sign`, and `Verify`. `Setup` and `Extract` are executed by the PKGs. Based on the security level parameter, `Setup` is executed to generate the master secret and common public parameters. `Extract` is used to generate the private key for any given identity. The

algorithm `Sign` is used to produce the signature of a signer on a message; `Verify` is used by any party to verify the signature of a message.

These algorithms must satisfy the standard consistency constraint of ID-based signature, i.e. if $\sigma = \texttt{Sign}(m, S_{ID})$, then we must have $\top = \texttt{Verify}(\sigma, m, ID)$.

Below are the construction of IBS from [Hes03] (which is scheme 3 of [Hes02], notice that scheme 4 of [Hes02] is shown to be universal forgeable by known-message attack in [Che02]).

Let H_1 and H_3 be two cryptographic hash functions where $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$ and $H_3 : \{0,1\}^n \times \mathbb{G}_2 \rightarrow \mathbb{F}_q^*$. `Setup` algorithm is similar to that of BasicIdent, with the hash function H_2 replaced by H_3, i.e. the public system parameters are

$$\texttt{params} =< \mathbb{G}_1, \mathbb{G}_2, \hat{e}(\cdot,\cdot), p, n, P, P_{pub}, H_1(\cdot), H_3(\cdot,\cdot) > .$$

`Extract` algorithm is exactly the same as that of previous two constructions of IBE.

`Sign`: To sign a message $m \in \{0,1\}^n$, user of identity ID follows the steps below.

1. Choose k from $\mathbb{F}_q^*$ randomly.
2. Choose P_1 from $\mathbb{G}_1^*$ randomly.
3. Compute $\hat{r} = \hat{e}(P_1, P)^k$.
4. Compute $v = H_3(m, \hat{r})$.
5. Compute $S = vS_{ID} + kP$.
6. The signature is $\sigma = (v, S)$.

`Verify:` On receiving a message m and a signature (v', S'), any one can perform the verification as follows.

1. Compute $\hat{r}' = \hat{e}(S', P)\hat{e}(Q_{ID}, -P_{pub})^{v'}$
2. Accept the signature if and only if $v' = H_3(m, \hat{r}')$.

5.3 Hierarchical ID-Based Signature

In the hierarchical ID-based cryptosystem, PKGs are arranged in a tree structure and the identities of users (and PKGs) can be represented as vectors. A vector of dimension ℓ represents an identity at depth ℓ. Each identity ID of depth ℓ is represented as an ID-tuple $ID|\ell = \{ID_1, \cdots, ID_\ell\}$. The four algorithms of HIBS have similar functions to that of IBS except that the `Extract` algorithm in HIBS will generate the private key for a given identity which is either a normal user or a lower level PKG. The private key for identity ID of depth ℓ is denoted as $S_{ID|\ell}$ or S_{ID} if the depth of ID is not important. The functions of `Setup`, `Extract`, `Sign`, and `Verify` in HIBS are described as follows.

- `Setup`: Based on the input of an unary string 1^k where k is a security parameter, it outputs the common public parameters $params$, which include a description of a finite message space together with a description of a finite signature space; and the master secret s, which is kept secret by the Private Key Generator.

- `Extract`: Based on the input of an arbitrary identity ID, it makes use of the master secret s (for root PKG) or $S_{ID|j-1}$ (for lower level PKGs) if ID is of depth j to output the private key $S_{ID|j}$ for ID corresponding to $params$.

- `Sign`: Based on the input (m, S_{ID}), it outputs a signature σ, corresponding to $params$.

- `Verify`: Based on the input (σ, m, ID), it outputs $\top$ for "true" or $\perp$ for "false", depending on whether σ is a valid signature of message m signed by ID or not, corresponding to $params$.

Again, these algorithms must satisfy the standard consistency constraint of ID-based signature.

Let H_1 and H_4 be two cryptographic hash functions where $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$ and $H_4 : \{0,1\}^* \rightarrow \mathbb{G}_1$. Below is the construction of HIBS from [GS02].

`Setup:` On input of a security parameter $k \in \mathbb{N}$, the BDH parameter generator [BF01] will generate $\mathbb{G}_1$, $\mathbb{G}_2$, q and $\hat{e}(\cdot,\cdot)$. Then the PKG executes the following steps.

1. Select an arbitrary generator P_0 from $\mathbb{G}_1$.
2. Pick a random s_0 from $\mathbb{Z}_p$, which is the system's master secret key.
3. Compute $Q_0 = s_0 P_0$.
4. The public system parameters are

$$\texttt{params} =< \mathbb{G}_1, \mathbb{G}_2, q, P_0, Q_0, \hat{e}(\cdot,\cdot), H_1(\cdot), H_4(\cdot) > .$$

`KeyGen:` For an entity with $ID|k-1 = \{ID_1, ID_2, ..., ID_{k-1}\}$ of depth $k-1$, it uses its secret key $S_{ID|k-1}$ to generate the secret key for a user $ID|k$ (where the first $k-1$ elements of $ID|k$ are those in $ID|k-1$) as follows.

1. Compute $P_k = H_1(ID_1, ID_2, \cdots, ID_{k-1}, ID_k)$.
2. Pick random s_{k-1} from $\mathbb{Z}_p$.
3. Set the private key of the user be $S_{ID|k} = S_{ID|k-1} + s_{k-1}P_k = \sum_{i=1}^{k} s_{i-1}P_i$.
4. Send the values of $Q_i = s_i P_0$ for $1 \leq i \leq t-1$ to the user.

`Sign:` For a user $ID|k$ with secret key $S_k = \sum_{i=1}^{k} s_{i-1}P_i$ and the points $Q_i = s_i P_0$ for $1 \leq i \leq k$ to sign on a message m, he/she follows the steps below.

1. Pick a random number s_k from $\mathbb{Z}_p^*$.
2. Compute $P_M = H_4(ID_1, ID_2, \cdots, ID_k, M)$.
3. Compute $\sigma = S_k + s_k P_M$.
4. Return signature = $\{\sigma, Q_1, Q_2, \cdots, Q_k\}$.

`Verify:` For $ID|k = \{ID_1, ID_2, \cdots, ID_k\}$'s signature $\{\sigma', Q'_1, Q'_2, \cdots, Q'_k\}$, everyone can do the following to verify its validity.

1. Compute $P'_M = H_4(ID_1, ID_2, \cdots, ID_k, M)$.

2. Return $\top$ if $\hat{e}(P_0, \sigma')/\prod_{i=2}^{k} \hat{e}(Q'_{i-1}, P_i) = \hat{e}(Q'_0, P_1)\hat{e}(Q'_t, P'_M)$.

5.4 Security Notions of ID-based Signcryptions

In most of the traditional public key cryptosystems, the private keys of users are usually independent of each other as these keys are independently generated. But the situation is different in ID-based cryptosystems: the private key of all users are generated by the same party (the PKG) using the same master secret key. Hence we have more considerations in formulating the security notion of ID-based Cryptosystem.

5.4.1 IND-IDSC2-CCIA2

Malone-Lee [ML02] extended the notion of semantic security for public key encryption schemes to ID-based signcryption schemes. We modify this definition slightly and our security notion is referred as *indistinguishability of identity-based signcryptions of 2 keys under adaptive chosen-ciphertext-and-identity attacks* (IND-IDSC2-CCIA2). A similar notion has been used in [LQ03]. Consider the following IND-IDSC2-CCIA2 game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

Setup: The challenger $\mathcal{C}$ takes a security parameter k and runs `Setup` to generate common public parameters $params$ and the master secret key s. $\mathcal{C}$ sends $params$ to $\mathcal{A}$.

Phase 1: The adversary $\mathcal{A}$ can perform a polynomially bounded number of queries in an adaptive manner (that is, each query may depend on the responses to the previous queries). The types of queries allowed are described below.

- `Extract`: $\mathcal{A}$ chooses an identity ID. $\mathcal{C}$ computes $\texttt{Extract}(ID) = (S_{ID}, D_{ID})$ and sends the result to $\mathcal{A}$.

- `Signcrypt`: $\mathcal{A}$ chooses two identities ID_i and ID_j, and a plaintext m. $\mathcal{C}$ signcrypts the plaintext by computing $\sigma = \texttt{Signcrypt}\ (m, S_{ID_i}, ID_j)$ and sends σ to $\mathcal{A}$.

- `Unsigncrypt`: $\mathcal{A}$ chooses two identities ID_i and ID_j, and a ciphertext σ. $\mathcal{C}$ computes the private decryption key D_{ID_j} by calling `Extract`(ID_j), then unsigncrypts the ciphertext σ by calling `Unsigncrypt`(σ, ID_i, D_{ID_j}) and sends the resulting plaintext m or the symbol $\perp$ to $\mathcal{A}$

Challenge: The adversary $\mathcal{A}$ decides when Phase 1 ends. Then, it outputs two equal length plaintexts, m_0 and m_1, and two identities, ID_A and ID_B, on which it wishes to be challenged. The identity ID_B should not appear in any `Extract` queries in Phase 1. The challenger $\mathcal{C}$ picks a random bit b from $\{0, 1\}$, computes $\sigma =$ `Signcrypt` (m_b, S_{ID_A}, ID_B), and returns σ to $\mathcal{A}$.

Phase 2: The adversary $\mathcal{A}$ can ask a polynomially bounded number of queries adaptively again as in Phase 1 with the restriction that it cannot make an `Extract` query on ID_B and cannot make an `Unsigncrypt` query on (σ, ID_A, D_{ID_B}) to obtain the plaintext for σ.

Guess: The adversary $\mathcal{A}$ has to output a guess b'. It wins the game if $b' = b$.

The *advantage* of $\mathcal{A}$ is defined as $Adv(\mathcal{A}) = |2P[b' = b] - 1|$ where $P[b' = b]$ denotes the probability that $b' = b$.

Definition 11. *An ID-based signcryption scheme is said to have the indistinguishability against adaptive chosen-ciphertext-and-identity attacks property (IND-IDSC2-CCIA2 secure) if no adversary has a non-negligible advantage in the IND-IDSC2-CCIA2 game.*

Notice that the adversary is allowed to make an `Extract` query on the signcrypting identity ID_A in the above definition. This condition corresponds to the stringent requirements of *insider-security* for confidentiality of signcryption [ADR02]. On the other hand, it ensures the *forward-security* of the scheme, i.e. confidentiality is preserved in case the sender's private signcryption key becomes compromised.

5.4.2 EUF-IDSC2-CMIA2

Again, Malone-Lee [ML02] extended the notion of existential unforgeability for signature schemes to ID-based signcryption schemes. The security notion in our work is referred as *existential unforgeability of identity-based signcryptions of 2 keys under adaptive chosen-message-and-identity attacks* (EUF-IDSC2-CMIA2). Its formal definition is based on the following EUF-IDSC2-CMIA2 game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

Setup: The challenger $\mathcal{C}$ takes a security parameter k and runs `Setup` to generate common public parameters $params$ and the master secret key s. $\mathcal{C}$ sends $params$ to $\mathcal{A}$.

Attack: The adversary $\mathcal{A}$ can perform a polynomially bounded number of queries in an adaptive manner (that is, each query may depend on the responses to the previous queries). The types of queries allowed are described below.

- `Extract`: $\mathcal{A}$ chooses an identity ID. $\mathcal{C}$ computes $\texttt{Extract}(ID) = (S_{ID}, D_{ID})$ and sends the result to $\mathcal{A}$.

- `Signcrypt`: $\mathcal{A}$ chooses two identities ID_i and ID_j, and a plaintext m. $\mathcal{C}$ signcrypts the plaintext by computing $\sigma = \texttt{Signcrypt}\ (m, S_{ID_i}, ID_j)$ and sends σ to $\mathcal{A}$.

- `Unsigncrypt`: $\mathcal{A}$ chooses two identities ID_i and ID_j, and a ciphertext σ. $\mathcal{C}$ computes the private decryption key D_{ID_j} by calling $\texttt{Extract}(ID_j)$, then unsigncrypts the ciphertext σ by calling $\texttt{Unsigncrypt}(\sigma, ID_i, D_{ID_j})$ and sends the resulting plaintext m or the symbol $\perp$ to $\mathcal{A}$

Forgery: The adversary $\mathcal{A}$ outputs (σ, ID_A, ID_B) where ID_A did not appear in any `Extract` query in the Attack phase. It wins the game if the response of the `Unsigncrypt` on (σ, ID_A, D_{ID_B}) is not equal to $\perp$.

The advantage of $\mathcal{A}$ is defined as the probability that it wins.

Definition 12. *An ID-based signcryption scheme is said to have the existential unforgeability against adaptive chosen-message-and-identity attacks property (EUF-IDSC2-CMIA2 secure) if no adversary has a non-negligible advantage in the EUF-IDSC2-CMIA2 game.*

Note that in the above definition, the adversary is allowed to make an `Extract` query on the forged message's recipient ID_B. Again, this condition corresponds to the stringent requirements of *insider-security* for signcryption [ADR02], which is to ensure the non-repudiation property by preventing a dishonest user who holds a valid user's private key of the system from generating a valid ciphertext to himself/herself on other's behalf and claim the forged authenticity.

5.4.3 REA-IDSC2-CCIA2

Ciphertext anonymity is considered in [Boy03b], which means the ciphertext must contain no information (in the clear) that identifies the sender or recipient of the ciphertext. Since our work aimed to provide *public ciphertext authenticity*, we consider the *recipient anonymity*in our scheme, which is defined formally by the following *recipient anonymity of identity-based signcryptions of 2 keys under adaptive chosen-ciphertext-and-identity attacks* (REA-IDSC2-CCIA2) game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

Setup: The challenger $\mathcal{C}$ takes a security parameter k and runs `Setup` to generate common public parameters $params$ and the master secret key s. $\mathcal{C}$ sends $params$ to $\mathcal{A}$.

Phase 1: The adversary $\mathcal{A}$ can perform a polynomially bounded number of queries in an adaptive manner (that is, each query may depend on the responses to the previous queries). The types of queries allowed are described below.

- `Extract`: $\mathcal{A}$ chooses an identity ID. $\mathcal{C}$ computes $\texttt{Extract}(ID) = (S_{ID}, D_{ID})$ and sends the result to $\mathcal{A}$.

- `Signcrypt`: $\mathcal{A}$ chooses two identities ID_i and ID_j, and a plaintext m. $\mathcal{C}$ sign-

crypts the plaintext by computing $\sigma = \texttt{Signcrypt}\ (m, S_{ID_i}, ID_j)$ and sends σ to $\mathcal{A}$.

- `Unsigncrypt`: $\mathcal{A}$ chooses two identities ID_i and ID_j, and a ciphertext σ. $\mathcal{C}$ computes the private decryption key D_{ID_j} by calling $\texttt{Extract}(ID_j)$, then unsigncrypts the ciphertext σ by calling $\texttt{Unsigncrypt}(\sigma, ID_i, D_{ID_j})$ and sends the resulting plaintext m or the symbol $\perp$ to $\mathcal{A}$

Challenge: The adversary $\mathcal{A}$ decides when Phase 1 ends. Then, it outputs a message m, a sender's identity ID_A and two recipients' identities ID_{B_0} and ID_{B_1}, on which it wishes to be challenged. The identities ID_{B_0} and ID_{B_1} should not appear in any `Extract` queries in Phase 1. The challenger $\mathcal{C}$ picks a random bit b from $\{0, 1\}$, computes $\sigma = \texttt{Signcrypt}\ (m, S_{ID_A}, ID_{B_b})$, and returns σ to $\mathcal{A}$.

Phase 2: The adversary $\mathcal{A}$ can ask a polynomially bounded number of queries adaptively again as in Phase 1 with the restriction that it cannot make `Extract` query on ID_{B_0} or ID_{B_1}, it also cannot make an `Unsigncrypt` query on $(\sigma, ID_A, D_{ID_{B_0}})$ or $(\sigma, ID_A, D_{ID_{B_1}})$ to obtain the plaintext for σ.

Guess: The adversary $\mathcal{A}$ has to output a guess b'. It wins the game if $b' = b$.

The *advantage* of $\mathcal{A}$ is defined as $Adv(\mathcal{A}) = |2P[b' = b] - 1|$ where $P[b' = b]$ denotes the probability that $b' = b$.

Definition 13. *An ID-based signcryption scheme is said to have the recipient anonymity against adaptive chosen-ciphertext-and-identity attacks property (REA-IDSC2-CCIA2 secure) if no adversary has a non-negligible advantage in the REA-IDSC2-CCIA2 game.*

Similar to the IND-IDSC2-CCIA2 game, the adversary is allowed to make an `Extract` query on the signcrypting identity ID_A in the above definition. This condition corresponds to the stringent requirements of *insider-security* for recipient anonymity of signcryption, which the scheme in [NR03] and the schemes with public ciphertext authenticity in [LQ03] cannot satisfy.

5.5 Chapter Summary

In this chapter, we review 4 major ID-based cryptographic schemes from bilinear pairings, which includes 2 version of ID-based encryption, an ID-based signature scheme and a hierarchical ID-based signature scheme. Furthermore, we have proposed the insider security notion of adaptive chosen-ciphertext/message-and-identity attacks in ID-based signcryption schemes with private signcryption key and the private decryption key separated, which includes indistinguishability, existential unforgeability and recipient anonymity.

□ End of chapter.

Part III

Forward-Secure Cryptosystems

Chapter 6

Forward-Secure ID-Based Signcryption

A new ID-based signcryption scheme that can satisfy all the above requirements we have identified in Chapter 3 is presented in this chapter. We also show that our proposed scheme is provably secure based on the assumption of the computational hardness of variants of the Decisional and Computational Bilinear Diffie-Hellman problems we presented in Chapter 4.

6.1 Construction

Define $\mathbb{G}_1$, $\mathbb{G}_2$ and $\hat{e}(\cdot,\cdot)$ as in Chapter 4. Let H_1, H_2 and H_3 be three cryptographic hash functions where $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$, $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^n$ and $H_3 : \{0,1\}^n \times \mathbb{G}_2 \rightarrow \mathbb{Z}_q^*$. Let $E_{(\cdot)}(\cdot)$, $D_{(\cdot)}(\cdot)$ be the encryption and decryption algorithms of a secure symmetric cipher which takes a key of length n and a plaintext/ciphertext of length n respectively. (For example, a one-time pad cipher as used in [Boy03b].) The following shows the details of the scheme.

6.1.1 System Setup

`Setup`: Let P be an arbitrary generator of $\mathbb{G}_1$, the Private Key Generator (PKG) chooses $s \in \mathbb{Z}_q^*$ randomly and $P_{pub} = sP$. The master-key is s, which is kept secret and known only by PKG. The system parameters are

$$\{\mathbb{G}_1, \mathbb{G}_2, q, n, P, P_{pub}, \hat{e}(\cdot,\cdot), H_1(\cdot), H_2(\cdot), H_3(\cdot,\cdot), E_{(\cdot)}(\cdot), D_{(\cdot)}(\cdot)\}$$

6.1.2 Private Key Extraction

`Extract`: The user with identity $ID \in \{0,1\}^*$ submits ID to PKG. PKG sets the user's public key Q_{ID} to be $H_1(ID) \in \mathbb{G}_1$, computes the user's private signcryption key S_{ID} by $S_{ID} = s^{-1}Q_{ID}$ and private decryption key by $D_{ID} = sQ_{ID}$. Then PKG sends the private keys to the user.

6.1.3 Signcryption

`Signcrypt`: To send a message $m \in \{0,1\}^n$ to B, A follows the steps below.

1. Choose x from $\mathbb{Z}_q^*$ randomly.
2. Compute $X_A = xQ_{ID_A}$.
3. Compute $\hat{k}_1 = \hat{e}(X_A, P)$ and $\hat{k}_2 = H_2[\hat{e}(X_A, Q_{ID_B})]$.
4. Compute $c = E_{\hat{k}_2}(m)$.
5. Compute $r = H_3(c, \hat{k}_1)$.
6. Compute $S = (x - r)S_{ID_A}$.
7. The ciphertext is $\sigma = (c, r, S)$.

