A Thesis for the Degree of Master of Science

Provably Secure Threshold Blind Signature Scheme Using Pairings

Vo Duc Liem

School of Engineering

Information and Communications University

2003

Provably Secure Threshold Blind Signature Scheme Using Pairings

Provably Secure Threshold Blind Signature Scheme Using Pairings

Advisor : Professor Kwangjo Kim

by

Vo Duc Liem
School of Engineering
Information and Communications University

A thesis submitted to the faculty of Information and Communications University in partial fulfillment of the requirements for the degree of Master of Science in the School of Engineering

Daejeon, Korea
Jul. 1. 2003
Approved by

(signed)

Professor Kwangjo Kim
Major Advisor

Provably Secure Threshold Blind Signature Scheme Using Pairings

Vo Duc Liem

We certify that this work has passed the scholastic standards required by Information and Communications University as a thesis for the degree of Master of Science

Jul. 1. 2003

Approved:

Chairman of the Committee
Kwangjo Kim, Professor
School of Engineering

Committee Member
Younghee Lee, Professor
School of Engineering

Committee Member
C. Pandu Rangan, Professor
School of Engineering

M.S. Vo Duc Liem
2001824

Provably Secure Threshold Blind Signature Scheme Using Pairings

School of Engineering, 2003, 41p.
Major Advisor : Prof. Kwangjo Kim.
Text in English

Abstract

Public Key Cryptosystems (PKC) were first introduced by Diffie and Hellman [10]. The advantages of PKC have been proved through various applications. A traditional application of PKC is to issue a digital signature on a document. In PKC, a person has a key pair which contains a public key and the corresponding private key. Using the private key, a person can produce digital signatures on documents to prove that those documents are indeed generated by him. Anyone can easily verify the signatures on those documents using the public key of that person. Thus, the integrity of the documents is protected by the digital signatures on those documents.

A *threshold* digital signature scheme requires a certain number of people to produce digital signatures on documents. In other words, the secret key used to sign documents is distributed (in pieces) among some people. To sign a document a certain number of people (threshold value) have to cooperate in order to construct the digital signature on the document.

A *blind* digital signature scheme, first proposed by Chaum [8], is an important cryptographic component in many applications. Using a blind signature scheme, a user can get signature on a message from a signer without revealing the message content. Both types of signature are playing important roles

in cryptography as well as practical applications such as e-cash and e-voting systems.

In this thesis, we construct a new threshold blind digital signature based on pairings without a trusted third party, which combines the notions of a threshold digital signature and a blind digital signature. A threshold blind digital signature allows possession of a private key to be distributed among a group of signers while a user can get a signature on a message without revealing the message content. Threshold blind digital signatures have various applications. For example, using such signatures we can design protocols for secure distributed electronic banking, or secure electronic voting with multiple administrators. Our scheme operates on Gap Diffie-Hellman (GDH) group, where CDH problems are hard while DDH problems are easy. For instance, we use pairings that could be built from Weil pairing or Tate pairing. With this construction, the scheme is more efficient than previous ones with respect to signature size while keeping the same level of security. We prove that the proposed signature scheme is secure against the well known attacks if CDH problem is intractable in the random oracle model. We also compare this scheme with other threshold blind signature schemes. To the best of our knowledge, we claim that our scheme is the first threshold blind signature using pairings with provable security in the random oracle model.