6.1.4 Unsigncryption

`Unsigncrypt`: To unsigncrypt a signcrypted message (c, r, S) from A, B follows the steps below.

1. Compute $R'_A = rQ_{ID_A}$.
2. Compute $\hat{k}'_1 = \hat{e}(S, P_{pub})\hat{e}(R'_A, P)$.
3. Compute $\hat{k}'_2 = H_2[\hat{e}(S, D_{ID_B})\hat{e}(R'_A, Q_{ID_B})]$.
4. Recover $m = D_{\hat{k}'_2}(c)$.
5. Compute $r' = H_3(c, \hat{k}'_1)$.

6. Accept the message if and only if $r' = r$, return $\perp$ otherwise.

7. Give $(\hat{k}_2', m, \sigma)$ to the third party.

6.1.5 Verification by Third Party

`TP_Verify`:

1. Compute $\hat{k}_1' = \hat{e}(S, P_{pub})\hat{e}(Q_{ID_A}, P)^r$.

2. Compute $r' = H_3(c, \hat{k}_1')$.

3. Accept the origin of ciphertext if and only if $r = r'$.

4. Moreover, accept the message authenticity if and only if $m = D_{\hat{k}_2'}(c)$.

5. Return $\top$ if all tests are passed, $\perp$ otherwise.

6.2 Improving the Efficiency

Although some research has been done in analyzing the complexity and speeding up the pairing computation (for examples, [BKLS02, GHS02, IT03, CL03]), pairing operations are still rather expensive. The following shows how to modify the `Signcrypt` algorithm of our scheme to make one pairing operation precomputable.

`SigncryptVariant`: To send a message $m \in \{0,1\}^*$ to B, A follows the steps below.

1. Choose x from $\mathbb{Z}_q^*$ randomly.

2. Compute $\hat{k}_1 = \hat{e}(Q_{ID_A}, P)^x$

3. Compute $\hat{k}_2 = H_2[\hat{e}(Q_{ID_A}, Q_{ID_B})^x]$.

4. Compute $c = E_{\hat{k}_2}(m)$.

5. Compute $r = H_3(c, \hat{k}_1)$.

6. Compute $S = (x - r)S_{ID_A}$.

7. The ciphertext is $\sigma = (c, r, S)$.

In this variant, $\hat{e}(Q_{ID_A}, P)$ is pre-computed since it is independent of the message and its intended recipient, so we only need a total of five pairing operations for signcryption, and unsigncryption processes. Note that this modification increments the total number of point multiplications and exponentiations by one; however, the scheme is as efficient as [Boy03b] and more efficient than other existing provable secure signcryption schemes with public verifiability.

6.3 Against Dishonest Recipient

Since the third party has no way to ensure the correctness of session key $\hat{k}_2'$ obtained from the recipient, dishonest recipient can randomly choose $\hat{k}_2'$ such that the signcrypted message (c, r, S) decrypts to a plaintext m' which is not equal to the real value $D_{H_2[\hat{e}(S,D_{ID_B})\hat{e}(Q_{ID_A},Q_{ID_B})^r]}(c)$. This issue is not addressed in previous work like [LQ03]. A simple fix to this attack is to ask the recipient to surrender the private decryption key, but this made the scheme rather inflexible and unrealistic.

Note that this existential forgery is not really dangerous in many cases as the resulting plaintext from the decryption using a random session key $\hat{k}_2'$ is usually unintelligible or not in a correct message format. Still we present modifications to our scheme which make our scheme secure against this attack. In the modifications, apart from the signcrypted message (c, r, S), recipient randomly chooses z from $\mathbb{Z}_q^*$ and sends zD_{ID_B}, $z^{-1}S$ and zP_{pub} to the third party. This does not compromise the recipient's decryption key and only enables the third party to decrypt signcrypted messages in the form of $(c^{\#}, r^{\#}, S^{\#})$ where $S^{\#} = S$, which is a rare case as S can be considered as randomly generated by the sender. The third party can compute a correct $\hat{k}_2'$ by itself after checking for the correctness of these additional data (zD_{ID_B}, $z^{-1}S$ and zP_{pub}) as follows.

`TP_VerifyVariant`

1. Compute $R'_A = rQ_{ID_A}$.
2. Compute $\hat{k}'_1 = \hat{e}(S, P_{pub})\hat{e}(R'_A, P)$.
3. Compute $r' = H_3(c, \hat{k}'_1)$.
4. Accept the origin of ciphertext if and only if $r = r'$.
5. Check whether $\hat{e}(zD_{ID_B}, P) = \hat{e}(zP_{pub}, Q_{ID_B})$.
6. Check whether $\hat{e}(z^{-1}S, zP_{pub}) = \hat{e}(S, P_{pub})$.
7. Compute $\hat{k}'_2 = H_2[e(z^{-1}S, zD_{ID_B})e(R'_A, Q_{ID_B})]$.
8. Accept the message authenticity if and only if $m = D_{\hat{k}'_2}(c)$.
9. Return $\top$ if all tests are passed, $\perp$ otherwise.

6.4 Analysis

6.4.1 Consistency

The consistency can be easily verified by the following equations.

$$\begin{aligned}
\hat{k}'_1 &= \hat{e}(S, P_{pub})\hat{e}(rQ_{ID_A}, P) \\
&= \hat{e}(xS_{ID_A}, P_{pub})\hat{e}(rS_{ID_A}, P_{pub})^{-1}\hat{e}(Q_{ID_A}, P)^r \\
&= \hat{e}(Q_{ID_A}, P)^x \\
\hat{k}'_2 &= H_2[\hat{e}(S, D_{ID_B})\hat{e}(rQ_{ID_A}, Q_{ID_B})] \\
&= H_2[\hat{e}(xS_{ID_A}, D_{ID_B})\hat{e}(rS_{ID_A}, D_{ID_B})^{-1}\hat{e}(Q_{ID_A}, Q_{ID_B})^r] \\
&= H_2[\hat{e}(Q_{ID_A}, Q_{ID_B})^x]
\end{aligned}$$

6.4.2 Confidentiality and Forward Security

Decryption requires the knowledge of $\hat{e}(Q_{ID_A}, Q_{ID_B})^x$. For a passive adversary, the information available is only σ and $\hat{k}_1$. It is difficult to get x from $\hat{k}_1$ since it is difficult to invert the bilinear pairing. Only S in the signature reveals x, but it is difficult to compute

x from S even with the knowledge of r and S_{ID_A} since it is difficult to compute discrete logarithm. Theorem 1 shows our scheme's confidentiality and forward security under adaptive chosen-ciphertext-and-identity attacks.

6.4.3 Unforgeability

Only the sender A with the knowledge of S_{ID_A} can compute S. Even with a previous valid signcrypted message of m from A, an adversary cannot make another signcrypted message m' where $m' \neq m$, since S in the signcrypted message is related to the ciphertext by $r = H_3(c, \hat{k}_1)$ and the hash function is assumed to be one-way and collision-free. The security of the scheme regarding the existential unforgeability under adaptive chosen-message-and-identity attacks is given in Theorem 2.

6.4.4 Recipient Anonymity or Recipient Verifiability

Only $\hat{k}_2$ of the ciphertext is related to the recipient's identity. Checking whether the ciphertext can be decrypted by a particular person requires either the knowledge of x or the private decryption key. As argued before, x is not revealed. Theorem 3 shows how our scheme achieves recipient anonymity under adaptive chosen-ciphertext-and-identity attacks.

Actually our scheme has dual support of recipient anonymity or recipient verifiability, according to the signer's choice. As in [JJR+03], if our scheme includes the receiver's public key in the signature (by setting $r = H_3(c, \hat{k}_1, Q_{ID_B})$), the *receiver* of the ciphertext is public verifiable as well.

6.4.5 Public Ciphertext Authenticity

Step 1 to 3 of `TP_Verify` only takes the ciphertext and the public system parameters as input and do not require the knowledge of $\hat{k}_2'$, hence the origin of ciphertext can be verified without knowing the content of messages and the help of intended recipient.

6.4.6 Public Verifiability

Similar to [LQ03], forwarding ephemeral key $\hat{k}_2'$ to any third parties convinces them that the ciphertext is the signcrypted version of a given plaintext message made by the sender (see Step 4 of `TP_Verify`), so our scheme satisfies our definition of public verifiability. Note that $\hat{k}_2'$ is just a random element computed from the public information as $\hat{k}_2 = H_2[\hat{e}(Q_{ID_A}, Q_{ID_B})^x]$, where x is randomly chosen.

6.4.7 Provable Security

Following the ideas in [LQ03] and [Boy03b], the following three theorems show that the proposed scheme is IND-IDSC2-CCIA2, EUF-IDSC2-CMIA2 and REA-IDSC2-CCIA2 secure.

Theorem 1. *In the random oracle model (the hash functions are modeled as random oracles), we assume that we have an adversary $\mathcal{A}$ that is able to win the IND-IDSC2-CCIA2 game (i.e. $\mathcal{A}$ is able to distinguish ciphertexts given by the challenger), with an advantage ϵ when running in a time t and asking at most q_H identity hashing queries, at most q_E key extraction queries, at most q_R H_3 queries, q_R* `Signcrypt` *queries and q_U* `Unsigncrypt` *queries. Then, there exists a distinguisher $\mathcal{C}$ that can solve the MDBDH problem in $O(t + (8{q_R}^2 + 4q_U)T_{\hat{e}})$ time with an advantage*

$$Adv(\mathcal{C})^{MDBDHP(\mathbb{G}_1,P)} > \frac{\epsilon(2^k - q_U) - q_U}{q_H 2^{(k+1)}}$$

where $T_{\hat{e}}$ denotes the computation time of the bilinear pairing and

$$\begin{aligned} Adv(\mathcal{C}) &= |P_{a,b,c\in\mathbb{Z}_q}[1 \leftarrow \mathcal{C}(aP, bP, cP, c^{-1}P, \hat{e}(P,P)^{abc})] \\ &- P_{a,b,c\in\mathbb{Z}_q, h\in\mathbb{G}_2}[1 \leftarrow \mathcal{C}(aP, bP, cP, c^{-1}P, h)]|. \end{aligned}$$

(For large k, the lower-bound of the advantage can be approximated by $\frac{\epsilon}{2q_H}$.)

Proof. Following the same idea as in [LQ03], we assume that the distinguisher $\mathcal{C}$ receives a random instance $(P, aP, bP, cP, c^{-1}P, h)$ of the MDBDH problem and has to

decide if $h = \hat{e}(P, P)^{abc}$. $\mathcal{C}$ will run $\mathcal{A}$ as a subroutine and act as $\mathcal{A}$'s challenger in the IND-IDSC2-CCIA2 game. During the game, $\mathcal{A}$ will consult $\mathcal{C}$ for answers to the random oracles H_1, H_2 and H_3. Roughly speaking, these answers are randomly generated, but to maintain the consistency and to avoid collision, $\mathcal{C}$ keeps three lists L_1, L_2, L_3 respectively to store the answers used. The following assumptions are made.

(1) $\mathcal{A}$ will ask for $H_1(ID)$ before ID is used in any `Signcrypt`, `Unsigncrypt` and `Extract` queries.

(2) $\mathcal{A}$ will not ask for `Extract`(ID) again if the query `Extract`(ID) has been already issued before.

(3) Ciphertext returned from a `Signcrypt` request will not be used by $\mathcal{A}$ in an `Unsigncrypt` request.

$\mathcal{C}$ gives $\mathcal{A}$ the system parameters with $P_{pub} = cP$. Note that c is unknown to $\mathcal{C}$. This value simulates the master key value for the PKG in the game.

H_1 requests: When $\mathcal{A}$ asks queries on the hash values of identities, $\mathcal{C}$ checks the list L_1, If an entry for the query is found, the same answer will be given to $\mathcal{A}$; otherwise, a value d_i from $\mathbb{Z}_q^*$ will be randomly generated and d_iP will be used as the answer, the query and the answer will then be stored in the list. Note that the associated private keys are d_icP and $d_ic^{-1}P$ which $\mathcal{C}$ knows how to compute.

The only exception is that $\mathcal{C}$ has to randomly choose one of the H_1 queries from $\mathcal{A}$, say the i-th query, and answers $H_1(ID_i) = bP$ for this query. Since bP is a value in a random instance of the MDBDH problem, it does not affect the randomness of the hash function H_1. Since both b, c and c^{-1} are unknown to $\mathcal{C}$, an `Extact` request on this identity will make $\mathcal{C}$ fails.

H_2, H_3 requests: When $\mathcal{A}$ asks queries on these hash values, $\mathcal{C}$ checks the corresponding list. If an entry for the query is found, the same answer will be given to $\mathcal{A}$; otherwise, a randomly generated value will be used as an answer to $\mathcal{A}$, the query and the answer will then be stored in the list.

`Signcrypt` requests: Let ID_A, ID_B be the identity of the sender and that of the recipient respectively and m be the plaintext used by $\mathcal{A}$ in a `Signcrypt` request. First we consider the simplest case that ID_A is not ID_i, then $\mathcal{C}$ can compute the private signcryption key S_{ID_A} correspondingly and the query can be answered by a call to $\texttt{Signcrypt}(m, S_{ID_A}, Q_{ID_B})$.

For the case $ID_A = ID_i$ and $ID_B \neq ID_i$, $\mathcal{C}$ answers $\texttt{Signcrypt}(m, S_{ID_A}, Q_{ID_B})$ query as follows. $\mathcal{C}$ randomly picks $r \in \mathbb{Z}_q$ and $S \in \mathbb{G}_1^*$, computes $\hat{k}_1$ by the formula $\hat{k}_1 = \hat{e}(S, P_{pub})\hat{e}(Q_{ID_A}, P)^r$ and $\hat{\tau} = \hat{e}(S, D_{ID_B})\hat{e}(Q_{ID_A}, Q_{ID_B})^r$ where D_{ID_B} is the private decryption key of ID_B. $\mathcal{C}$ finds $\hat{k}_2 = H_2(\hat{\tau})$ by running the simulation for H_2 and computes $c = E_{\hat{k}_2}(m)$. If there is a tuple $(c, \hat{k}_1, r')$ with $r' \neq r$ in L_3, $\mathcal{C}$ has to repeat the same process using another random pair (r, S) until the corresponding (c, $\hat{k}_1$) does not appear in any tuple in L_3. This process repeats at most $2q_R$ times as L_3 contains at most $2q_R$ entries ($\mathcal{A}$ can issue q_R H_3 queries and q_R `Signcrypt` queries, while each `Signcrypt` query contains a single H_3 query). When an appropriate pair (r, S) is found, the ciphertext (c, r, S) appears to be valid from $\mathcal{A}$'s viewpoint. $\mathcal{C}$ has to compute 4 pairing operations for each iteration of the process.

The last case to consider is when both of ID_A and ID_B are the identity ID_i, $\mathcal{C}$ signcrypts m as follows. $\mathcal{C}$ chooses $r^* \in \mathbb{Z}_q^*$ and $S^* \in \mathbb{G}_1$, computes $\hat{k}_1^*$ by the formula $\hat{k}_1^* = \hat{e}(S^*, P_{pub})\hat{e}(Q_{ID_A}, P)^{r^*}$ and randomly chooses $\hat{\tau}^* \in_R \mathbb{G}_2$ and $\hat{k}_2^* \in_R \{0,1\}^n$ such that no entry $(., \hat{k}_2^*)$ is in L_2 and computes $c^* = E_{\hat{k}_2^*}(m)$. He then checks if the list L_3 already contains an entry $(c^*, \hat{k}_1^*, r')$ such that $r' \neq r^*$. If not, $\mathcal{C}$ puts the tuple $(c^*, \hat{k}_1^*, r^*)$ into L_3 and $(\hat{\tau}^*, \hat{k}_2^*)$ into L_2. Otherwise, $\mathcal{C}$ chooses another random pair (r^*, S^*) and repeats the process as above until he finds a tuple $(c^*, \hat{k}_1^*, r^*)$ whose first two elements do not figure in an entry of L_3. Once an appropriate pair (r^*, S^*) is found, $\mathcal{C}$ gives the ciphertext $\sigma^* = (c^*, r^*, S^*)$ to $\mathcal{A}$. As $\mathcal{A}$ will not ask for the unsigncryption of σ^*, he will never see that σ^* is not a valid ciphertext of the plaintext m where $ID_A = ID_B = ID_i$ (since $\hat{\tau}^*$ may not equal to $\hat{e}(S^*, D_{ID_B})\hat{e}(Q_{ID_A}, Q_{ID_B})^{r^*}$). $\mathcal{C}$ has to compute 2 pairing operations for each iteration of the process.

`Unsigncrypt` requests: First we consider the case when $\mathcal{A}$ observes a ciphertext $\sigma' = (c', r', S')$ from ID_A to ID_B where $ID_B = ID_i$. $\mathcal{C}$ always answers $\mathcal{A}$ that σ is invalid when $\mathcal{A}$ submits an `Unsigncrypt` request. For $\mathcal{C}$ to fail the simulation, $\mathcal{A}$ has made H_3 request with the tuple $(c', \hat{e}(S', P_{pub})\hat{e}(Q_{ID_A}, P)^{r'})$ before and $\mathcal{C}$ has answered r'. There is a probability of at most $1/2^k$ that $\mathcal{C}$ answered r' (and that σ' was actually valid from $\mathcal{A}$'s point of view) and L_3 actually contains a tuple $(c', \hat{e}(S', P_{pub})\hat{e}(Q_{ID_A}, P)^{r'}, r')$ (as $\mathcal{C}$ rejected a valid ciphertext).

For the case that $ID_B \neq ID_i$, $\mathcal{C}$ first computes $\hat{k}'_1 = \hat{e}(S', P_{pub})\hat{e}(Q_{ID_A}, P)^{r'}$. $\mathcal{C}$ rejects the ciphertext if the tuple $(c', \hat{k}'_1, r')$ is not found in the list L_3; otherwise, he can recover r' and compute $\hat{\tau}' = \hat{e}(S', D_{ID_B})\hat{e}(Q_{ID_A}, Q_{ID_B})^{r'}$. Note that the knowledge of D_{ID_B} can be simulated using the same technique in the simulation for the `Signcrypt` query. $\mathcal{C}$ then searches for a tuple $(\hat{\tau}', \cdot)$ in list L_2. If no such tuple is found, $\mathcal{C}$ picks a random pair $(\hat{\tau}, \hat{k}'_2) \in \mathbb{G}_2 \times \{0,1\}^n$ such that no tuples with $\hat{k}'_2$ already exists in L_2 and inserts $(\hat{\tau}, \hat{k}'_2)$ in L_2. $\mathcal{C}$ can use the corresponding $\hat{k}'_2$ to find $m' = D_{\hat{k}'_2}(c')$ and returns m'. The probability to reject at least 1 valid ciphertext is equal to $1 - ((2^k - 1)/2^k)^{q_U} = (q_U 2^{k(q_U-1)} - C_2^{q_U} 2^{k(q_U-2)} + \cdots)/2^{kq_U}$ which does not exceed $q_U/2^k$. For each unsigncryption request, $\mathcal{C}$ has to compute 4 pairing operations.

After the first stage, $\mathcal{A}$ picks a pair of identities on which he wishes to be challenged. Note that $\mathcal{C}$ fails if $\mathcal{A}$ has asked an `Extract` query on ID_i during the first stage. We know that the probability for $\mathcal{C}$ not to fail in this stage is $(\frac{q_H-1}{q_H})(\frac{q_H-2}{q_H-1}) \cdots (\frac{q_H-q_E}{q_H-q_E+1}) = \frac{q_H-q_E}{q_H}$. Further, with a probability exactly $(\frac{q_H-q_E-1}{q_H-q_E})(\frac{1}{q_H-q_E-1}) = \frac{1}{q_H-q_E}$, $\mathcal{A}$ chooses to be challenged on the pair (ID_j, ID_i) with $j \neq i$, while ID_j is the sender's identity and ID_i is the recipient's identity. Note that if $\mathcal{A}$ has submitted an `Extract` query on ID_i, then $\mathcal{C}$ fails because he is unable to answer the question. On the other hand, if $\mathcal{A}$ does not choose (ID_j, ID_i) as target identities, $\mathcal{C}$ fails too. Hence the probability that $\mathcal{A}$'s response is helpful to $\mathcal{C}$ is $\frac{1}{q_H}$.

Then $\mathcal{A}$ produces two plaintexts m_0 and m_1, $\mathcal{C}$ randomly picks a bit $b \in \{0,1\}$ and signcrypts m_b. To do so, he sets $S' = aP$ and chooses $r' \in \mathbb{Z}_q^*$. Suppose $ID_j = dP$,

setting $S' = aP$ implies $(x - r')dc^{-1} = a$, i.e. $x = acd^{-1} + r'$. Since aP and cP belongs to a random instance of the MDBDH problem, x is random and this will not modify $\mathcal{A}$'s view. $\mathcal{C}$ computes $\hat{k}'_1 = \hat{e}(S', P_{pub})\hat{e}(Q_{ID_j}, P)^{r'} = \hat{e}(aP, cP)\hat{e}(Q_{ID_j}, P)^{r'}$, $\hat{\tau}' = h\hat{e}(Q_{ID_j}, bP)^{r'}$ (where h is $\mathcal{C}$'s candidate for the MDBDH problem) to obtain $\hat{k}'_2 = H_2(\hat{\tau}')$ (from the H_2 simulation algorithm) and $c_b = E_{\hat{k}'_2}(m_b)$. He then verifies as above if L_3 already contains an entry $(c_b, \hat{k}'_1, r'')$ such that $r'' \neq r'$. If not, he puts the tuple $(c_b, \hat{k}'_1, r')$ into L_3. Otherwise, $\mathcal{C}$ picks another random r' and repeats the process until a tuple $(c_b, \hat{k}'_1, r')$ whose first two elements do not appear in any entry of L_3 is found. After an appropriate r' is found, $\mathcal{C}$ sends the ciphertext $\sigma = (c_b, r', S')$ to $\mathcal{A}$.

$\mathcal{A}$ then performs another set of queries, $\mathcal{C}$ can handle these queries as in the first stage. At the end, $\mathcal{A}$ will produce a bit b' as $\sigma = \texttt{Signcrypt}(m_{b'}, S_{ID_j}, Q_{ID_i})$ from $\mathcal{A}$'s viewpoint. If $b = b'$, $\mathcal{C}$ then answers 1 as the result to the MDBDH problem as he has produced a valid signcrypted message of m_b using the knowledge of h. Otherwise, $\mathcal{C}$ should answer 0.

Taking into account all the probabilities that $\mathcal{C}$ will not fail its simulation, the probability that $\mathcal{A}$ chooses to be challenged on the pair (ID_j, ID_i), and also the probability that $\mathcal{A}$ wins the IND-IDSC2-CCIA2 game, the value of $Adv(\mathcal{C})$ is calculated as follows.

$$\begin{aligned} Adv(\mathcal{C}) &> (\frac{(\epsilon+1)}{2}(1 - \frac{q_U}{2^k}) - 1/2)(\frac{1}{q_H}) \\ &= \frac{\epsilon(2^k - q_U) - q_U}{q_H 2^{(k+1)}} \end{aligned}$$

Regarding the time complexity, it can be verified by counting the number of pairing operations required to answer all queries.

□

Theorem 2. *In the random oracle model (the hash functions are modeled as random oracles), we assume that we have an adversary $\mathcal{A}$ that is able to win the EUF-IDSC2-CMIA2 game with an advantage $\epsilon \geq 10(q_S + 1)(q_S + q_R)q_H/(2^k - 1)$ within a time span t for a security parameter k; and asking at most q_H identity hashing queries, at*

most q_E key extraction queries, at most q_K H_2 queries, q_R H_3 queries, q_S `Signcrypt` *queries and q_U* `Unsigncrypt` *queries. Then, there exists an algorithm $\mathcal{C}'$ that can solve the MCBDH problem in expected time* $\leq 120686 q_R q_H 2^k t/\epsilon(2^k - 1)$.

Proof. We use the forking lemma [PS00] to prove the security of the scheme. To apply the forking lemma, we need to show how our scheme fits into the signature scheme described in [PS00], the simulation step in which the signature can be simulated without the secret signcryption key of the sender (and thus, also without the master secret), and how we can solve a difficult problem (MCBDH problem in our case) based on the forgery.

First, we observe that our scheme satisfies all the required properties of a generic signature scheme as described : during the signcryption of message m, the tuple (σ_1, h, σ_2) is produced which corresponds to the required three-phase honest-verifier zero-knowledge identification protocol, where $\sigma_1 = \hat{e}(xQ_{ID_A}, P)$ is the commitment of the prover (σ_1 can be considered to be chosen randomly from a large set since x is chosen randomly from $\mathbb{Z}_q^*$ and $\mathbb{G}_2$ is a cyclic group of prime order q), $h = H_3(c, \hat{k}_1)$ is the hash value depending on m and σ_1 (as c is a function of m) substituted for the verifier's challenge, and $\sigma_2 = S$ (which depends on x in σ_1, h and the signcryption key S_{ID_A}) is the response of the prover. As pointed out in [PS00], σ_1 can be omitted in the final signature produced in the scheme to optimize the size of signature since it can be correctly recovered during the verification process.

Next, we need to show a simulation step that provides a faithful simulation to the forger $\mathcal{A}$ and how to solve the MCBDH problem by interacting with $\mathcal{A}$. The distinguisher $\mathcal{C}$ receives a random instance $(P, aP, bP, cP, c^{-1}P)$ of the MCBDH problem and is required to compute $h = \hat{e}(P, P)^{abc}$. $\mathcal{C}$ will run $\mathcal{A}$ as a subroutine and act as $\mathcal{A}$'s challenger in the EUF-IDSC2-CMIA2 game. $\mathcal{C}$ needs to maintain lists L_1, L_2, L_3 to keep track values reported for random oracle queries H_1, H_2, and H_3 to avoid collision and maintain consistency for answers to these hashing oracles. $\mathcal{C}$ publishes the system parameters and handles H_1, H_2, H_3, `Signcrypt` and `Unsigncrypt` requests in the same

way as that in the proof of Theorem 1.

We calculate the probability of success of $\mathcal{C}$ as follows. For $\mathcal{C}$ to succeed, $\mathcal{A}$ did not ask an `Extract` query on ID_i. And the corresponding probability is at least $1/q_H$, as there are at most q_H entries in H_1. The probability of having a faithful simulation is at least $(1 - \frac{q_U}{2^k})(\frac{q_H - q_E}{q_H}) = \frac{(q_H - q_E)(2^k - q_U)}{q_H 2^k}$.

We follow the same idea used in [Boy03b] to coalesce the signing identity ID_i and the message m into a "generalized" forged message (ID_i, m) so as to hide the identity-based aspect of the EUF-IDSC2-CMIA2 attacks, and simulate the setting of an identity-less adaptive-CMA existential forgery for which the forking lemma is proven.

It follows from the forking lemma that if $\mathcal{A}$ is a sufficiently efficient forger in the above interaction, then we can construct a Las Vegas machine $\mathcal{A}'$ that outputs two signed messages $((ID_i, m), r, S)$ and $((ID_i, m), r', S')$ with $r \neq r'$ and the same commitment x.

Finally, to solve the MCBDH problem given the machine $\mathcal{A}'$ derived from $\mathcal{A}$, we construct a machine $\mathcal{C}'$ as follows.

1. $\mathcal{C}'$ runs $\mathcal{A}'$ to obtain two distinct forgeries $((ID_i, m), r, S)$ and $((ID_i, m), r', S')$.

2. $\mathcal{C}'$ derives the value of $bc^{-1}P$ as $(r' - r)^{-1}(S - S')$ (since the value of x is the same in both forgeries).

3. $\mathcal{C}'$ derives the value of $\hat{e}(P, P)^{abc^{-1}}$ by $\hat{e}(aP, bc^{-1}P)$ (the role of c and c^{-1} are interchangeable).

Note that the machine $\mathcal{C}'$ is our reduction from the MCBDH problem. Based on the bound from the forking lemma [PS00] and the lemma on the relationship between given-identity attack and chosen-identity attack [CC02], if $\mathcal{A}$ succeeds in time $\leq t$ with probability $\epsilon \geq 10(q_R+1)(q_S+q_R)q_H/(2^k-1)$, then $\mathcal{C}'$ can solve the MCBDH problem in expected time $\leq 120686 q_R q_H 2^k t/\epsilon(2^k - 1)$. □

Theorem 3. *In the random oracle model (the hash functions are modeled as random oracles), we assume that we have an adversary $\mathcal{A}$ that is able to win the REA-IDSC2-CCIA2 game (i.e. $\mathcal{A}$ is able to distinguish the recipient of the ciphertexts given by the challenger), with an advantage ϵ when running in a time t and asking at most q_H identity hashing queries, at most q_E key extraction queries, at most q_R H_3 queries, q_R* `Signcrypt` *queries and q_U* `Unsigncrypt` *queries. Then, there exists a distinguisher $\mathcal{C}$ that can solve the MDBDH problem in $O(t + (8{q_R}^2 + 4q_U)T_{\hat{e}})$ time with an advantage*

$$Adv(\mathcal{C})^{MDBDHP(\mathbb{G}_1,P)} > \frac{\epsilon(2^k - q_U) - q_U}{q_H 2^k}$$

where $T_{\hat{e}}$ denotes the computation time of the bilinear pairing and

$$\begin{aligned} Adv(\mathcal{C}) &= |P_{a,b,c\in\mathbb{Z}_q}[1 \leftarrow \mathcal{C}(aP, bP, cP, c^{-1}P, \hat{e}(P,P)^{abc})] \\ &- P_{a,b,c\in\mathbb{Z}_q, h\in\mathbb{G}_2}[1 \leftarrow \mathcal{C}(aP, bP, cP, c^{-1}P, h)]|. \end{aligned}$$

(For large k, the lower-bound of the advantage can be approximated by $\frac{\epsilon}{q_H}$.)

Proof. The major part of the proof is similar to that of Theorem 1. We assume that the distinguisher $\mathcal{C}$ receives a random instance $(P, aP, bP, cP, c^{-1}P, h)$ of the MDBDH problem and has to decide if $h = \hat{e}(P,P)^{abc}$. The simulation of random oracles (H_1, H_2 and H_3), `Signcrypt` oracle and `Unsigncrypt` oracle are the same in the proof of Theorem 1.

After the first stage, $\mathcal{A}$ picks a message m', a sender identity (ID_j and a pair of identities on which he wishes to be challenged. The probability that $\mathcal{A}$ has not asked an `Extract` query on ID_i during the first stage is $\frac{q_H - q_E}{q_H}$ as shown in the proof of Theorem 1. $\mathcal{A}$ chooses to be challenged on the pair (ID_k, ID_i) with $k \neq i$ with a probability exactly $(\frac{q_H - q_E - 1}{q_H - q_E})(\frac{1}{q_H - q_E - 1}) + \frac{1}{q_H - q_E} = \frac{2}{q_H - q_E}$. If $\mathcal{A}$ has submitted an `Extract` query on ID_i, then $\mathcal{C}$ fails because he is unable to answer the question. On the other hand, if the pair of identity chosen by $\mathcal{A}$ does not include ID_i as one of the target identity, $\mathcal{C}$ fails. Hence the probability that $\mathcal{A}$'s response is helpful to $\mathcal{C}$ is $\frac{2}{q_H}$. Without loss of generality we assume $ID_b = ID_k$ when $b = 0$ and $ID_b = ID_i$ when $b = 1$.

$\mathcal{C}$ gives the challenge to $\mathcal{A}$ by firstly computing $\hat{k}_1' = \hat{e}(aP, cP)\hat{e}(Q_{ID_j}, P)^{r'}$, $\hat{\tau}' = h\hat{e}(Q_{ID_j}, bP)^{r'}$ (where h is $\mathcal{C}$'s candidate for the MDBDH problem) to obtain $\hat{k}_2' = H_2(\hat{\tau}')$ (from the H_2 simulation algorithm) and $c' = E_{\hat{k}_2'}(m')$. He then verifies as above if L_3 already contains an entry $(c', \hat{k}_1', r'')$ such that $r'' \neq r'$. If not, he puts the tuple $(c', \hat{k}_1', r')$ into L_3. Otherwise, $\mathcal{C}$ picks another random r' and repeats the process until a tuple $(c', \hat{k}_1', r')$ whose first two elements do not appear in any entry of L_3 is found. After an appropriate r' is found, $\mathcal{C}$ sends the ciphertext $\sigma = (c', r', S')$ to $\mathcal{A}$.

These steps are the same as the challenge given by $\mathcal{C}$ in the proof of Theorem 1, which produce a ciphertext σ from ID_j to ID_i that is of correct distribution with the view of $\mathcal{A}$. Although the simulation of challenge ciphertext is independent of ID_k chosen by $\mathcal{A}$, it is not deterministic as $ID_i = bP$ belongs to a random instance of MDBDHP. $\mathcal{A}$ doesn't know the fact that $\mathcal{C}$ cannot compute the private key of ID_i since he does not requested for the private key of both ID_i and ID_k. To conclude, $\mathcal{A}$ cannot gain any extra advantage in winning the game from this simulation of ciphertext.

$\mathcal{A}$ then performs another set of queries, $\mathcal{C}$ can handle these queries as in the first stage. At the end, $\mathcal{A}$ will produce a bit b' as $\sigma = \texttt{Signcrypt}(m', S_{ID_j}, Q_{ID_{b'}})$ from $\mathcal{A}$'s viewpoint. If $b' = 1$, $\mathcal{C}$ then answers 1 as the result to the MDBDH problem as he has produced a valid signcrypted message of m' from ID_j to ID_i using the knowledge of h. Otherwise, $\mathcal{C}$ should answer 0.

Taking into account all the probabilities that $\mathcal{C}$ will not fail its simulation, the probability that $\mathcal{A}$ chooses to be challenged on the pair of recipient identities (ID_k, ID_i), and also the probability that $\mathcal{A}$ wins the REA-IDSC2-CCIA2 game, the value of $Adv(\mathcal{C})$ is calculated as follows.

$$\begin{aligned} Adv(\mathcal{C}) &> (\frac{(\epsilon+1)}{2}(1-\frac{q_U}{2^k}) - 1/2)(\frac{2}{q_H}) \\ &= \frac{\epsilon(2^k - q_U) - q_U}{q_H 2^k} \end{aligned}$$

Regarding the time complexity, it can be verified by counting the number of pairing operations required to answer all queries.

□

6.4.8 Independence of Signcryption key and Decryption Key

In our scheme, the signcryption key and the decryption key are separated. Moreover, given one of the private keys, the computation of the other key required the solving of inverse computational Diffie-Hellman problem, which is equivalent to computational Diffie-Hellman problem [BDZ03].

This property is indeed essential. From the view point of key archival requirements, it is better to ensure that the signcryption key is destroyed securely after its life time while the decryption key should be properly archived (for example, it should be made available to the government in the United States). From the view point of security, in case either one of the private signcryption key or the private decryption key is compromised by an adversary, the other key remains safe.

6.4.9 Efficiency

Considering the computational efficiency of the proposed scheme, signcryption requires one pairing operation, two exponentiations and one point multiplication, while unsigncryption and verification need four pairing operations and one point multiplication. For the third party's verification, one exponentiation and two pairing operations are required.

Another dimension to consider in signcryption scheme is the ciphertext size, our scheme's ciphertext size is $|\mathbb{G}_1| + |\mathbb{Z}_q^*| + |m|(+|ID|)$, i.e. our scheme's ciphertext consists of one element from $\mathbb{G}_1$, one element from $\mathbb{Z}_q^*$ together with number of bits necessary to represent a message and the identity (if it is not known to the recipient). This size is similar to the ciphertext size produced by most of the other signcryption schemes we considered. (In the second scheme of [LQ03], shorter ciphertext size is achieved by restricting the length of the message to be signcrypted, and we believe that the same technique can be used in our proposed schemes as well.) On the other hand, the ciphertext size from the simple "Encrypt-then-Sign" approach (based on Boneh-

Franklin's ID-based encryption scheme [BF01] and Hess's signature scheme [Hes03]) is $2|\mathbb{G}_1| + |\mathbb{Z}_q^*| + 2|m|(+|ID|)$.

The small ciphertext size is especially useful in situation where bandwidth is limited like MANET. The study in [PML04] showed that using ID-based signcryption scheme in secure routing protocol can save the communication overhead significantly when compared with existing solutions like [PH02] and [SDL+02].

6.5 Chapter Summary

Table 6.1 shows a summary of comparing our scheme with other existing schemes in terms of the identified requirements and efficiency. We use ML, NR, XB to denote the scheme in [ML02], [NR03], [Boy03b] respectively. In [LQ03], there are three signcryption schemes, a basic version, a modified version with shorter ciphertext, and a modified version with forward security. These three schemes are denoted as LQ-Basic, LQ-Short and LQ-Fwd respectively. Finally, we use SCSC 1 to denote our first proposed scheme and SCSC 2 to denote our proposed scheme's variant.

For efficiency, we compare the number of operations needed for the signcryption, unsigncryption, and the verification processes. We only consider the operations executed by the sender and the recipient, but not the third party as not all schemes are publicly verifiable. The following shows the types of operations that we consider.

1. *Pairing (Pa)*: The total number of pairing computations required. In the table, we represent this total in the form of $x(+y)$ where y is the number of operations that are independent of *the message and the recipient*, so these y operations can be pre-computed.

2. *Point Multiplication (Mu)*: The total number of point multiplications required.

3. *Exponentiation (Ex)*: The total number of exponentiations required.

[2] Its receipt anonymity is not yet proved/disproved.
[3] Its semantic security is not yet proved/disproved.
[4] We believe that it can be modified so that the number of Mu + Ex is 4 with 6 (+ 0) pairing operations.

	Security Requirement					Efficiency		
Schemes	FwSec	PubVer	PubCAuth	ReAnon	ProvSec	Pa	Mu	Ex
ML	Y	Y	Y	*N*	*N*	5 (+ 0)	3	1
SK[2]	Y	*N*	*N*	Y	*N*	2 (+ 0)	2	2
NR[3]	Y	*N*	*N*	*N*	*N*	4 (+ 0)	2	2
LQ-Basic[4]	*N*	Y	Y	*N*	Y	5 (+ 1)	2	4
LQ-Short[4]	*N*	Y	*N*	*N*	Y	5 (+ 1)	2	4
LQ-Fwd[2]	Y	*N*	*N*	Y	Y	3 (+ 0)	4	0
XB	Y	Y	*N*	Y	Y	5 (+ 0)	2	2
SCSC 1	Y	Y	Y	Y	Y	6 (+ 0)	3	0
SCSC 2	Y	Y	Y	Y	Y	5 (+ 1)	2	2

Table 6.1: *Comparison on Efficiency and Features of Existing Signcryption Schemes*

As shown above, our proposed scheme is the most efficient one in terms of the total number of point multiplications and exponentiations. Considering the number of pairing operations, our scheme is comparable to all of the existing schemes, in particular, for those provable secure scheme with public verifiability (without public verifiability, the scheme is an authenticated encryption scheme only rather than a signcryption scheme).

To summarize, our proposed scheme can satisfy all the requirements that we have identified. With pre-computed pairing result, our scheme can be executed at a cost comparable or even lower than those provably secure schemes that provide public verifiability. Our scheme's ciphertext size is comparable to all other existing work, but it is significantly smaller than that of traditional "Encrypt-then-Sign" approach.