Contents

List of Tables

List of Figures

List of Abbreviations

CDH Computational Diffie-Hellman

CT-CDH Chosen-target Computational Diffie-Hellman

DH Diffie-Hellman

DDH Decisional Diffie-Hellman

DLP Discrete Logarithm Problem

ECDLP Elliptic Curve Discrete Logarithm problem

GDH Gap Diffie-Hellman

IFP Integer Factorization Problem

PPT Probabilistic Polynomial Time

ROM Random Oracle Model

RSA Rivest-Shamir-Adleman (encryption scheme)

TTP Trusted Third Party

CA Certification Authority

List of Notations

$\mathcal{A}, \mathcal{B}$ polynomial-time adversaries

`pk` a public key

`sk` a secret key

σ a signature on a message

σ_i a signature share of the i-th player

$\hat{e}$ a bilinear pairing

$\mathbb{G}$ a cyclic group of a prime order

H a one-way hash function, hash from bit string $\{0,1\}^*$ onto group $\mathbb{G}$ element

HP a group of the honest players

M a message $\{0,1\}^*$

M' a blind message (an element of $\mathbb{G}$)

$\mathcal{O}$ point at infinity (on an elliptic curve)

Q, s a public key and an implicit secret key of group of users

Q_i, s_i a public share and a secret share of the i-th player.

P, P' generators of group $\mathbb{G}$

L_i the i-th player

$\in_{\mathcal{R}}$ chosen at random

$\mathbb{Z}$ integers

$\mathbb{Z}_q$ integers modulo q

$\mathbb{Z}_q^*$ a group under multiplication modulo q

Chapter 1
Introduction

1.1 Digital signatures

Authenticity, or proving origin, of the electronic documents is very important in the electronic applications. In the symmetric cryptosystems, Message Authentication Codes (MACs) can ensure authenticity as well as integrity of the documents. In MAC algorithms, one uses a secret key to generate a MAC of a document then sends both the MAC and the document to the verifiers. To verify the correctness of the issued MAC on a given document, the verifiers need to know the secret key used to generate that MAC. The problem here is how the producer of the MAC can transfer the secret key to the verifiers securely? With the invention of the public key cryptosystems, or the asymmetric cryptosystems, the verifying authenticity of the electronic documents can be done easily. The similar notion to the MACs is called the digital signatures. In the public key cryptosystems, a user possesses a key pair: a private key, which is known by the user only, and a corresponding public key, which is known by everyone (i.e., by publishing the user's public key into a directory). To produce a digital signature, the signer uses his private key to *sign* a document and sends the document along with its signature to the verifiers. The verifiers can get the signer's public key easily from a public key directory, and use this public key to verify the signature on the document. Because of the uniqueness of a key pair in the public key cryptosystems, if the signature is really issued by the signer, the verifiers can use the signer's

public key to verify the signature successfully, and vice versa.

As the handwriting signatures, the digital signatures also must have properties such as easy to issue, easy to verify and difficult to forge. Since the notion of the digital signatures is realized, there are many proposals of the public key cryptosystems and their corresponding digital signature schemes, such as ElGamal, Schnorr, RSA, etc. In addition, the digital signatures also have been designed to have various properties to support different applications. We can find in literature many types of the digital signatures, for example group signatures, multisignatures, threshold signatures, blind signatures, and so forth. In this thesis, we deal with the threshold blind digital signature which combines two types of well known digital signatures, i.e., threshold digital signatures [9], [12], [13], [14], [15] and blind digital signature [28], [29]. The practical applications of threshold blind digital signatures are e-voting, e-payment systems, etc.

1.2 Threshold blind digital signatures

A threshold signature scheme distributes the signing abilities to a group of signers such that a digital signature on a message cannot be produced by a predetermined number of signers. With this property, misbehavior caused by a single dishonest signer will be eliminated in many applications. For example, in the e-voting system managed by a single administrator [24], the administrator has full power to validate any vote. If the administrator is dishonest, he can change any vote that he wants for some purposes. In these situations, we want to have multiple administrators to authorize votes, while the system is working correctly.

A blind signature scheme, on the other hand, gives users ability to get a digital signature from a signer without revealing the message content. This property is very important for implementing e-voting, e-commerce, and e-payment systems, etc. For instance, when a buyer purchases merchandize

from a shop, the buyer gets a bank's signature on the payment given to the shop and keeps secret what merchandize is from the bank.