□ End of chapter.

Chapter 7

Forward-Secure Threshold Signature

New construction of forward-secure threshold signature scheme based on the technique of polynomial secret sharing is proposed in this chapter. For key updates, our scheme is more efficient than the previous constructions. For signing, our scheme outperforms all previous schemes in terms of number of communication rounds as it is the first round-optimal (one-round) forward-secure threshold signature scheme. Besides, we also integrate the concept of proactive security into our scheme.

7.1 Building Blocks

In our proposed polynomial-based scheme, we integrate the round-optimal distributed key generation in [ZI03] and the forward-secure signature scheme in [KPH04].

7.1.1 Threshold Secret Sharing

Many threshold schemes are based on Shamir's secret sharing, which is derived from the concept of Lagrange polynomial interpolation.

For a (t, n) instantiation, a trusted dealer first selects t random coefficients $a_0, a_1, \cdots, a_{t-1}$ from $\mathbb{Z}_q$ where a_0 is the master secret to be shared. Then n different public points $x_{i_j} \in \mathbb{Z}_q^*$ are chosen (where $1 \leq j \leq n$), one for each participant. Let f be a polynomial of degree $t-1$ and $f(x) = a_0 + a_1 x + \cdots + a_{t-1} x^{t-1}$, the share to be distributed to the participant with public point x_{i_j} assigned is $f(x_{i_j})$.

When t participants decided to reconstruct the secret, they can do so by recovering the polynomial. With the knowledge of t points $(x_{i_j}, f(x_{i_j}) = s_{i_j})$ on the curve, the coefficients $(a_0, \cdots, a_t)$ of f can be computed by solving the following system of equations.

$$\begin{aligned} s_{i_1} &= a_0 + a_1 x_{i_1} + \cdots + a_{t-1} x_{i_1}^{t-1}, \\ s_{i_2} &= a_0 + a_1 x_{i_2} + \cdots + a_{t-1} x_{i_2}^{t-1}, \\ &\vdots \\ s_{i_t} &= a_0 + a_1 x_{i_t} + \cdots + a_{t-1} x_{i_t}^{t-1}, \end{aligned}$$

The above system has a unique solution for $(a_0, \cdots, a_t)$ since

$$\Delta = \begin{pmatrix} 1 & x_{i_1} & \ldots & x_{i_1}^{t-1} \\ 1 & x_{i_2} & \ldots & x_{i_2}^{t-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{i_t} & \ldots & x_{i_t}^{t-1} \end{pmatrix}$$

is a non-zero Vandermonde determinant (all of its elements are non-zero and pair-wise unique). The unique solution and hence the polynomial can be found by the Lagrange interpolation of these t points by using the below formula.

$$f(x) = \sum_{j=1}^{t} s_{i_j} \prod_{1 \le l \le t, l \ne j} \frac{x - x_{i_l}}{x_{i_j} - x_{i_l}}.$$

Thus the secret $a_0 = f(0)$ can be obtained by $\sum_{j=1}^{t} b_j s_{i_j}$ where $b_j = \prod_{1 \le l \le t, l \ne j} \frac{x_{i_l}}{x_{i_l} - x_{i_j}}$.

7.1.2 Forward Secure Signature from HIBS

The forward-signature schemes in [HWI03, ZX04, KPH04] borrowed the idea of hierarchical ID-based signature (HIBS) [GS02]. In HIBS, there is a tree structure of PKGs, each PKG verifies the PKGs at one level lower and generates private keys for

them. Finally the end-users at the leaf node of the hierarchy receive certification and its own private key from the chain of PKGs. In this setting, PKGs at higher level of hierarchy can generate the private key of its children, but the converse is not true. If we associate different time periods as the name of each PKGs of this hierarchy according to the above key generation property, so the private key of one time period can be used to generate the private keys of later time period but not those prior, then a forward-secure signature scheme is yielded.

We first consider the case for only using the leaf-node of the tree of hierarchy to represent the time periods as [HWI03] did. If we name the root PKG as ϵ (empty string), and uses binary representation of its position in j bits, where j is the depth of the node, to name the "intermediate" PKGs (e.g. the children of node ϵ are 0 and 1 respectively), then a PKG hierarchy of height l can be used to implement a forward-secure signature scheme, where each leave node of the tree is used to represent one of the $2^l - 1$ time period of a forward-secure scheme.

For each "key extract" operation, a node will use its private key to generate private keys for its children node. In the below description, we use the notations 0^a (and 1^b) to represent the bit string containing a '0's (and b '1's respectively). At the first time period, the master secret key of the node ϵ is randomly chosen, then node ϵ and subsequently the nodes 0, 00, $\cdots$, 0^{l-1} execute the extract operation to generate the "local keys" of the nodes 0, 00, $\cdots$, 0^l, notice that the keys for the nodes 1, 01, $\cdots$, $0^{l-2}1$ have been generated in these operations too. In the next time period, the key update is done by using the local key of the node 0^{l-1} to generate the local key for the node $0^{l-1}1$, at the same time, the local key of the node 0^l is deleted. Since this new key is randomly generated and the old local key is deleted, the knowledge of evolved private key cannot help in getting the old private key. In the third time period, we no longer use the local key for the node 0^{l-1} to generate the new key but use the local key of the node $0^{l-2}1$ to generate the local key for the node $0^{l-2}10$. A similar process continues until the local key of the node 1^l has been generated, which means the scheme has come to the end of

its service time.

Now we consider using all the nodes of the tree of hierarchy, including the root node [ZX04, KPH04]. If we name the root PKG as ϵ (empty string), and uses binary representation of its position in j bits, where j is the depth of the node, to name the "intermediate" PKGs (e.g. the children of node ϵ are 0 and 1 respectively), then a PKG hierarchy of height h can be used to implement a forward-secure signature scheme, where each node of the tree is used to represent one of the $2^h - 1$ time period of a forward-secure scheme, in a pre-order traversal. Notice that our scheme starts by the "zero-th" time period.

In the below description, we denote the node (named by bit string) corresponding to the j-th time period by w^j, hence the left (and right) child of the node w^j is w^j0 (and w^j1 respectively). We use w'^j to represent the longest bit string that makes w'^j0 a prefix of w^j. To better illustrate, the pre-order traversal can be defined with the notation w'^j. If w^j is an internal node, then the node representing the next time period (w^{j+1}) is w^j0. On the other hand, if w^j is a leaf node (and $j < 2^h - 1$, i.e. the scheme has not reached the end of its service time), then $w^{j+1} = w'^j1$. For example, for a tree with height = 2, consider leaf node $w^3(= 01)$, w'^3 is ϵ and hence $w^4 = \epsilon 1 = 1$.

We use the term "local secret key" for the secret key of a node which is responsible for signing and the key evolution of certain time periods. In the secret storage, there will be elements other than the local secret key for the key evolution of other time periods. Considering the key extract operations, if w^j is an internal node, the local secret key will be used for the generation of local secret key for the next time period and the time period w^j1; if w^j is a leaf node, the local secret key for the next time period has been already generated at prior time period.

Below is a concrete example. The first node (which is for the "zero-th" time period) is the root node $w^0 = \epsilon$. At this time period, the master secret key of the node ϵ is randomly chosen. When this time period finishes, the node ϵ executes the extract operation to generate the local keys of the nodes 0 and 1. After these two local keys are

generated, the local key of the node ϵ is deleted so as to maintain the forward security. The local key of the node 0 will be used to sign message at the first ($w^1 = 0$) time period and generate the key for the second ($w^2 = 00$) time period to $(2^{h-1} - 1)$-th time period; while the local key of the node 1 is responsible for signing the message at the (2^{h-1})-th ($w^{2^{h-1}} = 1$) time period and the key generation of $(2^{h-1} + 1)$-th ($w^{2^{h-1}+1} = 10$) time period to $(2^h - 1)$-th time period. At any time, the secret key storage of a node contains its own local secret key (i.e. the key for the current period) and also the local secret key of the node w'^j1 whenever w'^j0 is a prefix of w^j, for later key evolution.

7.1.3 Round-Optimal Distributed Key Generation

In the rest of the thesis, we use `DisKeyGen` to refer the below distributed key generation algorithm proposed in [ZI03].

Let g be a generator of the group $\mathbb{Z}_q$ (input), and x_j be the secret share given to each entity ID_j (output).

1. For $1 \leq i \leq n$, ID_i generates $z_i \in_R \mathbb{Z}_q$, computes $y_i = g^{z_i}$. ID_i then constructs a polynomial of degree t based on Shamir's secret sharing scheme [Sha79] to share z_i as the secret: let $a_{i,0} = s_{i,0} = z_i$ and picks $a_{i,k}$ at random from $\mathbb{Z}_q$ for $1 \leq k \leq t$. The polynomial is $f_i(x) = \sum_{k=0}^{t} a_{i,k}x^k \in \mathbb{Z}_q[x]$. ID_i computes $s_{i,j} = f_i(j)$ and broadcasts the following: $A_{i,k} = g^{a_{i,k}}$ and $Y_{i,j} = g^{s_{i,j}}$ for $k \in \{0, \cdots, t\}$, and an encryption $E_{PK_j}(s_{i,j})$ for $1 \leq j \leq n$ of secret key share for entity ID_j under the correct *publicly verifiable encryption with proof of fairness* [FS01, CS03, ZI03]. ID_j will keep the share when $j = i$.

2. Each entity, $ID_1, \cdots, ID_n$, will make sure the distribution is correct by verifying $\prod_{k=0}^{t} A_{i,k}^{j^k} = \prod_{k=0}^{t} (a_{i,k})^{j^k} = g^{\sum_{k=0}^{t} j^k(a_{i,k})} = g^{f_i(j)}$ and checking if $g^{f_i(j)} = y_{i,j}$. The proofs that $E_{PK_j}(s_{i,j})$ is the correct encryption of $s_{i,j}$ to the public key PK_j are also verified.

3. The entities who do not follow the protocol will be disqualified. Let the set of

remaining entities be Λ. They can now generate the threshold key system with the public key $(y, y_1, \cdots, y_n)$ and secret shares $(x_1, \cdots, x_n)$ where entity ID_j will get correct shares from $i \in \Lambda$, and compute $x_j = \sum_{i \in \Lambda} s_{i,j}$. (i.e. The secret is $x = \sum_{i \in \Lambda} z_i \mod q$.)

In our scheme we need to reconstruct xP where $P \in \mathbb{G}_1$. Suppose Λ is the group of any t out of n signers, xP can be constructed by $xP = \sum_{j \in \Lambda} b_j x_j P$ where $b_j = \prod_{k \in \Lambda, k \neq j} \frac{k}{k-j}$.

7.1.4 Proactive Secret Sharing

Proactive security can be added to a polynomial-based threshold scheme easily [HJKY95]. Suppose x is the secret distributed by the polynomial $f_t(z)$ where $f_t(0) = x$, consider updating $f_t(z)$ by $f_{t+1}(z) = f_t(z) + g(z)$ where $g(z)$ is a polynomial with the same degree as $f_t(z)$ and $g(0) = 0$. By the linearity of the polynomial evaluation operation, we have $\forall i \quad f_{t+1}(i) = f_t(i) + g(i)$, i.e. each signer can use `DisKeyGen` to distributively "share" the new secret 0 and update his/her own secret x_j by $x_j = x_j + \sum_{i \in Q} s_{i,j}$.

7.2 Construction

Define $\mathbb{G}_1$, $\mathbb{G}_2$ and $\hat{e}(\cdot, \cdot)$ as in Chapter 4 and define H_1 and H_4 as in Chapter 5. We use $\mathcal{S}$ and $\mathcal{Q}$ to represent storage of secret parts and public parts respectively. For the notation about bit strings, we denote the binary representation of the time period j by $< j >$ and the i-th bit of $< j >$ by j_i.

Our construction consists of five main parts, namely, key generation, key evolution, proactive key update, signing and verification.

7.2.1 Key Generation

1. For $\zeta = \{\epsilon, 0, 00, \cdots, 0^{l-1}\}$, do

   (a) All signers run `DisKeyGen` to generate s_ζ. (Each signer i gets $s_\zeta^{(i)}$ as the share of s_ζ.)

(b) An arbitrary set of t signers reconstruct $Q_\zeta = s_\zeta P$ and sends to remaining signers.

(c) Each signer i computes $S^{(i)}_{(\zeta||0)} = S^{(i)}_\zeta + s^{(i)}_\zeta H_1(\zeta||0)$ and $S^{(i)}_{(\zeta||1)} = S^{(i)}_\zeta + s^{(i)}_\zeta H_1(\zeta||1)$.

(d) Each signer i stores $\mathcal{S}^{(i)}_{0^l} = \mathcal{S}^{(i)}_{0^l} \bigcup \{S^{(i)}_{(\zeta||1)}\}$, while $s^{(i)}_\zeta$ and $S^{(i)}_\zeta$ are deleted.

(e) Set $\mathcal{Q}_{\zeta||0} = \mathcal{Q}_{\zeta||1} = \mathcal{Q}_\zeta \bigcup \{Q_\zeta\}$.

2. All signers invoke `DisKeyGen` to generate s_{0^l}. (Each signer i gets $s^{(i)}_{0^l}$ as the share of s_{0^l}.)

3. An arbitrary set of t signers reconstruct $Q = s_{0^l} P$ and sends to the remaining $n - t$ signers.

4. The private key of each signer i is $SK^{(i)}_0 = \{s^{(i)}_{0^l}, S^{(i)}_{0^l}, \mathcal{S}^{(i)}_{0^l}, \mathcal{Q}_{0^l} \bigcup \{Q_{0^l}\}\}$ ($\mathcal{Q}_{0^l} \bigcup \{Q_{0^l}\}$ can be stored in non-secure storage as they can be revealed to the public.)

5. Public key of the system is $PK = \{P, Q\}$.

7.2.2 Key Evolution

All signers do the following at the end of the current time period j.

1. If $j = T - 1$, deletes $SK^{(i)}_j$ and sets $SK^{(i)}_j = \epsilon$ (an empty string), the algorithm terminates.

2. If $j_l = 0$, get $S_{<j+1>}$ from $\mathcal{S}_{<j>}$ and set $\mathcal{S}_{<j+1>} = \mathcal{S}_{<j>} - \{S_{<j+1>}\}$.

3. Else, suppose l' ($1 \leq l' \leq l$) is the maximum possible value that makes $j_{l'} = 0$.

(a) Let $n = j_0 j_1 \cdots j_{l'-1} 1$, where j_0 denotes ϵ for convenience (i.e. $< j+1 >= n0^{l-l'}$).

(b) Get S_n from $\mathcal{S}_{<j>}$ and set $\mathcal{S}_{<j+1>} = \mathcal{S}_{<j>} - \{S_n\}$.

(c) For $\zeta = \{n, n0, \cdots, n0^{l-l'-1})\}$, do

i. Invoke `DisKeyGen` to generate s_ζ. (Each signer i gets $s_\zeta^{(i)}$ as the share of s_ζ.)

ii. An arbitrary set of t signers reconstruct $Q_\zeta = s_\zeta P$ and sends to remaining signers.

iii. Each signer i computes $S_{(\zeta||0)}^{(i)} = S_\zeta^{(i)} + s_\zeta^{(i)} H_1(\zeta||0)$ and $S_{(\zeta||1)}^{(i)} = S_\zeta^{(i)} + s_\zeta^{(i)} H_1(\zeta||1)$.

iv. Each signer i stores $\mathcal{S}_{0^l}^{(i)} = \mathcal{S}_{0^l}^{(i)} \bigcup \{S_{(\zeta||1)}^{(i)}\}$, while $s_\zeta^{(i)}$ and $S_\zeta^{(i)}$ are deleted.

v. Set $\mathcal{Q}_{\zeta||0} = \mathcal{Q}_{\zeta||1} = \mathcal{Q}_\zeta \bigcup \{Q_\zeta\}$.

4. Invoke `DisKeyGen` to generate $s_{<j+1>}$. (Each signer i gets $s_{<j+1>}^{(i)}$ as the share of $s_{<j+1>}$.)

5. Any t signers reconstruct $Q_{<j+1>} = s_{<j+1>} P$ and sends to remaining signers.

6. If $j_l = 0$, set $\mathcal{Q}_{<j+1>} = \mathcal{Q}_{<j>} - Q_{<j>}$. (If $j_l = 1$, $\mathcal{Q}_{<j+1>}$ has been constructed already.)

7. Set $SK_{j+1}^{(i)} = \{s_{<j+1>}^{(i)}, S_{<j+1>}^{(i)}, \mathcal{S}_{<j+1>}^{(i)}, \mathcal{Q}_{<j+1>} \bigcup \{Q_{<j+1>}\}\}$ and delete $SK_j^{(i)}$.

7.2.3 Proactive Key Update

At the time period j, each signer i invokes `DisKeyGen` with $z_i = 0$ to get x_j, and proactively update their shares by $s_{<j>}^{(i)} = s_{<j>}^{(i)} + x_j$.

7.2.4 Signing

Suppose Λ is the group of any t out of n signers. At the time period j, each signer of Λ compute the partial signature of message m by $V^{(i)} = b_i(S_{<j>}^{(i)} + s_{<j>}^{(i)} H_4(i_1 i_2 \cdots i_l||m))$, where $b_i = \prod_{1 \le k \le t, k \ne i} \frac{k}{k-i}$. Then, any party can construct the signature as $\{j, V = \sum_{i \in \Lambda} V^{(i)}, \mathcal{Q}_{<j>}\}$.

7.2.5 Verification

At the time period j, any party can verify the validity of the signature $\{j, V, \mathcal{Q}\}$ by verifying whether

$$\hat{e}(P,V)/\prod_{r=2}^{l}\hat{e}(Q_{j_1j_2\cdots j_{r-1}}, H_1(j_1j_2\cdots j_r)) = \hat{e}(Q, H_1(j_1))\hat{e}(Q_{<j>}, H_4(j_1j_2\cdots j_l||m)).$$

If equality holds then return $\top$, else return $\perp$. It is easy to see that the verification works. For any valid signature produced by our forward-secure threshold signature scheme:

$$\begin{aligned}
L.H.S. &= \hat{e}(P,V)/\prod_{r=2}^{l}\hat{e}(Q_{j_1j_2\cdots j_{r-1}}, H_1(j_1j_2\cdots j_r)) \\
&= \frac{\hat{e}(P, s_\epsilon H_1(j_1) + \sum_{r=2}^{l} s_{j_1j_2\cdots j_{r-1}} H_1(j_1j_2\cdots j_r) + s_{<j>}H_4(j_1j_2\cdots j_r||m))}{\prod_{r=2}^{l}\hat{e}(P, H_1(j_1j_2\cdots j_r))^{s_{j_1j_2\cdots j_{r-1}}}} \\
&= \frac{\hat{e}(P, s_\epsilon H_1(j_1)) \cdot \prod_{r=2}^{l}\hat{e}(P, s_{j_1j_2\cdots j_{r-1}} H_1(j_1j_2\cdots j_r)) \cdot \hat{e}(P, s_{<j>}H_4(j_1j_2\cdots j_r||m))}{\prod_{r=2}^{l}\hat{e}(P, s_{j_1j_2\cdots j_{r-1}} H_1(j_1j_2\cdots j_r)))} \\
&= \hat{e}(P, s_\epsilon H_1(j_1)) \cdot \hat{e}(P, s_{<j>}H_4(j_1j_2\cdots j_r||m)) \\
&= \hat{e}(Q, H_1(j_1)) \cdot \hat{e}(Q_{<j>}, H_4(j_1j_2\cdots j_r||m)) = R.H.S.
\end{aligned}$$

7.3 Improving the Efficiency

Now we give the description of the forward-secure threshold signature scheme using all the nodes of the tree of hierarchy. Define $\mathbb{G}_1$, $\mathbb{G}_2$ and $\hat{e}(\cdot,\cdot)$ as in Chapter 4. Let H_5, H_6 and H_7 be three cryptographic hash functions where $H_5 : \{0,1\}^* \times \mathbb{G}_1 \rightarrow \mathbb{G}_1$, $H_6 : \{0,1\}^* \times \{0,1\}^* \times \mathbb{G}_1 \rightarrow \mathbb{G}_1$ and $H_7 : \{0,1\}^* \times \mathbb{G}_1 \rightarrow \mathbb{Z}_q^*$. For the notation about bit strings, we use w to denote the name of the node corresponding to the current time period, $|w|$ to denote the bit length of w and $w|_i$ to denote the first i bits of w. To ease the description, we use a stack (a last-in-first-out data structure) to hold the secret keys generated. In actual implementation, the extra spaces incurred from the duplication of keys in different entries of the stack can be minimized by using an extra level of redirection such as a pointer pointing to the actual storage of the key.