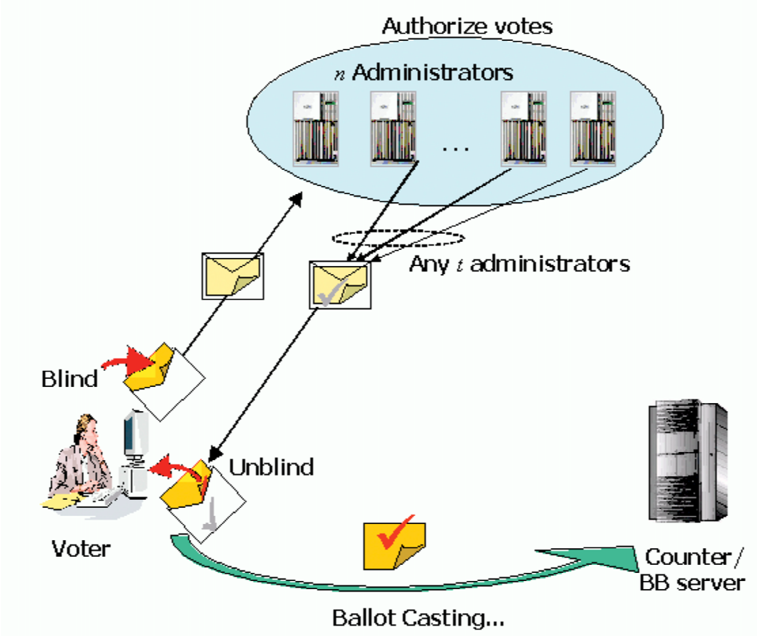

Figure 1.1: A threshold blind signature scheme in an e-voting system with multiple administrators

A threshold blind signature combines a threshold signature and a blind one to exhibit both properties. Therefore, a threshold blind signature, while giving user ability to get signature on a message without revealing its content, still maintains the shared secret key to be distributed among signers. This type of signature can be applied in any application for blind signatures (e.g. e-cash, e-voting) where the signing secret key needs to be distributed to enhance security level. An example is illustrated in Figure 1.1. This is an e-voting system with multiple administrators. In this system, a voter selects his candidate then blinds his selection. The voter then requests the authorize vote system to validate his vote. The authorize vote system needs at least t administrators to perform this work. With this configuration, a single

administrator problem is eliminated. Moreover, the security of the system is improved, that is even several secret keys are compromised, the system is still working correctly.

There are several studies on this type of signature schemes such as in [19] and [20]. Both papers have proposed the threshold blind signature scheme based on the discrete logarithm problem and have used the Okamoto-Schnorr blind signature scheme as the underlying signature scheme. However, the scheme [19] has shown better performance than that of the scheme [20].

1.3 Our contributions

In this thesis, we propose a new threshold blind signature scheme based on pairings. As mentioned above, the previous threshold blind digital signature schemes [19], [20] are based on the discrete logarithm problem over finite fields. They both were built from the Okamoto-Schnorr blind digital signature [28] using secret sharing techniques. Working on an elliptic curve over finite field, our proposed signature scheme has achieved efficiency in terms of the signature size compared to the previous schemes [19] and [20]. This is the first contribution of this thesis.

Recently, the Okamoto-Schnorr signature scheme and its blind version were proved to be breakable under Generalized Birthday acttack [33]. Therefore, any signature scheme based on these types of signature schemes will be insecure. Our constructed signature scheme is based on CDH problem which is intractable in *GDH* group, and we also prove security of our construction in a formal way. This is our second contribution.

To the best of our knowledge, our proposed scheme is the first threshold blind digital signature scheme using pairing and provably secure in the random oracle model.

1.4 Outline of the thesis

Chapter 2 introduces some background on the bilinear pairings and the cryptographic primitives that we use in our proposed scheme. We define the notions of security as well as the security model in Chapter 3. Chapter 4 presents our proposed threshold blind signature scheme. Chapter 5 comes up with the security proof of the proposed scheme. In Chapter 6, we will evaluate performance of our scheme and compare with other schemes as well. Chapter 7 will be given our conclusions and suggestions for future work.

Chapter 2
Background and related work

2.1 Concepts of bilinear pairings

We summarize some concepts of bilinear pairings using similar notations used by Zhang and Kim [34] which were used to design ID-based blind signature and ring signature based on pairings.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be additive and multiplicative groups of the same prime order q, respectively. Let P be a generator of $\mathbb{G}_1$. Assume that the discrete logarithm problems in both $\mathbb{G}_1$ and $\mathbb{G}_2$ are hard. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a pairing which satisfies the following properties:

1. *Bilinear*: $\hat{e}(aP, bP') = \hat{e}(P, P')^{ab}$ for all $P, P' \in \mathbb{G}_1$ and all $a, b \in \mathbb{Z}$.
2. *Non-degenerate*: If $\hat{e}(P, P') = 1 \; \forall \, P' \in \mathbb{G}_1$ then $P = \mathcal{O}$.
3. *Computable*: There is an efficient algorithm such as [2] to compute $\hat{e}(P, P')$ for any $P, P' \in \mathbb{G}_1$.

To construct the bilinear pairing, we can use the Weil pairing or Tate pairing associated with supersingular elliptic curves.

Under such group $\mathbb{G}_1$, we can define the following hard cryptographic problems:

– **Discrete Logarithm (DL) Problem:** Given $P, P' \in \mathbb{G}_1$, find an integer n such that $P = nP'$ whenever such integer exists.

– **Computational Diffie-Hellman (CDH) Problem:** Given a triple $(P, aP, bP) \in \mathbb{G}_1$ for $a, b \in \mathbb{Z}_q^*$, find the element abP.

– **Decision Diffie-Hellman (DDH) Problem:** Given a quadruple $(P,\ aP,\ bP,\ cP) \in \mathbb{G}_1$ for $a, b, c \in \mathbb{Z}_q^*$, decide whether $c = ab \pmod q$ or not.

– **Gap Diffie-Hellman (GDH) Problem:** A class of problems where the CDH problems are hard but the DDH problems are easy. That is, given a triple $(P, aP, bP) \in \mathbb{G}_1$ for $a, b \in \mathbb{Z}_q^*$, find the element abP with the help of the DDH oracle, which can answer whether a given quadruple $(P,\ aP,\ bP,\ cP) \in \mathbb{G}_1$ is a Diffie-Hellman quadruple or not.

Groups, where the CDH problems are hard but the DDH problems are easy, are called Gap Diffie-Hellman (*GDH*) groups. Details about *GDH* groups can be found in [5], [6], [17], and [25].

2.2 Blind digital signature scheme

The blind digital signature was first introduced by Chaum [8] to provide anonymity without revealing message contents. Basically, we can understand a blind signature scheme as a cryptographic protocol involving two parties, a user A and a signer B. The user A wants to get a signature of the signer B on a message M. Firstly, A blinds M into M' and then sends M' to B. B generates signature σ' on M' and returns σ' to A. Receiving σ', A unblinds σ' into σ and outputs σ as the signature on the message M. By this way, A can protect content of M from B. In addition, whenever B is given a pair of (M, σ), B cannot determine when or for whom he signed that message. This concept is very important in electronic payment systems, electronic voting systems. For example, in the electronic cash systems, a buyer represents a user, an electronic coin represents a document need to be signed and a bank

represents a signer. In the payment transactions, if the user wants to spend a coin, he needs the signature of the bank on that coin, but he does not want to let the bank know the purpose of the spent coin. With the help of the blind signature schemes, the user can blind that coin and requests the bank to sign on it. After that, the user can spend the coin while the bank cannot determine how the coin is spent. Since Chaum's proposal [8], there are many