Our construction consists of five main parts, namely, key generation, key evolution, proactive key update, signing and verification.

7.3.1 Key Generation

1. All signers run `DisKeyGen` to generate s_ϵ.

   (Each signer i gets $s_\epsilon^{(i)}$ as the share of s_ϵ.)

2. An arbitrary set of t signers reconstruct $Q_\epsilon = s_\epsilon P$ and sends to remaining signers.

3. Each signer i computes $S_\epsilon^{(i)} = s_\epsilon^{(i)} H_5(\epsilon, Q_\epsilon)$

4. The private key of each signer i is $SK_\epsilon^{(i)} = \{Q_\epsilon, S_\epsilon^{(i)}\}$.

5. Push the private key onto the stack.

6. Public key of the system is $PK = \{P, Q_\epsilon\}$.

7.3.2 Key Evolution

All signers do the following at the end of the time period to evolute the private key shares.

1. Pops SK_w off the stack.

2. If w is an internal node,

   (a) Invoke `DisKeyGen` to generate s_{w0} and s_{w1}.

   (Each signer i gets $s_{w0}^{(i)}$ as the share of s_{w0} and $s_{w1}^{(i)}$ as the share of s_{w1}.)

   (b) An arbitrary set of t signers reconstruct $Q_{w0} = s_{w0}P$ and $Q_{w1} = s_{w1}P$, then sends these to remaining signers.

   (c) Compute $h_{w0} = H_6(w0, Q_{w0})$ and $h_{w1} = H_6(w1, Q_{w1})$.

   (d) Compute $S_{w0}^{(i)} = S_w^{(i)} + h_{w0} s_{w0}^{(i)} H_5(\epsilon, Q_\epsilon)$ and $S_{w1}^{(i)} = S_w^{(i)} + h_{w1} s_{w1}^{(i)} H_5(\epsilon, Q_\epsilon)$.

   (e) Delete $S_w^{(i)}$ securely.

   (f) Push $SK_{w1}^{(i)} = \{Q_{w|_1}, \cdots, Q_{w|_{|w|-1}}, Q_w, Q_{w||1}, S_{w||1}\}$ onto the stack.

   (g) Push $SK_{w0}^{(i)} = \{Q_{w|_1}, \cdots, Q_{w|_{|w|-1}}, Q_w, Q_{w||0}, S_{w||0}\}$ onto the stack.

3. If w is a leaf, each signer simply deletes $S_w^{(i)}$ securely.

4. Invoke `DisKeyGen` to generate r_w.

   (Each signer i gets $r_w^{(i)}$ as the share of r_w.)

5. Reconstruct $U_w = r_w P$.

7.3.3 Proactive Key Update

Each signer i invokes `DisKeyGen` with $z_i = 0$ to get ${s'}_w^{(i)}$, and proactively update their shares by ${S'}_w^{(i)} = S_w^{(i)} + h_w {s'}_w^{(i)} H_5(\epsilon, Q_\epsilon)$.

7.3.4 Signing

Suppose Λ is the group of any t out of n signers. At the time period j, each signer of Λ compute the partial signature of message m as follows.

1. Compute $P_m = H_7(m, j, U_w)$.

2. Compute the partial signature by $V^{(i)} = b_i(S_w^{(i)} + r_w^{(i)} P_m)$ where $b_i = \prod_{1 \le k \le t, k \ne i} \frac{k}{k-i}$.

Then, any party can produce the signature $\{j, V = \sum_{i \in \Lambda} V^{(i)}, \mathcal{Q}_w = \{Q_{w|_1}, \cdots, Q_{w|_{|w|}}, U_w\}\}$.

7.3.5 Verification

At the time period j, any party can verify the validity of the signature $\{j, V, \mathcal{Q}\}$ as follows.

1. Reconstruct ${h'}_{w|_k} = H_6(w|_k, Q_{w|_k})$ for $k \in \{1, 2, \cdots, |w|_\}$.

2. Reconstruct ${P'}_m = H_7(m, j, U_w)$.

3. Verify whether

$$\hat{e}(P, V) = \hat{e}(Q_\epsilon + \sum_{k=1}^{|w|} {h'}_{w|_k} Q_{w|_k}, H_5(\epsilon, Q_\epsilon)) \hat{e}(U_w, {P'}_m)$$

If equality holds then return $\top$, else return $\perp$.

7.4 Against Adaptive Adversary

As suggested in [ZI03], multiple uses of `DisKeyGen` cannot achieve security against an adaptive adversary as homomorphic encryption is used [ZI03]. However, this problem can be fixed if independent parameters are chosen at each invocation or using other adaptive chosen-ciphertext secure publicly verifiable encryption with proof of fairness.

With the help of `DisKeyGen`, each party will not contribute an inconsistent share. But in the proactive key update procedure, the secret key will be spoiled if z_i from one party is not equal to 0. To address this problem, we can simply check whether $y_0 = g$ when using `DisKeyGen` during the proactive key update.

7.5 Analysis

Again we analyze our second scheme's consistency, security and efficiency.

7.5.1 Consistency

We first prove the correctness of the scheme. For any valid signature produced by our forward-secure threshold signature scheme:

$$\begin{aligned}
& \hat{e}(Q_\epsilon + \sum_{k=1}^{|w|} h_{w|_k} Q_{w|_k}, H_5(\epsilon, Q_\epsilon))\hat{e}(U_w, P_m) \\
= \; & \hat{e}(s_\epsilon P + \sum_{k=1}^{|w|} h_{w|_k} s_{w|_k} P, H_5(\epsilon, Q_\epsilon))\hat{e}(r_w P, P_m) \\
= \; & \hat{e}(P, (s_\epsilon + \sum_{k=1}^{|w|} h_{w|_k} s_{w|_k}) H_5(\epsilon, Q_\epsilon))\hat{e}(P, r_w P_m) \\
= \; & \hat{e}(P, (s_\epsilon + \sum_{k=1}^{|w|} h_{w|_k} s_{w|_k}) H_5(\epsilon, Q_\epsilon) + r_w P_m) \\
= \; & \hat{e}(P, V)
\end{aligned}$$

7.5.2 Security

The security of our scheme is based on the security of [HJKY95, KPH04, ZI03], which relies on the intractabilities of the GDH problem and the discrete logarithm problem.

Theorem 4. *Let FS-DS denote the single-user signature scheme in [KPH04]. Our proposed scheme is a forward-secure threshold signature scheme secure against adaptive chosen message attack as long as FS-DS is a forward-secure signature scheme in the single-user sense.*

Proof. Let $\mathcal{A}$ be the adversary who control up to $t-1$ signers during execution of our scheme before the j^{th} time period and control up to t signers (i.e. knowing the secret key of the system) at the j^{th} time period. $\mathcal{A}$ is allowed to launch chosen-message attack, i.e. it can obtain valid signatures for message $m_1, m_2, \cdots$ on its wishes. If $\mathcal{A}$ can produce with non-negligible probability a valid signature for an un-queried message m, (i.e. $m \neq m_i$ for $i \geq 1$) for time period j', where $j' < j$, we can construct a forger $\mathcal{F}$ to forge a signature of FS-DS using the procedure $\mathcal{A}$ and the signing oracle $\mathcal{O}_{sig}$ of FS-DS.

Let (P, Q) be the public key of FS-DS. We set (P, Q) to be the public key of our (simulated) scheme. As $\mathcal{F}$ does not know the corresponding secret key of (P, Q), $\mathcal{F}$ can only assign each signer ID_i a random secret share $x_j \in \mathbb{Z}_q$ during the key generation phase of our proposed scheme. Then it runs the simulator proposed in the Theorem 2 of [FS01] and the simulator of the corresponding publicly verifiable encryption with proof of fairness to produce a transcript of communication between signers during the invocation of `DisKeyGen`, with right distribution.

For signing requests issued by $\mathcal{A}$, $\mathcal{F}$ simulates our scheme by simulating all signers as follows. When the adversary $\mathcal{A}$ requests for a signature for m_i, $\mathcal{F}$ queries $\mathcal{O}_{sig}$ to obtain a signature $\{j, V, \mathcal{Q}\}$. Then it runs the simulator of our proposed scheme with input m_i and $\{j, V, \mathcal{Q}\}$ to produce the transcript of communication for $\mathcal{A}$. (by using the simulator proposed in the Theorem 2 of [FS01], the simulator of the corresponding publicly verifiable encryption with proof of fairness and the simulation proposed in the proof of Theorem 2 in [KPH04]). Therefore, $\mathcal{F}$ provided a faithful simulation of our proposed scheme to $\mathcal{A}$.

At the j^{th} time period we provide the correct secret shares of t signers to $\mathcal{A}$ by choosing j as the break-in period of FS-DS. On input of these correct shares and pre-

vious transcripts, $\mathcal{A}$ finally produce a valid signature $\{j', V', \mathcal{Q}'\}$ for a new message m, $m \neq m_i$ at time period j, where $j' < j$, then $\{j', V', \mathcal{Q}\}$ is the forged signature for FS-DS. Thus, FS-DS is not unforgeable under the chosen message attack, which is a contradiction.

□

7.5.3 Requirement on the Network

Synchronous network seems to be a strong requirement for MANET. Our scheme can be easily modified to work in asynchronous network by assuming there exists an incorruptible third party (ITP), which is not trusted but only honest. We refer the reader to [FS01] for a discussion of how ITP can help in making the protocol secure even in the asynchronous network.

Moreover, our scheme does not rely on the existence of secure channel with the help of publicly verifiable encryption with proof of fairness This requirement is not as strong as one may think because a variety of choices are available (e.g. the schemes in [FS01, CS03, ZI03]), so all participants are not required to use the same algorithm for publicly verifiable encryption with proof of fairness.

7.5.4 Efficiency

Table 7.2 shows a summary of the efficiency of our proposed scheme. The following shows the types of operations that we consider.

1. *Pairing (Pa)*: The total number of pairing computations required.

2. *Point Multiplication (Mu)*: The total number of point multiplications required.

We compare the performance of our scheme with the most efficient one [CLT03] of the existing schemes. Regarding key update, our scheme runs in $O(\log T)$ time while [CLT03] runs in $O(\log^5 T)$ time where T is the total number of time periods. On the other hand, our scheme does not require any interaction between the signers in signing while [CLT03] requires quite a number of communication rounds in signing.

Algorithm	Computation		Interaction
	Mu	Pa	Rounds
Key Generation	$O(1)$	0	1
Key Evolution	$O(1)$	0	2
Proactive Key Update	$O(1)$	0	1
Signing	$O(1)$	0	0
Verification[5]	$O(\log T)$	3	0

Table 7.2: *Computational Efficiency of Our Forward-Secure Threshold Scheme*

7.6 Chapter Summary

Table 7.3 shows a summary of comparing our scheme with other existing schemes in terms of the maximum number of rounds required in signing or key update. We use AMN, TT, CLT and SCFTS to denote the scheme in [AMN01], [TT01], [CLT03] and our proposed scheme respectively.

Scheme	Maximum Number of Rounds in Signing or Key Update[6]
AMN	$O(\log l)$
TT	$O(l)$
CLT	10
SCFTS	2

Table 7.3: *Comparison on Efficiency of Existing Forward-Secure Threshold Schemes*

Obviously, existing forward-secure threshold signature schemes requires quite a few number of communication rounds among the signers in either the signing or the key update process. Thus, these schemes are not useful in situation such as an ad-hoc network. In this chapter, we propose a forward-secure threshold signature scheme which is more

[5] T is the total number of time period supported.
[6] l is the number of bits in the hash function's output, typically 160.

efficient in key updates and supports proactive mechanism. Our scheme can work in asynchronous network without secure channel and is round-optimal in signing, which is especially useful in situation like mobile ad-hoc network.

□ **End of chapter.**

Chapter 8

Conclusion

8.1 Review of Results

1. We proposed a new paradigm for identity-based signcryption, in which the private signcryption key and the private decryption key are separated. [Chapter 3]

2. We proposed an identity-based signcryption scheme that satisfies the security requirements of forward security, public verifiability and public ciphertext authenticity. [Chapter 6]

3. Our identity-based signcryption scheme closed the open problem previously proposed by Libert and Quisquater [LQ03]. [Chapter 6] (In May 2004, Noel McCullagh and Paulo S. L. M. Barreto claimed that their paper "Efficient and Forward-Secure Identity-Based Signcryption" were the first one closing the open problem proposed by [LQ03]; however, the scheme in this thesis was published at The Sixth International Conference of Information Security and Cryptology - ICISC 2003 held at Seoul, Korea on November 27-28, 2003.)

4. We proposed variants of Bilinear Diffie-Hellman (BDH) problems, namely, Modified Decisional Bilinear Diffie-Hellman Problem (MDBDHP) and Modified Computational Bilinear Diffie-Hellman Problem (MDBCHP). [Chapter 4]

5. Our identity-based signcryption scheme can be shown to be provably secure under

the random oracle model, with the assumption that variants of the BDH problems are hard to compute. [Chapter 6]

6. We proposed a new forward-secure threshold signature scheme that is more efficient in key updates when compared with the previous work. The scheme can work in asynchronous network without secure channel and is round-optimal in signing, which is especially useful in situation like mobile ad-hoc network. [Chapter 7]

7. Our forward-secutre threshold signature scheme is secure against adaptive chosen message attack as long as the GDH problem and the discrete logarithm problem are hard to solve. [Chapter 7]

8.2 Extensions of Results from Former Papers

This thesis describes the major results in the following papers:

1. Sherman S.M. Chow, S.M. Yiu, Lucas C.K. Hui, and K.P. Chow. Efficient Forward and Provably Secure ID-Based Signcryption Scheme with Public Verifiability and Public Ciphertext Authenticity. In Jong In Lim and Dong Hoon Lee, editors, *The Sixth Annual International Conference on Information Security and Cryptology (ICISC 2003) (Acceptance Rate: 32/163 = 19.6%)*, volume 2971 of *Lecture Notes in Computer Science*, pages 352–369, Seoul, Korea, 2004. Springer-Verlag.

2. Sherman S.M. Chow, H.W. Go, Lucas C.K. Hui, S.M. Yiu, and K.P. Chow. Two Forward-Secure Threshold Signature Schemes. In *The Second International Conference of Applied Cryptography and Network Security (ACNS 2004), Yellow Mountain, China, Technical Track Proceedings, a special issue of Journal of Information Security and Data Confidentiality (Acceptance Rate: 87/297 = 29.29%)*, pages 10–19, 2004.

with the following extensions:

1. Analysis of "Receipt Anonymity" for the forward-secure identity-based signcryption scheme is added. [Chapter 6]

2. More efficient forward-secure threshold signature scheme (constant number of pairing operations) is proposed. [Chapter 7]

8.3 Open Problems and Future Research

We have considered the forward security for the confidentiality of identity-based signcryption, but not yet the forward security for the unforgeability. One possible future research direction is to devise a forward-secure identity-based signcryption in both senses. Other directions include making identity-based signcryption work in a hierarchical setting, possibly enabled by a new identity-based hierarchical signature; We can also try to improve its security, possibly by the derivation of a new identity-based signature with exact security first.

We leave it as an open question to devise a more efficient forward-secure threshold signature scheme, and also other forward-secure schemes such as forward-secure blind signature and forward-secure multisignature.

Bilinear pairings provide a very "rich structure" for the construction of cryptographic schemes, it is anticipated that the surge in pairing-based cryptography in these years will continue. Some interesting notions in pairing-based cryptography or ID-based cryptography includes ID-based partially blind signature, ID-based threshold ring signature, efficient ID-based ring signature, ID-based multi designated verifiers signature, anonymous identity-based private key issuing without secure channel, etc. Adding more "functionalities" to the pairings is also a promising direction, for example, making the result of pairings with private key as input to be publicly verifiable.

Moreover, we believe that MDBDHP and MCBDHP are interesting in their own rights. In this thesis, we see that these two newly proposed problems give raise to new cryptosystems in which the private signcryption key and the private decryption key are separated. We expected there are many other useful applications.

List of Abbreviations

Abbreviation	Details
ACM	Association for Computing Machinery
BC	Before Christ
BDH	Bilinear Diffie-Hellman
CA	Certificate Authority
CBDHP	Computational Bilinear Diffie-Hellman Problem
CCA2	Adaptive Chosen-Ciphertext Attack
CMA	Chosen-Message Attack
CMA2	Adaptive Chosen-Message Attack
DBDHP	Decisional Bilinear Diffie-Hellman Problem
DDH	Decisional Diffie-Hellman
DLP	Discrete Logarithm Problem
DSA	Digital Signature Algorithm
DSS	Digital Signature Standard
ECDLP	Elliptic Curve Discrete Logarithm Problem
EUF-IDSC2-CCIA2	Existential UnForgeability of IDentity-based SignCryptions of 2 keys under Adaptive Chosen-Ciphertext-and-Identity Attacks
Ex	Exponentiations
FS-DS	Forward Secure Digital Signature
FwSec	Forward Security
GDH	Gap Diffie-Hellman
GQ	Guillou-Quisquater
GSM	Global System for Mobile communication
HIBS	Hierarchical Identity Based Signature
IBE	Identity Based Encryption
IND-IDSC2-CCIA2	INDistinguishability of IDentity-based SignCryptions of 2 keys under Adaptive Chosen-Ciphertext-and-Identity Attacks
ID	IDentity
ITP	Incorruptible Third Party
KeyGen	Key Generation

Abbreviation	Details
LHS	Left Hand Side
MANET	Mobile Ad-hoc NETwork
MCBDH	Modified Computational Bilinear Diffie-Hellman
MCBDHP	Modified Computational Bilinear Diffie-Hellman Problem
MDBDH	Modified Decisional Bilinear Diffie-Hellman
MDBDHP	Modified Decisional Bilinear Diffie-Hellman Problem
MPhil	Master of Philosophy
Mu	Point Multiplication
OAEP	Optimal Asymmetric Encryption Padding
OWF	One Way Function
Pa	Pairing
PDF	Portable Document Format
PKCS	Public Key Cryptography Standards
PKG	Private Key Generator
PKI	Public Key Infrastructure
ProvSec	Provable Security
PS	Post Script
PubCAuth	Public Ciphertext Authenticity
PubVer	Public Verifiability
PVE	Publicly Verifiable Encryption
PVSS	Publicly Verifiable Secret Sharing
REA-IDSC2-CCIA2	REcipient Anonymity of IDentity-based SignCryptions of 2 keys under Adaptive Chosen-Ciphertext-and-Identity Attacks
RHS	Right Hand Side
RSA	Rivest-Shamir-Adleman
SEM	SEcurity Mediator
TP	Third Party
VSS	Verifiable Secret Sharing

Curriculum Vitae

SHERMAN S.M. CHOW
CURRICULUM VITAE

Address

Department of Computer Science
Faculty of Engineering
The University of Hong Kong
Pokfulam, Hong Kong
http://www.cs.hku.hk/~smchow
smchow@cs.hku.hk