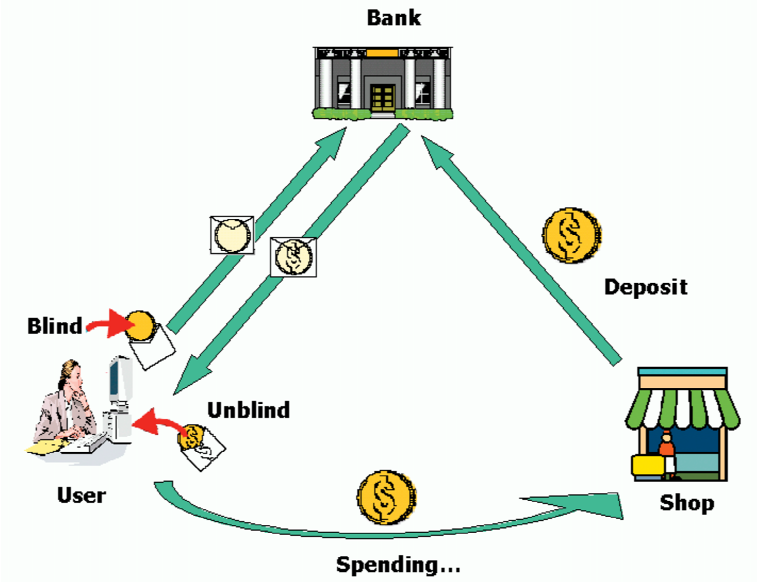

Figure 2.1: Blind signatures in e-cash system

intensive researches [3], [28], [29] dealing with the blind digital signatures as well as their security.

2.3 Blind signature scheme based on GDH problem

Recently, Boldyreva [7] introduced a blind digital signature scheme based on Gap Diffie-Hellman problem and proved its security. For our threshold blind

signature scheme, we build a blind version of the signature scheme in [6], which is similar to that of [7], defined as follows:

Let $\mathbb{G}_1$ be GDH group of prime order q. Public information is $I = (q, P, H)$ where P is a generator of $\mathbb{G}_1$ and $H : \{0,1\}^* \rightarrow \mathbb{G}_1$ is an one-way hash function. The new blind GDH signature scheme $\mathsf{BGS} = (\mathcal{BK}, \mathcal{BS}, \mathcal{BV})$, where $\mathcal{BK}, \mathcal{BS}$, and $\mathcal{BV}$ are key generation, blind signing and verification protocols, respectively, is defined as:

- $\mathcal{BK}(I)$: Pick randomly $s \stackrel{R}{\leftarrow} \mathbb{Z}_q^*$ and compute $Q \leftarrow sP$. The algorithm will return the public key $\mathtt{pk}$ and the secret key $\mathtt{sk}$, where $\mathtt{pk} = (q, P, H, Q)$, and $\mathtt{sk} = s$.
- $\mathcal{BS}(I, \mathtt{sk}, M)$: The user wants a message $M \in \{0,1\}^*$ to be signed "blindly".
  - The user picks a random number $r \stackrel{R}{\leftarrow} \mathbb{Z}_q^*$ and computes a blind message M' of the message M, $M' = r \cdot H(M)$. He sends M' to the signer.
  - The signer signs on M', $\sigma' = s \cdot M'$ and sends back σ' to the user.
  - Receiving σ', the user unblinds it and gets the signature of the message M, $\sigma = r^{-1} \cdot \sigma'$ and outputs (M, σ).
- $\mathcal{BV}(\mathtt{pk}, M, \sigma)$: If $\mathcal{V}_{DDH}(P, Q, H(M), \sigma) = 1$ then return 1 else return 0, where $\mathcal{V}_{DDH}(\cdot)$ is an efficient algorithm which solves the DDH problem in $\mathbb{G}_1$.

The blind signature scheme based on GDH problem works as depicted on Figure 2.2

2.4 Threshold scheme

In the traditional public key cryptosystems, an individual (a server or a person) keeps the secret key and performs all secret-key-related operations such

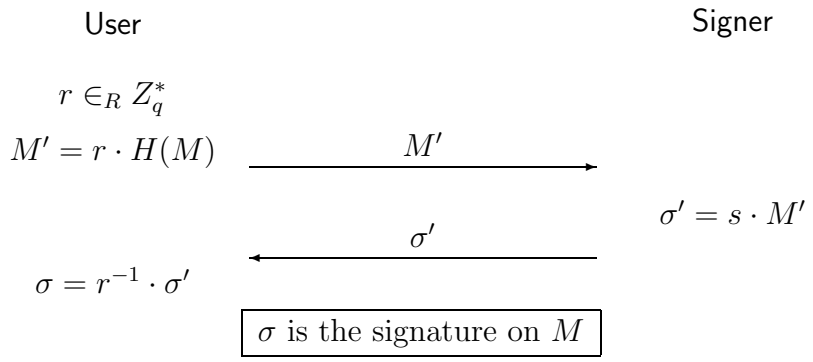

Figure 2.2: Blind signature scheme based on *GDH* problem

as decrypting ciphertext encrypted by the corresponding public key, issuing a digital signature. In some scenarios, it can be dangerous since the secret key holder is very powerful. The threshold scheme can be used in these situations to distribute the secret information to several parties to increase security level. The concept of a threshold scheme was first introduced by Shamir [30]. In the (t, n)-threshold scheme, a secret information D is divided into n pieces $D_1, D_2, \ldots, D_n$ such that:

1. Knowledge of any t or more D_i pieces makes D easily computable;

2. Knowledge of any $t-1$ or fewer D_i pieces leaves D uncomputable.

As mentioned above, the (t, n)-threshold scheme enables possession of the secret key to be distributed to n parities in a public key cryptosystem. Consequently, only t or more parties can decrypt a ciphertext encrypted by the corresponding public key or produce a digital signature on a message. With fewer t parties, the decryption of the ciphertext cannot be done.

Threshold schemes are very suitable to applications in which the secret need to be shared among parties such as electronic cash, electronic voting, fault-tolerant applications, or trust distribution, etc. For example, in Public Key Infrastructure (PKI), Root Certification Authority (CA) is the most

powerful entity. Root CA certifies a public key of every lower level CA by using its own secret key digitally signed on the public keys of the lower level CAs. If the Root CA's secret key is kept by only individual, it will be very risky when this key is compromised. An adversary, who steals this secret key by some ways, can make certificates for any illegal CA for his benefit. If a (t, n)-threshold scheme is applied in this scenario, the Root CA's secret key will be distributed to n parties. Compromising up to $t-1$ parties will not help the adversaries to recover the original secret key. Therefore, the adversaries cannot make any illegal certificate in this case.

Many researches on the threshold schemes and the additional security aspects were found in [9], [12], [13] [19], [20], [26], [27], and [31]. The original threshold scheme proposed by Shamir [30] requires a dealer to distribute the shared secrets to the parties. Later, Feldman [11], Pedersen [26], [27] proposed a threshold cryptosystem without a **TTP**. In these schemes, each party acts as a dealer to choose the secret key and distribute it verifiably to other parties. Subsequently, a group of the honest parties is formed and the group members recover their secret shares. Using Pedersen's protocol [27], Gennaro *et al.* [13] proposed the Distributed Key Generation (DKG) protocol which is based on the discrete logarithm problem. In this thesis, we use the DKG protocol as one of components to build our new threshold blind signature scheme.