Research Interests

Identity Based Cryptography, Pairing Based Cryptography, Applied Cryptography

Education

2002 - 2004 Master of Philosophy in Computer Science
Department of Computer Science, The University of Hong Kong
Thesis Title:
Forward Security from Bilinear Pairings: Signcryption and Threshold Signature
Academic Advisor: Dr. Lucas C.K. Hui, Dr. K.P. Chow

Graduate Coursework:
Advanced Database Technologies (Dr. Nikos Mamoulis)
Analysis and Design of Enterprise Applications in UML (Dr. T.H. Tse)
Computer and Network Security (Dr. Russell S.W. Yiu and Dr. S.M. Yiu)

1999 - 2002 First Class Honours in Computer Engineering
Department of Computer Science and Information Systems,
and Department of Electrical and Electronic Engineering
The University of Hong Kong
Thesis Title: *Spyware and Internet Anonymity*
Academic Advisor: Dr. K.P. Chow

Selected Academic Honors and Awards

2002 - 2004 Postgraduate Studentship from The University of Hong Kong

2001 - 2002 Hong Kong Chiu Chow Chamber of Commerce Scholarship

2001 - 2002 Dean's Honours List

2000 - 2001 Dean's Honours List

Publications (derived from this thesis and from undergraduate work)

1. Sherman S.M. Chow, S.M. Yiu, Lucas C.K. Hui, and K.P. Chow. Efficient Forward and Provably Secure ID-Based Signcryption Scheme with Public Verifiability and Public Ciphertext Authenticity. In Jong In Lim and Dong Hoon Lee, editors, *The Sixth Annual International Conference on Information Security and Cryptology (ICISC 2003) (Acceptance Rate: 32/163 = 19.6%)*, volume 2971 of *Lecture Notes in Computer Science*, pages 352–369, Seoul, Korea, 2004. Springer-Verlag.
2. Sherman S.M. Chow, H.W. Go, Lucas C.K. Hui, S.M. Yiu, and K.P. Chow. Two Forward-Secure Threshold Signature Schemes. In *The Second International Conference of Applied Cryptography and Network Security (ACNS 2004), Yellow Mountain, China, Technical Track Proceedings, a special issue of Journal of Information Security and Data Confidentiality (Acceptance Rate: 87/297 = 29.29%)*, pages 10–19, 2004.
3. Sherman S.M. Chow, Lucas C.K. Hui, S.M. Yiu, and K.P. Chow. Forward-Secure Multisignature and Blind Signature Schemes. *Applied Mathematics and Computation*, Accepted, September 2004. To Appear.
4. Sherman S.M. Chow, H.W. Go, and Ricky W.M. Tang. Impact of Recent Advances in Cryptography on Online Game. In *The Third International Conference on Application and Development of Computer Games, Hong Kong, April 26-27, 2004*, pages 68–73, 2004.
5. Sherman S.M. Chow, Lucas C.K. Hui, S.M. Yiu, K.P. Chow, and Richard W.C. Lui. A Generic Anti-Spyware Solution by Access Control List at Kernel Level. *Journal of Systems and Software*, 75(2):227–234, 2004. A preliminary version appeared in 5th ACM Postgraduate Research Day, January 31, 2004, pp. 307-312.

Related Working / Teaching Experience

Fall 2003 Teaching Assistant for the graduate course "Computer and Network Security"

Fall 2002 Teaching Assistant for the course "Database Management System"

Summer 2001 Research Assistant, Center of Information Security and Cryptography

Technical Certifications

2004 Sun Certified Programmer for the Java 2 Platform 1.4

2003 IBM Certified Associate Developer (WebSphere Studio Application Developer, V5.0)

2003 IBM Certified Solutions Expert (DB2 UDB v7.1 Database Administrator)

2003 IBM Certified Specialist (DB2 UDB v6.1/v7.1 User)

Technical Skills

Computer Languages:	Java (includes JCE, RMI, JDBC, JNI, JMF, GUI), C/C++, VB, C#, XML
Web Technologies:	ASP/ASP.NET, JSP/Servlet, PHP, web services, HTML, CSS, JavaScript
Database Technologies:	Oracle, mySQL, SQL
Programming Experiences:	Secure programming, cryptographic application, Unix
Software:	VS.NET, Rational Rose, NUnit, Flash, Photoshop, Office, Latex

Languages Spoken

- Cantonese, Chinese - native
- English - advanced
- Mandarin - intermediate
- Japanese - basic

Bibliography

[AAB+98] Hal Abelson, Ross Anderson, Steven M. Bellovin, Josh Benaloh, Matt Blaze, Whitfield Diffie, John Gilmore, Peter G. Neumann, Ronald L. Rivest, Jeffrey I. Schiller, and Bruce Schneier. The Risks of Key Recovery, Key Escrow, and Trusted Third-Party Encryption. Technical report, An Ad Hoc Group of Cryptographers and Computer Scientists, 1998. Available at http://www.cdt.org/crypto/risks98.

[AB83] Charles Asmuth and John Bloom. A Modular Approach to Key Safeguarding. *IEEE Transactions on Information Theory*, IT-29(2):208–211, 1983.

[ADR02] Jee Hea An, Yevgeniy Dodis, and Tal Rabin. On the Security of Joint Signature and Encryption. In Lars R. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002, International Conference on the Theory and Applications of Cryptographic Techniques, Amsterdam, The Netherlands, April 28 - May 2, 2002, Proceedings*, volume 2332 of *Lecture Notes in Computer Science*, pages 83–107. Springer-Verlag Heidelberg, 2002.

[AG00] N. Asokan and Philip Ginzboorg. Key-Agreement in Ad-Hoc Networks. *Computer Communications*, 23(17):1627–1637, 2000.

[AI03] Mohamed Al-Ibrahim. A Signcryption Scheme Based on Secret Sharing Technique. In Vladimir Gorodetsky, Leonard J. Popyack, and Victor A. Skormin, editors, *Computer Network Security, Second International Workshop on Mathematical Methods, Models, and Architectures for Computer Network Security, MMM-ACNS 2003, St. Petersburg, Russia, September 21-23, 2003, Proceedings*, volume 2776 of *Lecture Notes in Computer Science*, pages 279–288. Springer, 2003.

[AMN01] Michel Abdalla, Sara Miner, and Chanathip Namprempre. Forward-Secure Threshold Signature Schemes. In David Naccache, editor, *Topics in Cryptology - CT-RSA 2002, The Cryptographer's Track at the RSA*

Conference, 2002, San Jose, CA, USA, February 18-22, 2002, Proceedings, volume 2020 of *Lecture Notes in Computer Science*, pages 441–456. Springer, 2001.

[An01] Jee Hea An. Authenticated Encryption in the Public-Key Setting: Security Notions and Analyses. Cryptology ePrint Archive, Report 2001/079, September 2001. Available at http://eprint.iacr.org.

[And97] Ross Anderson. Two Remarks on Public Key Cryptology. Fourth ACM Conference on Computer and Communications Security, 1997. Invited Talk.

[AR00] Michel Abdalla and Leonid Reyzin. A New Forward-Secure Digital Signature Scheme. In Tatsuaki Okamoto, editor, *Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings*, volume 1976 of *Lecture Notes in Computer Science*, pages 116–129. Springer, 2000.

[BBDW96] Simon R. Blackburn, Mike Burmester, Yvo Desmedt, and Peter R. Wild. Efficient Multiplicative Sharing Schemes. In Ueli M. Maurer, editor, *Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, Spain, May 12-16, 1996, Proceeding*, volume 1070 of *Lecture Notes in Computer Science*, pages 107–118. Springer, 1996.

[BCKM01] Mike Burmester, Vassilios Chrissikopoulos, Panayiotis Kotzanikolaou, and Emmanouil Magkos. Strong Forward Security. In *Proceedings of the 16th International Conference on Information Security: Trusted Information*, pages 109–121, 2001.

[BD98] Feng Bao and Robert H. Deng. A Signcryption Scheme with Signature Directly Verifiable by Public Key. In Hideki Imai and Yuliang Zheng, editors, *Public Key Cryptography, First International Workshop on Practice and Theory in Public Key Cryptography, PKC '98, Pacifico Yokohama, Japan, February 5-6, 1998, Proceedings*, volume 1431 of *Lecture Notes in Computer Science*, pages 55–59. Springer, 1998.

[BDPR98] Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. Relations Among Notions of Security for Public-Key Encryption Schemes. In Hugo Krawczyk, editor, *Advances in Cryptology - CRYPTO '98, 18th Annual International Cryptology Conference, Santa Barbara, California, USA, August 23-27, 1998, Proceedings*, volume 1462 of *Lecture Notes in Computer Science*, pages 26–45. Springer, 1998.

[BDTW01] Dan Boneh, Xuhua Ding, Gene Tsudik, and Chi Ming Wong. Instantaneous Revocation of Security Capabilities. In *Proceedings of the 10th*

USENIX Security Symposium (SECURITY-01). The USENIX Association, August 13–17 2001.

[BDZ03] Feng Bao, Robert H. Deng, and Huafei Zhu. Variations of Diffie-Hellman Problem. In Dieter Qing, Dieter Gollmann, and Jianying Zhou, editors, *Information and Communications Security, 5th International Conference, ICICS 2003, Inner-Mongolia 10-13 October, 2003, Proceedings*, volume 2836 of *Lecture Notes in Computer Science*. Springer, 2003.

[Bel00] Mihir Bellare, editor. *On the Exact Security of Full Domain Hash*, volume 1880 of *Lecture Notes in Computer Science*. Springer, 2000.

[BF97] Dan Boneh and Matthew K. Franklin. Efficient Generation of Shared RSA Keys (Extended Abstract). In Burton S. Kaliski Jr., editor, *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, volume 1294 of *Lecture Notes in Computer Science*, pages 425–439. Springer, 1997.

[BF01] Dan Boneh and Matt Franklin. Identity-Based Encryption from the Weil Pairing. In Joe Kilian, editor, *Advances in Cryptology - CRYPTO 2001, 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19-23, 2001, Proceedings*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer-Verlag Heidelberg, 2001.

[BGLS03] Dan Boneh, Craig Gentry, Ben Lynn, and Hovav Shacham. Aggregate and Verifiably Encrypted Signatures from Bilinear Maps. In Eli Biham, editor, *Advances in Cryptology - EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4-8, 2003, Proceedings*, volume 2656 of *Lecture Notes in Computer Science*, pages 416–432. Springer, 2003.

[BHBR01] Stefano Basagni, Kris Herrin, Danilo Bruschi, and Emilia Rosti. Secure Pebblenets. In *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 156–163. ACM Press, 2001.

[BKLS02] Paulo S.L.M. Barreto, Hae Y. Kim, Ben Lynn, and Michael Scott. Efficient Algorithms for Pairing-Based Cryptosystems. In Moti Yung, editor, *Advances in Cryptology: Proceedings of CRYPTO 2002 22nd Annual International Cryptology Conference Santa Barbara, California, USA, August 18-22, 2002*, volume 2442 of *Lecture Notes in Computer Science*, pages 354–368. Springer-Verlag Heidelberg, 2002.

[BL90] Josh Cohen Benaloh and Jerry Leichter. Generalized Secret Sharing and Monotone Functions. In Shafi Goldwasser, editor, *Advances in Cryptology - CRYPTO '88, 8th Annual International Cryptology Conference,*

Santa Barbara, California, USA, August 21-25, 1988, Proceedings, volume 403 of *Lecture Notes in Computer Science*, pages 27–35. Springer, 1990.

[Bla79] G. R. Blakley. Safeguarding Cryptographic Keys. In Richard E. Merwin, Jacqueline T. Zanca, and Merlin. Smith, editors, *Proceedings of the AFIPS 1979 National Computer Conference: June 4–7, 1979, New York, New York*, volume 48 of *AFIPS Conference proceedings*, pages 313–317, Montvale, NJ, USA, 1979. AFIPS Press.

[BLS01] Dan Boneh, Ben Lynn, and Hovav Shacham. Short Signatures from the Weil Pairing. In Colin Boyd, editor, *Advances in Cryptology - ASIACRYPT 2001, 7th International Conference on the Theory and Application of Cryptology and Information Security, Gold Coast, Australia, December 9-13, 2001, Proceedings*, volume 2248 of *Lecture Notes in Computer Science*, pages 514–532. Springer, 2001.

[BM99] Mihir Bellare and Sara K. Miner. A Forward-Secure Digital Signature Scheme. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 431–448. Springer, 1999.

[Boy03a] Colin Boyd. *Protocols for Authentication and Key Establishment*. Springer-Verlag, 2003.

[Boy03b] Xavier Boyen. Multipurpose Identity-Based Signcryption : A Swiss Army Knife for Identity-Based Cryptography. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, volume 2729 of *Lecture Notes in Computer Science*, pages 382–398. Springer, 2003.

[BR93] Mihir Bellare and Phillip Rogaway. Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In *The First ACM Conference on Computer and Communications Security*, pages 62–73, 1993.

[BR95] Mihir Bellare and Phillip Rogaway. Optimal Asymmetric Encryption. In Alfredo De Santis, editor, *Advances in Cryptology - EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 9-12, 1994, Proceedings*, volume 950 of *Lecture Notes in Computer Science*, pages 92–111. Springer, 1995.

[BR96] Mihir Bellare and Phillip Rogaway. The Exact Security of Digital Signatures - How to Sign with RSA and Rabin. In Ueli M. Maurer, editor, *Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa,*

Spain, May 12-16, 1996, Proceeding, volume 1070 of *Lecture Notes in Computer Science*, pages 399–416. Springer, 1996.

[BSS02] Emmanuel Bresson, Jacques Stern, and Michael Szydlo. Threshold Ring Signatures and Applications to Ad-hoc Groups. In Moti Yung, editor, *Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings*, volume 2442 of *Lecture Notes in Computer Science*, pages 465–480. Springer, 2002.

[BSZ02] Joonsang Baek, Ron Steinfeld, and Yuliang Zheng. Formal Proofs for the Security of Signcryption. In David Naccache and Pascal Paillier, editors, *Public Key Cryptography, 5th International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 2002, Paris, France, February 12-14, 2002, Proceedings*, volume 2274 of *Lecture Notes in Computer Science*, pages 80–98. Springer, 2002.

[BY03] Mihir Bellare and Bennet S. Yee. Forward-Security in Private-Key Cryptography. In Marc Joye, editor, *Topics in Cryptology - CT-RSA 2003, The Cryptographers' Track at the RSA Conference 2003, San Francisco, CA, USA, April 13-17, 2003, Proceedings*, volume 2612 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2003.

[CC02] Jae Choon Cha and Jung Hee Cheon. An Identity-Based Signature from Gap Diffie-Hellman Groups . In Yvo Desmedt, editor, *Public Key Cryptography - PKC 2003, 6th International Workshop on Theory and Practice in Public Key Cryptography, Miami, FL, USA, January 6-8, 2003, Proceedings*, volume 2567 of *Lecture Notes in Computer Science*, pages 18–30. Springer, 2002.

[CD99] Jan Camenisch and Ivan Damgaard. Verifiable Encryption and Applications to Group Signatures and Signature Sharing. Cryptology ePrint Archive, Report 1999/008, 1999. Available at http://eprint.iacr.org.

[CGH04] Ran Canetti, Oded Goldreich, and Shai Halevi. The Random Oracle Methodology, revisited. *Journal of the ACM*, 51(4):557–594, July 2004.

[CGJ+99] Ran Canetti, Rosario Gennaro, Stanislaw Jarecki, Hugo Krawczyk, and Tal Rabin. Adaptive Security for Threshold Cryptosystems. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 98–115. Springer, 1999.

[Che98] Kefei Chen. Signature with Message Recovery. *Electronics Letters*, 34(20):1934, 1998.

[Che99] Kefei Chen. Reply To Comment - Signature with Message Recovery. *Electronics Letters*, 35(3):217, 1999.

[Che02] Jung Hee Cheon. A Universal Forgery of Hess's Second ID-based Signature against the Known-message Attack. Cryptology ePrint Archive, Report 2002/028, 2002. Available at http://eprint.iacr.org.

[CHK03] Ran Canetti, Shai Halevi, and Jonathan Katz. A Forward-Secure Public-Key Encryption Scheme. In Eli Biham, editor, *Advances in Cryptology - EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4-8, 2003, Proceedings*, volume 2656 of *Lecture Notes in Computer Science*, pages 255–271. Springer, 2003.

[CJMM03] Eric Cronin, Sugih Jamin, Tal Malkin, and Patrick McDaniel. On the Performance, Feasibility, and Use of Forward-secure Signatures. In *Proceedings of the 10th ACM conference on Computer and Communications Security*, pages 131–144. ACM Press, 2003.

[CL03] YoungJu Choie and Eunjeong Lee. Implementation of Tate Pairing of Hyperelliptic Curves of Genus 2. In Jong In Lim and Dong Hoon Lee, editors, *Information Security and Cryptology - ICISC 2003, 6th International Conference Seoul, Korea, November 27-28, 2003, Revised Papers*, volume 2971 of *Lecture Notes in Computer Science*, pages 97–111. Springer, 2003.

[CLT03] Cheng-Kang Chu, Li-Shan Liu, and Wen-Guey Tzeng. A Threshold GQ Signature Scheme. Cryptology ePrint Archive, Report 2003/016, 2003. Available at http://eprint.iacr.org.

[CM99] S. Corson and J. Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. Request For Comments (RFC) 2501, January 1999. Available at http://www.ietf.org/rfc/rfc2501.txt.

[CMS99] Christian Cachin, Silvio Micali, and Markus Stadler. Computationally Private Information Retrieval with Polylogarithmic Communication. In Jacques Stern, editor, *Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding*, volume 1592 of *Lecture Notes in Computer Science*, pages 402–414. Springer, 1999.

[Coc01] Clifford Cocks. An Identity Based Encryption Scheme Based on Quadratic Residues. In Bahram Honary, editor, *Cryptography and Coding, 8th IMA International Conference, Cirencester, UK, December 17-19, 2001, Proceedings*, volume 2260 of *Lecture Notes in Computer Science*, pages 360–363. Springer, 2001.

[Coo71] Stephen A. Cook. The Complexity of Theorem-Proving Procedures. In *Proceedings of the Third Annual ACM symposium on Theory of Computing*, pages 151–158. ACM Press, 1971.

[CS98] Ronald Cramer and Victor Shoup. A Practical Public Key Cryptosystem Provably Secure Against Adaptive Chosen Ciphertext Attack. In Hugo Krawczyk, editor, *Advances in Cryptology - CRYPTO '98, 18th Annual International Cryptology Conference, Santa Barbara, California, USA, August 23-27, 1998, Proceedings*, volume 1462 of *Lecture Notes in Computer Science*, pages 13–25. Springer, 1998.

[CS03] Jan Camenisch and Victor Shoup. Practical Verifiable Encryption and Decryption of Discrete Logarithms. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, volume 2729 of *Lecture Notes in Computer Science*, pages 126–144. Springer, 2003.

[DCB95] Yvo Desmedt, Giovanni Di Crescenzo, and Mike Burmester. Multiplicative Non-abelian Sharing Schemes and their Application to Threshold Cryptography. In Josef Pieprzyk and Reihaneh Safavi-Naini, editors, *Advances in Cryptology - ASIACRYPT '94*, volume 917 of *Lecture Notes in Computer Science*, pages 21–32. Springer, 1995.

[DCK03] Dang Nguyen Duc, Jung Hee Cheon, and Kwangjo Kim. A Forward-Secure Blind Signature Scheme Based on the Strong RSA Assumption. In Robert H. Deng, Sihan Qing, Feng Bao, and Jianying Zhou, editors, *Information and Communications Security, Fifth International Conference, ICICS 2003, Huhehaote City, Inner-Mongolia, October 10-13, 2003, Proceedings*, volume 2836 of *Lecture Notes in Computer Science*, pages 11–21. Springer, 2003.

[DDFY94] Alfredo De Santis, Yvo Desmedt, Yair Frankel, and Moti Yung. How to Share a Function Securely. In *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing*, pages 522–533. ACM Press, 1994.

[DDN91] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-Malleable Cryptography. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing*, pages 542–552. ACM Press, 1991.

[Des88] Yvo Desmedt. Society and Group Oriented Cryptography: A New Concept. In Carl Pomerance, editor, *Advances in Cryptology - CRYPTO '87, A Conference on the Theory and Applications of Cryptographic Techniques, Santa Barbara, California, USA, August 16-20, 1987, Proceedings*, volume 293 of *Lecture Notes in Computer Science*, pages 120–127. Springer, 1988.

[DF92] Yvo Desmedt and Yair Frankel. Shared Generation of Authenticators and Signatures. In Joan Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91, 11th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1991, Proceedings*, volume 576 of *Lecture Notes in Computer Science*, pages 457–469. Springer, 1992.

[DFJW04] Yevgeniy Dodis, Michael J. Freedman, Stanislaw Jarecki, and Shabsi Walfish. Optimal Signcryption from Any Trapdoor Permutation. Cryptology ePrint Archive, Report 2004/020, January 2004. Available at http://eprint.iacr.org.

[DFK+03] Yevgeniy Dodis, Matthew K. Franklin, Jonathan Katz, Atsuko Miyaji, and Moti Yung. Intrusion-Resilient Public-Key Encryption. In Marc Joye, editor, *Topics in Cryptology - CT-RSA 2003, The Cryptographers' Track at the RSA Conference 2003, San Francisco, CA, USA, April 13-17, 2003, Proceedings*, volume 2612 of *Lecture Notes in Computer Science*, pages 19–32. Springer, 2003.

[DFK+04] Yevgeniy Dodis, Matthew K. Franklin, Jonathan Katz, Atsuko Miyaji, and Moti Yung. A Generic Construction for Intrusion-Resilient Public-Key Encryption. In Tatsuaki Okamoto, editor, *Topics in Cryptology - CT-RSA 2004, The Cryptographers' Track at the RSA Conference 2004, San Francisco, CA, USA, February 23-27, 2004, Proceedings*, volume 2964 of *Lecture Notes in Computer Science*. Springer, 2004.

[DH76] Whitfield Diffie and Martin E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, November 1976.

[DKXY02a] Yevgeniy Dodis, Jonathan Katz, Shouhuai Xu, and Moti Yung. Key-Insulated Public Key Cryptosystems. In Lars R. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002, International Conference on the Theory and Applications of Cryptographic Techniques, Amsterdam, The Netherlands, April 28 - May 2, 2002, Proceedings*, volume 2332 of *Lecture Notes in Computer Science*, pages 65–82. Springer, 2002.

[DKXY02b] Yevgeniy Dodis, Jonathan Katz, Shouhuai Xu, and Moti Yung. Strong Key-Insulated Signature Schemes. In Yvo Desmedt, editor, *Public Key Cryptography - PKC 2003, 6th International Workshop on Theory and Practice in Public Key Cryptography, Miami, FL, USA, January 6-8, 2003, Proceedings*, volume 2567 of *Lecture Notes in Computer Science*, pages 130–144. Springer, 2002.

[ElG85] Taher ElGamal. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology, Proceedings of CRYPTO '84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer, 1985.

[FGMY97] Yair Frankel, Peter Gemmell, Philip D. MacKenzie, and Moti Yung. Optimal-Resilience Proactive Public-Key Cryptosystems. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 384–393, 1997.

[FGY96] Yair Frankel, Peter Gemmell, and Moti Yung. Witness-based Cryptographic Program Checking and Robust Function Sharing. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, pages 499–508. ACM Press, 1996.

[FMY99a] Yair Frankel, Philip D. MacKenzie, and Moti Yung. Adaptively-Secure Distributed Public-Key Systems. In Jaroslav Nesetril, editor, *Algorithms - ESA '99, 7th Annual European Symposium, Prague, Czech Republic, July 16-18, 1999, Proceedings*, volume 1643 of *Lecture Notes in Computer Science*, pages 4–27. Springer, 1999.

[FMY99b] Yair Frankel, Philip D. MacKenzie, and Moti Yung. Adaptively-Secure Optimal-Resilience Proactive RSA. In Kwok-Yan Lam, Eiji Okamoto, and Chaoping Xing, editors, *Advances in Cryptology - ASIACRYPT '99, International Conference on the Theory and Applications of Cryptology and Information Security, Singapore, November 14-18, 1999, Proceedings*, volume 1716 of *Lecture Notes in Computer Science*, pages 180–194. Springer, 1999.

[FO99] Eiichiro Fujisaki and Eiji Okamoto. Secure Integration of Asymmetric and Symmetric Encryption Schemes. In Michael Wiener, editor, *Advances in Cryptology – CRYPTO ' 99*, volume 1666 of *Lecture Notes in Computer Science*, pages 537–554. International Association for Cryptologic Research, Springer-Verlag, Berlin Germany, 1999.

[FOPS01] Eiichiro Fujisaki, Tatsuaki Okamoto, David Pointcheval, and Jacques Stern. RSA-OAEP is Secure under the RSA Assumption. In Joe Kilian, editor, *Advances in Cryptology - CRYPTO 2001, 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19-23, 2001, Proceedings*, volume 2139 of *Lecture Notes in Computer Science*, pages 260–274. Springer-Verlag Heidelberg, 2001.

[FS87] Amos Fiat and Adi Shamir. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. In Andrew M. Odlyzko, editor, *Advances in Cryptology - CRYPTO '86, Santa Barbara, California, USA, 1986, Proceedings*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer, 1987.

[FS01] Pierre-Alain Fouque and Jacques Stern. One Round Threshold Discrete-Log Key Generation without Private Channels. In Kwangjo Kim, editor, *Public Key Cryptography, 4th International Workshop on Practice*

and Theory in Public Key Cryptography, PKC 2001, Cheju Island, Korea, February 13-15, 2001, Proceedings, volume 1992 of *Lecture Notes in Computer Science*, pages 300–316. Springer, 2001.

[Gar77] Martin Gardner. Mathematical Games: A New Kind of Cipher that would take Millions of Years to Break. *Scientific American*, 237(2):120–124, August 1977.

[GDH+04] H. W. Go, Y. Dong, Lucas C. K. Hui, Siu Ming Yiu, and Victor O. K. Li. Applying Forward Security and Threshold Cryptography in Ad Hoc Networks. In *The 2004 International Conference on Wireless Networks (ICWN04)*, volume 1, pages 202–205, June 2004.

[GHS02] Steven D. Galbraith, Keith Harrison, and David Soldera. Implementing the Tate Pairing. In Claus Fieker and David R. Kohel, editors, *Algorithmic Number Theory, 5th International Symposium, ANTS-V, Sydney, Australia, July 7-12, 2002, Proceedings*, volume 2369 of *Lecture Notes in Computer Science*, pages 324–337. Springer, 2002.

[Gil99] Niv Gilboa. Two Party RSA Key Generation. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 116–129. Springer, 1999.

[GJKR96a] Rosario Gennaro, Stanislaw Jarecki, Hugo Krawczyk, and Tal Rabin. Robust and Efficient Sharing of RSA Functions. In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*, volume 1109 of *Lecture Notes in Computer Science*, pages 157–172. Springer, 1996.

[GJKR96b] Rosario Gennaro, Stanislaw Jarecki, Hugo Krawczyk, and Tal Rabin. Robust Threshold DSS Signatures. In Ueli M. Maurer, editor, *Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, Spain, May 12-16, 1996, Proceeding*, volume 1070 of *Lecture Notes in Computer Science*, pages 354–371. Springer, 1996.

[GJKR99] Rosario Gennaro, Stanislaw Jarecki, Hugo Krawczyk, and Tal Rabin. Secure Distributed Key Generation for Discrete-Log Based Cryptosystems. In Jacques Stern, editor, *Advances in Cryptology - EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 295–310. Springer, 1999.

[GLZ99] Chandana Gamage, Jussipekka Leiwo, and Yuliang Zheng. Encrypted Message Authentication by Firewalls. In Hideki Imai and Yuliang Zheng,

editors, *Public Key Cryptography: Second International Workshop on Practice and Theory in Public Key Cryptography, PKC'99 Kamakura, Japan, March 1-3, 1999*, volume 1560 of *Lecture Notes in Computer Science*, pages 69–81. Springer-Verlag, Heidelberg, 1999.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[GMR88] Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks. *SIAM Journal on Computing, Special Issue on Cryptography*, 17(2):281–308, 1988.

[GMW87] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to Play Any Mental Game or a Completeness Theorem for Protocols with Honest Majority. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, pages 218–229, 1987.

[GQ90] Louis C. Guillou and Jean-Jacques Quisquater. A Paradoxical Identity-Based Signature Scheme Resulting from Zero-Knowledge. In Shafi Goldwasser, editor, *Advances in Cryptology - CRYPTO '88, 8th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1988, Proceedings*, volume 403 of *Lecture Notes in Computer Science*, pages 216–231. Springer, 1990.

[GS02] Craig Gentry and Alice Silverberg. Hierarchical ID-Based Cryptography. In Yuliang Zheng, editor, *Advances in Cryptology - ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1-5, 2002, Proceedings*, volume 2501 of *Lecture Notes in Computer Science*, pages 548–566. Springer, 2002. Available at http://eprint.iacr.org.

[Gün90] Christoph G. Günther. An Identity-Based Key-Exchange Protocol. In Jean-Jacques Quisquater and Joos Vandewalle, editors, *Advances in Cryptology - EUROCRYPT '89, Workshop on the Theory and Application of Cryptographic Techniques, Houthalen, Belgium, April 10-13, 1989, Proceedings*, volume 434 of *Lecture Notes in Computer Science*, pages 29–37. Springer, 1990.

[Gut02] Peter Gutmann. PKI: It's Not Dead, Just Resting. *IEEE Computer*, 35(8):41 – 49, 2002.

[HC03] Hui-Feng Huang and Chin-Chen Chang. An Efficient Convertible Authenticated Encryption Scheme and Its Variant. In Sihan Qing, Dieter Gollmann, and Jianying Zhou, editors, *Information and Communications Security, 5th International Conference, ICICS 2003, Huhehaote, China, October 10-13, 2003, Proceedings*, volume 2836 of *Lecture Notes in Computer Science*, pages 382–392. Springer, 2003.

[Hes02] Florian Hess. Exponent Group Signature Schemes and Efficient Identity Based Signature Schemes Based on Pairings. Cryptology ePrint Archive, Report 2002/012, 2002. Available at http://eprint.iacr.org.

[Hes03] Florian Hess. Efficient Identity Based Signature Schemes based on Pairings. In Kaisa Nyberg and Howard M. Heys, editors, *Selected Areas in Cryptography, 9th Annual International Workshop, SAC 2002, St. John's, Newfoundland, Canada, August 15-16, 2002. Revised Papers*, volume 2595 of *Lecture Notes in Computer Science*, pages 310–324. Springer, 2003.

[HJKY95] Amir Herzberg, Stanislaw Jarecki, Hugo Krawczyk, and Moti Yung. Proactive Secret Sharing Or: How to Cope With Perpetual Leakage. In Don Coppersmith, editor, *Advances in Cryptology - CRYPTO '95, 15th Annual International Cryptology Conference, Santa Barbara, California, USA, August 27-31, 1995, Proceedings*, volume 963 of *Lecture Notes in Computer Science*, pages 339–352. Springer, 1995.

[HMP94] Patrick Horster, Markus Michels, and Holger Petersen. Authenticated Encryption Schemes with Low Communication Costs. *Electronics Letters*, 30(15):1212–1213, 1994.

[HMP95] Patrick Horster, Markus Michels, and Holger Petersen. Meta-Message Recovery and Meta-Blind Signature Schemes Based on the Discrete Logarithm Problem and Their Applications. In Josef Pieprzyk and Reihaneh Safavi-Naini, editors, *Advances in Cryptology - ASIACRYPT '94, 4th International Conference on the Theory and Applications of Cryptology, Wollongong, Australia, November 28 - December 1, 1994, Proceedings*, volume 917 of *Lecture Notes in Computer Science*, pages 224–237. Springer, 1995.

[HW99] W.H. He and Tsung Cheng Wu. Cryptanalysis and Improvement of Petersen-Michels Signcryption Scheme. *Computers and Digital Techniques, IEE Proceedings*, 146(2):123–124, 1999.

[HWI03] Fei Hu, Chwan-Hwa Wu, and J. David Irwin. A New Forward Secure Signature Scheme using Bilinear Maps. Cryptology ePrint Archive, Report 2003/188, 2003. Available at http://eprint.iacr.org.

[IR01] Gene Itkis and Leonid Reyzin. Forward-Secure Signatures with Optimal Signing and Verifying. In Joe Kilian, editor, *Advances in Cryptology - CRYPTO 2001, 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19-23, 2001, Proceedings*, volume 2139 of *Lecture Notes in Computer Science*, pages 332–354. Springer, 2001.

[IR02] Gene Itkis and Leonid Reyzin. SiBIR: Signer-Base Intrusion-Resilient Signatures. In Moti Yung, editor, *Advances in Cryptology - CRYPTO*

2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings, volume 2442 of *Lecture Notes in Computer Science*, pages 499–514. Springer, 2002.

[ISN87] M. Ito, A. Saito, and T. Nishizeki. Secret Sharing Scheme Realizing General Access Structure. In *Proceedings IEEE Globecom '87*, pages 99–102. IEEE, 1987.

[IT03] Tetsuya Izu and Tsuyoshi Takagi. Efficient Computations of the Tate Pairing for the Large MOV Degrees. In Pil Joong Lee and Chae Hoon Lim, editors, *Information Security and Cryptology - ICISC 2002, 5th International Conference Seoul, Korea, November 28-29, 2002, Revised Papers*, volume 2587 of *Lecture Notes in Computer Science*, pages 283–297. Springer-Verlag Heidelberg, 2003.

[Itk03] Gene Itkis. Intrusion-Resilient Signatures: Generic Constructions, or Defeating Strong Adversary with Minimal Assumptions. In Stelvio Cimato, Clemente Galdi, and Giuseppe Persiano, editors, *Security in Communication Networks, Third International Conference, SCN 2002, Amalfi, Italy, September 11-13, 2002. Revised Papers*, volume 2576 of *Lecture Notes in Computer Science*, pages 102–118. Springer, 2003.

[JJR+03] Ik Rae Jeong, Hee Yun Jeong, Hyun Sook Rhee, Dong Hoon Lee, and Jong In Lim. Provably Secure Encrypt-then-Sign Composition in Hybrid Signcryption. In Pil Joong Lee and Chae Hoon Lim, editors, *Information Security and Cryptology - ICISC 2002, 5th International Conference Seoul, Korea, November 28-29, 2002, Revised Papers*, volume 2587 of *Lecture Notes in Computer Science*, pages 16–34. Springer-Verlag Heidelberg, 2003.

[JL00] Stanislaw Jarecki and Anna Lysyanskaya. Adaptively Secure Threshold Cryptography: Introducing Concurrency, Removing Erasures. In Bart Preneel, editor, *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*, volume 1807 of *Lecture Notes in Computer Science*, pages 221–242. Springer, 2000.

[Jou02] Antoine Joux. The Weil and Tate Pairings as Building Blocks for Public Key Cryptosystems. In Claus Fieker and David R. Kohel, editors, *Algorithmic Number Theory, 5th International Symposium, ANTS-V, Sydney, Australia, July 7-12, 2002, Proceedings*, volume 2369 of *Lecture Notes in Computer Science*, pages 20–32. Springer, 2002.

[KaG+02] Jiejun Kong, Kaixin Luo Haiyun andXu, Daniel Lihui Gu, Mario Gerla, and Songwu Lu. Adaptive Security for Multi-layer Ad-hoc Networks. In *Special Issue of Wireless Communications and Mobile Computing*. John Wiley InterScience Press, 2002.

[Kal98] B. Kaliski. PKCS #1: RSA Encryption Version 1.5. Network Working Group Request For Comments (RFC) 2313, 1998. Available at http://www.ietf.org/rfc/rfc2313.txt.

[Ker83] Auguste Kerckhoff. La cryptographie militaire. *Journal des sciences militaires*, IX, Jan-Feb 1883.

[KM03] DongJin Kwak and SangJae Moon. Efficient Distributed Signcryption Scheme as Group Signcryption. In Jianying Zhou, Moti Yung, and Yongfei Han, editors, *Applied Cryptography and Network Security, ACNS '03*, volume 2846 of *Lecture Notes in Computer Science*, pages 403–417. Springer, 2003.

[KPH04] Bo Gyeong Kang, Je Hong Park, and Sang Geun Hahn. A New Forward Secure Signature Scheme. Cryptology ePrint Archive, Report 2004/183, 2004. Available at http://eprint.iacr.org.

[KR03] Anton Kozlov and Leonid Reyzin. Forward-Secure Signatures with Fast Key Update. In Stelvio Cimato, Clemente Galdi, and Giuseppe Persiano, editors, *Security in Communication Networks, Third International Conference, SCN 2002, Amalfi, Italy, September 11-13, 2002. Revised Papers*, volume 2576 of *Lecture Notes in Computer Science*, pages 241–256. Springer, 2003.

[Kra00] Hugo Krawczyk. Simple Forward-Secure Signatures from Any Signature Scheme. In *Proceedings of the 7th ACM conference on Computer and Communications Security*, pages 108–115. ACM Press, 2000.

[KS00] Jonathan Katz and Bruce Schneier. A Chosen Ciphertext Attack Against Several E-Mail Encryption Protocols. In *Proceedings of the 9th USENIX Security Symposium (SECURITY-00)*, pages 241–246. The USENIX Association, August 14–17 2000.

[KY00] Jonathan Katz and Moti Yung. Complete Characterization of Security Notions for Probabilistic Private-Key Encryption. In *Proceedings of The Thirty-Second Annual ACM Symposium on Theory of Computing*, pages 245–254. ACM Press, 2000.

[LBD+04] Byoungcheon Lee, Colin Boyd, Ed Dawson, Kwangjo Kim, Jeongmo Yang, and Seungjae Yoo. Secure Key Issuing in ID-based Cryptography. In James Hogan, Paul Montague, Martin Purvis, and Chris Steketee, editors, *Australasian Information Security Workshop (AISW 2004), Dunedin, New Zealand, 2004*, volume 32 of *Conferences in Research and Practice in Information Technology*, 2004.

[LC95] Wei-Bin Lee and Chin-Chen Chang. Authenticated Encryption Schemes Without Using a One Way Function. *Electronics Letters*, 31(19):1656–1657, 1995.

[LC04] Lihua Liu and Zhengjun Cao. Universal Forgeability of a Forward-Secure Blind Signature Scheme Proposed by Duc et al. Cryptology ePrint Archive, Report 2004/262, 2004. Available at http://eprint.iacr.org.

[LCT03] Li-Shan Liu, Cheng-Kang Chu, and Wen-Guey Tzeng. A Threshold GQ Signature Scheme. In Jianying Zhou, Moti Yung, and Yongfei Han, editors, *Applied Cryptography and Network Security, First International Conference, ACNS 2003. Kunming, China, October 16-19, 2003, Proceedings*, volume 2846 of *Lecture Notes in Computer Science*, pages 137–150. Springer, 2003.

[LHL95] Chuan-Ming Li, Tzonelih Hwang, and Narn-Yih Lee. Threshold-Multisignature Schemes where Suspected Forgery Implies Traceability of Adversarial Shareholders. In Alfredo De Santis, editor, *Advances in Cryptology - EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 9-12, 1994, Proceedings*, volume 950 of *Lecture Notes in Computer Science*, pages 194–204. Springer, 1995.

[LKP00] Mun-Kyu Lee, Dong Kyue Kim, and Kunsoo Park. An Authenticated Encryption Scheme with Public Verifiability. In *4th Korea-Japan Joint Workshop on Algorithms and Computation*, pages 49–56, June 28 2000.

[LQ03] Benoît Libert and Jean-Jacques Quisquater. New Identity Based Signcryption Schemes from Pairings. In *IEEE Information Theory Workshop*, pages 155–158, 2003. Full Version Available at http://eprint.iacr.org.

[LQ04] Benoît Libert and Jean-Jacques Quisquater. Efficient Signcryption with Key Privacy from Gap Diffie-Hellman Groups. In Feng Bao, Robert H. Deng, and Jianying Zhou, editors, *Public Key Cryptography - PKC 2004, 7th International Workshop on Theory and Practice in Public Key Cryptography, Singapore, March 1-4, 2004*, volume 2947 of *Lecture Notes in Computer Science*, pages 187–200. Springer, 2004.

[Mao03] Wenbo Mao. *Modern Cryptography: Theory & Practice*. Prentice Hall, 2003.

[MC03] Changshe Ma and Kefei Chen. Publicly Verifiable Authenticated Encryption. *Electronics Letters*, 39(3):281–282, 2003.

[ML02] John Malone-Lee. Identity Based Signcryption. Cryptology ePrint Archive, Report 2002/098, 2002. Available at http://eprint.iacr.org.

[MLM03] John Malone-Lee and Wenbo Mao. Two Birds One Stone: Signcryption Using RSA. In Marc Joye, editor, *Topics in Cryptology - CT-RSA 2003, The Cryptographers' Track at the RSA Conference 2003, San Francisco, CA, USA, April 13-17, 2003, Proceedings*, volume 2612 of *Lecture Notes in Computer Science*, pages 211–225. Springer, 2003.

[MMM02] Tal Malkin, Daniele Micciancio, and Sara K. Miner. Efficient Generic Forward-Secure Signatures with an Unbounded Number Of Time Periods. In Lars R. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002, International Conference on the Theory and Applications of Cryptographic Techniques, Amsterdam, The Netherlands, April 28 - May 2, 2002, Proceedings*, volume 2332 of *Lecture Notes in Computer Science*, pages 400–417. Springer, 2002.

[MR01] Silvio Micali and Ronald L. Rivest. Micropayments Revisited. In David Naccache, editor, *Topics in Cryptology - CT-RSA 2001, The Cryptographer's Track at RSA Conference 2001, San Francisco, CA, USA, April 8-12, 2001, Proceedings*, volume 2020 of *Lecture Notes in Computer Science*, pages 149–163. Springer, 2001.

[MV00] Yi Mu and Vijay Varadharajan. Distributed Signcryption. In Bimal K. Roy and Eiji Okamoto, editors, *Progress in Cryptology - INDOCRYPT 2000, First International Conference in Cryptology in India, Calcutta, India, December 10-13, 2000, Proceedings*, volume 1977 of *Lecture Notes in Computer Science*, pages 155–164. Springer, 2000.

[MVN99] Yi Mu, Vijay Varadharajan, and Khanh Quoc Nguyen. Delegated Decryption. In Michael Walker, editor, *Cryptography and Coding, 7th IMA International Conference, Cirencester, UK, December 20-22, 1999, Proceedings*, volume 1746 of *Lecture Notes in Computer Science*, pages 258–269. Springer, 1999.

[MY99] Chris J. Mitchell and Chan Yeob Yeun. Comment - Signature with Message Recovery. *Electronics Letters*, 35(3):217, 1999.

[Nat95] National Institute of Standards and Technology (NIST). Federal Information Processing Standards Publication 186 (1994 May 19): Specifications for the Digital Signature Standard (DSS). *Building in Big Brother: the Cryptographic Policy Debate*, pages 84–86, 1995.

[Nat00] National Institute of Standards and Technology (NIST). The Digital Signature Standard (DSS). Federal Information Processing Standards Publication (FIPS PUB) 186-2, January 2000. updated 2001-10-05.

[NR95] Kaisa Nyberg and Rainer A. Rueppel. Message Recovery for Signature Schemes Based on the Discrete Logarithm Problem. In Alfredo De Santis, editor, *Advances in Cryptology - EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 9-12, 1994, Proceedings*, volume 950 of *Lecture Notes in Computer Science*, pages 182–193. Springer, 1995.

[NR03] Divya Nalla and K. Chandrasekhar Reddy. Signcryption Scheme for Identity-Based Cryptosystems. Cryptology ePrint Archive, Report 2003/066, 2003. Available at http://eprint.iacr.org.

[NY90] Moni Naor and Moti Yung. Public-Key Cryptosystems Provably Secure Against Chosen Ciphertext Attacks. In *Proceedings of The Twenty-Second Annual ACM Symposium on Theory of Computing*, pages 427–437. ACM Press, 1990.

[OP01] Tatsuaki Okamoto and David Pointcheval. The Gap-Problems: A New Class of Problems for the Security of Cryptographic Schemes. In Kwangjo Kim, editor, *Public Key Cryptography, 4th International Workshop on Practice and Theory in Public Key Cryptography, PKC 2001, Cheju Island, Korea, February 13-15, 2001, Proceedings*, volume 1992 of *Lecture Notes in Computer Science*, pages 104–118. Springer, 2001.

[OS91] H. Ong and Claus-Peter Schnorr. Fast Signature Generation With a Fiat Shamir-Like Scheme. In Ivan Damgård, editor, *Advances in Cryptology - EUROCRYPT '90, Workshop on the Theory and Application of of Cryptographic Techniques, Aarhus, Denmark, May 21-24, 1990, Proceedings*, volume 473 of *Lecture Notes in Computer Science*, pages 432–440. Springer, 1991.

[OY91] Rafail Ostrovsky and Moti Yung. How to Withstand Mobile Virus Attacks (Extended Abstract). In *Proceedings of the Tenth Annual ACM Symposium on Principles of Distributed Computing*, pages 51–59. ACM Press, 1991.

[Pai99] Pascal Paillier. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In Jacques Stern, editor, *Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238. Springer, 1999.

[Ped91] Torben P. Pedersen. A Threshold Cryptosystem without a Trusted Party (Extended Abstract). In Donald W. Davies, editor, *Advances in Cryptology - EUROCRYPT '91, Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK, April 8-11, 1991, Proceedings*, volume 547 of *Lecture Notes in Computer Science*, pages 522–526. Springer, 1991. Rump Session.

[Ped92] Torben P. Pedersen. Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing. In Joan Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91, 11th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1991, Proceedings*, volume 576 of *Lecture Notes in Computer Science*, pages 129–140. Springer, 1992.

[PH02] Panagiotis Papadimitratos and Zygmunt J. Haas. Secure Routing for Mobile Ad Hoc Networks. In *SCS Communication Networks and Distributed*

Systems Modeling and Simulation Conference (CNDS 2002) San Antonio, TX, January 27-31, 2002.

[PH03] Panagiotis Papadimitratos and Zygmunt J. Haas. Securing Mobile Ad Hoc Networks. *The Handbook of Ad Hoc Wireless Networks*, pages 551–567, 2003.

[PM98] Holger Petersen and Markus Michels. Cryptanalysis and Improvement of Signcryption Schemes. *IEE Proceedings - Computers and Digital Techniques*, 145(2):149–151, 1998.

[PML04] Bok-Nyong Park, Jihoon Myung, and Wonjun Lee. ISSRP: A Secure Routing Protocol Using Identity-Based Signcryption Scheme in Ad-Hoc Networks. In Kim-Meow Liew, Hong Shen, Simon See, Wentong Cai, Pingzhi Fan, and Susumu Horiguchi, editors, *Parallel and Distributed Computing: Applications and Technologies, 5th International Conference, PDCAT 2004, Singapore, December 8-10, 2004, Proceedings*, volume 3320 of *Lecture Notes in Computer Science*, pages 711–714. Springer, 2004.

[PP03] Josef Pieprzyk and David Pointcheval. Parallel Authentication and Public-Key Encryption. In Reihaneh Safavi-Naini and Jennifer Seberry, editors, *Information Security and Privacy, 8th Australasian Conference, ACISP 2003, Wollongong, Australia, July 9-11, 2003, Proceedings*, volume 2727 of *Lecture Notes in Computer Science*, pages 387–401. Springer, 2003.

[Pre99] Bart Preneel. The State of Cryptographic Hash Functions. In Ivan Damgård, editor, *Lectures on Data Security, Modern Cryptology in Theory and Practice, Summer School, Aarhus, Denmark, July 1998*, volume 1561 of *Lecture Notes in Computer Science*, pages 158–182. Springer, 1999.

[PS98] Guillaume Poupard and Jacques Stern. Generation of Shared RSA Keys by Two Parties. In Kazuo Ohta and Dingyi Pei, editors, *Advances in Cryptology - ASIACRYPT '98, International Conference on the Theory and Applications of Cryptology and Information Security, Beijing, China, October 18-22, 1998, Proceedings*, volume 1514 of *Lecture Notes in Computer Science*, pages 11–24. Springer, 1998.

[PS00] David Pointcheval and Jacques Stern. Security Arguments for Digital Signatures and Blind Signatures. *Journal of Cryptology: The Journal of the International Association for Cryptologic Research*, 13(3):361–396, 2000.

[QPV02] Michael Quisquater, Bart Preneel, and Joos Vandewalle. On the Security of the Threshold Scheme Based on the Chinese Remainder Theorem. In David Naccache and Pascal Paillier, editors, *Public Key Cryptography,*

5th International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 2002, Paris, France, February 12-14, 2002, Proceedings, volume 2274 of *Lecture Notes in Computer Science*, pages 199–210. Springer, 2002.

[Rab98] Tal Rabin. A Simplified Approach to Threshold and Proactive RSA. In Hugo Krawczyk, editor, *Advances in Cryptology - CRYPTO '98, 18th Annual International Cryptology Conference, Santa Barbara, California, USA, August 23-27, 1998, Proceedings*, volume 1462 of *Lecture Notes in Computer Science*, pages 89–104. Springer, 1998.

[Riv92] Ronald L. Rivest. The MD5 Message-Digest Algorithm. Network Working Group Request For Comments (RFC) 1321, April 1992. Available at http://www.ietf.org/rfc/rfc2501.txt.

[RS92] Charles Rackoff and Daniel R. Simon. Non-Interactive Zero-Knowledge Proof of Knowledge and Chosen Ciphertext Attack. In Joan Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91, 11th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1991, Proceedings*, volume 576 of *Lecture Notes in Computer Science*, pages 433–444. Springer, 1992.

[RSA78] 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, February 1978.

[Sch91] Claus-Peter Schnorr. Efficient Signature Generation by Smart Cards. *Journal of Cryptology: The Journal of the International Association for Cryptologic Research*, 4(3):161–174, 1991.

[SDL+02] Kimaya Sanzgiri, Bridget Dahill, Brian Neil Levine, Clay Shields, and Elizabeth M. Belding-Royer. A Secure Routing Protocol for Ad Hoc Networks. In *10th IEEE International Conference on Network Protocols (ICNP 2002), 12-15 November 2002, Paris, France, Proceedings*, pages 78–89. IEEE Computer Society, 2002.

[Sha48] Claude E. Shannon. A Mathematical Theory of Communication. *Bell Systems Technical Journal*, 27:379–423, 623–656, 1948.

[Sha49] Claude E. Shannon. Communication Theory of Secrecy Systems. *Bell Systems Technical Journal*, 28(3):656–715, October 1949.

[Sha79] Adi Shamir. How to Share A Secret. *Communications of the ACM*, 22(11):612–613, 1979.

[Sha85] Adi Shamir. Identity-Based Cryptosystems and Signature Schemes. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology, Proceedings of CRYPTO 1984, Santa Barbara, California, USA, August 19-*

22, 1984, Proceedings, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, 19–22 August 1985.

[Sho00] Victor Shoup. Practical Threshold Signatures. In Bart Preneel, editor, *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*, volume 1807 of *Lecture Notes in Computer Science*, pages 207–220. Springer, 2000.

[Sim91] Gustavus J. Simmons. Geometric Shared Secret and/or Shared Control Schemes. In Alfred Menezes and Scott A. Vanstone, editors, *Advances in Cryptology - CRYPTO '90, 10th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1990, Proceedings*, volume 537 of *Lecture Notes in Computer Science*, pages 216–241. Springer, 1991.

[SK03] Ryuichi Sakai and Masao Kasahara. ID-based Cryptosystems with Pairing on Elliptic Curve. Cryptology ePrint Archive, Report 2003/054, 2003. Available at http://eprint.iacr.org.

[SLS03] Jun-Bum Shin, Kwangsu Lee, and Kyungah Shim. New DSA-Verifiable Signcryption Schemes. In Pil Joong Lee and Chae Hoon Lim, editors, *Information Security and Cryptology - ICISC 2002, 5th International Conference Seoul, Korea, November 28-29, 2002, Revised Papers*, volume 2587 of *Lecture Notes in Computer Science*, pages 35–47. Springer-Verlag Heidelberg, 2003.

[Son01] Dawn Xiaodong Song. Practical Forward Secure Group Signature Schemes. In *Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 225–234. ACM Press, 2001.

[SZ00] Ron Steinfeld and Yuliang Zheng. A Signcryption Scheme Based on Integer Factorization. In Josef Pieprzyk, Eiji Okamoto, and Jennifer Seberry, editors, *Information Security, Third International Workshop, ISW 2000, Wollongong, NSW, Australia, December 20-21, 2000, Proceedings*, volume 1975 of *Lecture Notes in Computer Science*, pages 308–322. Springer, 2000.

[Tsu03] Gene Tsudik. Weak Forward Security in Mediated RSA. In Stelvio Cimato, Clemente Galdi, and Giuseppe Persiano, editors, *Security in Communication Networks, Third International Conference, SCN 2002, Amalfi, Italy, September 11-13, 2002. Revised Papers*, volume 2576 of *Lecture Notes in Computer Science*, pages 45–54. Springer, 2003.

[TT01] Wen-Guey Tzeng and Zhi-Jia Tzeng. Robust Forward-Secure Signature Schemes with Proactive Security. In Kwangjo Kim, editor, *Public Key Cryptography, 4th International Workshop on Practice and Theory*

in Public Key Cryptography, PKC 2001, Cheju Island, Korea, February 13-15, 2001, Proceedings, volume 1992 of *Lecture Notes in Computer Science*, pages 264–276. Springer, 2001.

[Uni95] United State Department of Commerce. FIPS PUB 180-1: Secure Hash Standard. Federal Information Processing Standards Publication, April 1995.

[Uni01] United State Department of Commerce. FIPS PUB 180-2: Secure Hash Standard. Draft Federal Information Processing Standards Publication, 2001.

[VOT04] Raja Rai Singh Verma, Donal O'Mahony, and Hitesh Tewari. Progressive Authentication in Ad Hoc Networks. In *The Fifth European Wireless Conference: Mobile and Wireless Systems beyond 3G*, 2004.

[Wan04] Guilin Wang. On the Security of a Group Signature Scheme with Forward Security. In Jong In Lim and Dong Hoon Lee, editors, *Information Security and Cryptology - ICISC 2003, 6th International Conference, Seoul, Korea, November 27-28, 2003, Revised Papers*, volume 2971 of *Lecture Notes in Computer Science*, pages 27–39. Springer, 2004.

[WBMC04] Guilin Wang, Feng Bao, Changshe Ma, and Kefei Chen. Efficient Authenticated Encryption Schemes with Public Verifiability. In *Proceedings of the 60th IEEE Vehicular Technology Conference (VTC 2004-Fall) - Wireless Technologies for Global Security*. IEEE Computer Society, 2004.

[WDKM04] Guilin Wang, Robert H. Deng, Dongjin Kwak, and Sangjae Moon. Security Analysis of Two Signcryption Schemes. In *Information Security Conference (ISC 2004)*, Lecture Notes in Computer Science. Springer-Verlag, 2004. To Appear.

[WH02] Tzong-Sun Wu and Chien-Lung Hsu. Convertible Authenticated Encryption Scheme. *Journal of Systems and Software*, 62(3):205–209, 2002.

[Whe97] David Wheeler. Transactions Using Bets. In T. Mark A. Lomas, editor, *Security Protocols, International Workshop, Cambridge, United Kingdom, April 10-12, 1996, Proceedings*, volume 1189 of *Lecture Notes in Computer Science*, pages 89–92. Springer, 1997.

[WLH03] Hsiang-An Wen, Chein-Min Lo, and Tzonelih Hwang. Comment - Publicly Verifiable Authenticated Encryption. *Electronics Letters*, 39(19):1382–1383, 2003.

[WLXZ00] Huaxiong Wang, Kwok-Yan Lam, Guozhen Xiao, and Huanhui Zhao. On Multiplicative Secret Sharing Schemes. In Ed Dawson, Andrew Clark, and Colin Boyd, editors, *Information Security and Privacy, 5th Australasian Conference, ACISP 2000, Brisbane, Australia, July 10-12, 2000,*

Proceedings, volume 1841 of *Lecture Notes in Computer Science*, pages 342–351. Springer, 2000.

[WWW02] Theodore M. Wong, Chenxi Wang, and Jeannette M. Wing. Verifiable Secret Redistribution for Threshold Sharing Schemes. CMU SCS Technical Report CMU-CS-02-114, February 2002. Available at http://www.pdl.cmu.edu/Pasis.

[Yao82] Andrew Chi-Chih Yao. Protocols for Secure Computations. In *Proceedings of the 23th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 160–164, 1982.

[Yeu99] Chan Yeob Yeun. Digital Signature with Message Recovery and Authenticated Encryption (Signcryption) - A Comparison. In Michael Walker, editor, *Cryptography and Coding, 7th IMA International Conference, Cirencester, UK, December 20-22, 1999, Proceedings*, volume 1746 of *Lecture Notes in Computer Science*, pages 307–312. Springer, 1999.

[YL02] Dae Hyun Yum and Pil Joong Lee. New Signcryption Schemes Based on KCDSA. In Kwangjo Kim, editor, *Information Security and Cryptology - ICISC 2001, 4th International Conference Seoul, Korea, December 6-7, 2001, Proceedings*, volume 2288 of *Lecture Notes in Computer Science*, pages 305–317. Springer, 2002.

[YY04] Adam L. Young and Moti Yung. *Malicious Cryptography: Exposing Cryptovirology*. Wiley, 2004.

[Zhe97] Yuliang Zheng. Digital Signcryption or How to Achieve Cost (Signature & Encryption) $<<$ Cost(Signature) + Cost(Encryption). In Burton S. Kaliski Jr., editor, *Advances in Cryptology: Proceedings of CRYPTO 1997 5th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997*, volume 1294 of *Lecture Notes in Computer Science*, pages 165–179. Springer-Verlag, 1997.

[Zhe98] Yuliang Zheng. Signcryption and Its Applications in Efficient Public Key Solutions. In Eiji Okamoto, George I. Davida, and Masahiro Mambo, editors, *Information Security, First International Workshop, ISW '97, Tatsunokuchi, Japan, September 17-19, 1997, Proceedings*, volume 1396 of *Lecture Notes in Computer Science*, pages 291–312. Springer, 1998. Invited Lecture.

[ZI03] Rui Zhang and Hideki Imai. Round Optimal Distributed Key Generation of Threshold Cryptosystem Based on Discrete Logarithm Problem. In Jianying Zhou, Moti Yung, and Yongfei Han, editors, *Applied Cryptography and Network Security, First International Conference, ACNS 2003. Kunming, China, October 16-19, 2003, Proceedings*, volume 2846 of *Lecture Notes in Computer Science*, pages 96–110. Springer, 2003.

[ZX04] Yuefei Zhu and Dan Xu. An Efficient Key-Evolving Signature Scheme Based on Pairing. In *10th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS 2004), 26-28 May 2004, Suzhou, China*, pages 68–73. IEEE Computer Society, 2004.