2.5 Distributed Key Generation

The DKG protocol defined in [13] is a secret sharing protocol without any **TTP** (dealer). DKG works as a main component of the threshold cryptosystems. Feldman [11] proposed a verifiable secret sharing (VSS) protocol based on the discrete logarithm problem. However, his protocol has been shown to have a security flaw, since an adversary can influence the distribution of the result of his protocol to a non-uniform distribution. Next, by utilizing Feldman's VSS protocol, Pedersen [26] proposed the first DKG protocol. Re-

cently, Gennaro *et al.* in [13] carried out a secure DKG protocol with the complete security proof. We now describe this DKG protocol briefly. The secure DKG protocol is as follows:

Generating r:

1. Each player L_i Pedersen-VSS of a random value r_i as a dealer:

   (a) L_i chooses two random polynomials f_i and f_i' over $\mathbb{Z}_q$ of degree $t-1$ where:

   $$f_i(x) = a_{i0} + a_{i1}x + \cdots + a_{it-1}x^{t-1}$$

   and

   $$f_i'(x) = b_{i0} + b_{i1}x + \cdots + b_{it-1}x^{t-1}$$

   Let $r_i = a_{i0} = f_i(0)$ and $r_i' = b_{i0} = f_i'(0)$. L_i broadcasts $C_{ik} = g^{a_{ik}}h^{b_{ik}} \bmod p$, for $k = 0, \ldots, t-1$. L_i computes the shares $s_{ij} = f_i(j) \bmod p, s_{ij}' = f_i'(j) \bmod p$, for $j = 1, 2, \ldots, n$ and sends (s_{ij} and s_{ij}') secretly to player L_j.

   (b) Each player L_j verifies the shares (s_{ij}, s_{ij}'). L_j checks if

   $$g^{s_{ij}}h^{s_{ij}'} = \prod_{k=0}^{t-1}(C_{ik})^{j^k} \bmod p. \tag{2.1}$$

   If the check fails for an index i, L_j broadcasts a complaint against L_i.

   (c) Each player L_i who received a complaint defends himself by broadcasting the value (s_{ij}, s_{ij}') that satisfies Eq. (2.1).

   (d) Any player who either received more than $t-1$ complaints, or answered wrong value to a complaint is disqualified.

2. Let $H_0 := \{L_j | L_j \text{ is a not disqualified player}\}$.

3. The distributed secret value r is not explicitly computed by any party but it equals $r = \sum_{j \in H_0} r_i$. Each player L_i sets his share of the secret as $s_i = \sum_{j \in H_0} s_{ij} \mod q$.

Extracting $y = g^r \mod p$:

1. Each player $L_j \in H_0$ broadcasts $A_{jk} = g^{a_{jk}} \mod p$, for $k = 0, \ldots, t-1$

2. Each player L_j verifies the values broadcasted by other players in H_0, namely, for each $i \in H_0$, L_j checks if

$$g^{s_{ij}} = \prod_{k=0}^{t-1} (A_{ik})^{j^k} \mod p. \tag{2.2}$$

   If the check fails for an index i, L_j complains against L_i by broadcasting the value (s_{ij}, s'_{ij}) that satisfies Eq. (2.1) but does not satisfy Eq. (2.2).

3. For players L_i who received at least one valid complaint, i.e., value which satisfies Eq. (2.1) but does not satisfy Eq. (2.2), the other players run the reconstruction phase of Pedersen's VSS to compute $r_i, f_i(\cdot), A_{ik}$ for $k = 0, \ldots, t-1$ in the clear. For all players in H_0, set $y_i = A_{i0} = g^{r_i} \mod p$. Compute $y = \prod_{i \in H_0} y_i \mod p$.

Chapter 3
Definitions of security

In this chapter, we present the security model as well as the security definitions of the threshold blind signature scheme.

3.1 Standard model and Random oracle model

Like every cryptographic scheme, a digital signature scheme after being proposed need to be shown to be secure. There are two common formal methods which are used to prove security of cryptographic schemes. One is using the standard model and the other is using the random oracle model.

3.1.1 The standard model

The standard model, or the complexity-based model, is a preferred approach, mathematical cryptography. In this model, one starts by assuming an underlying well-defined problem to be hard, that is the problem is widely believed to be very hard. For example, integer factoring problem and discrete logarithm problem are hard problems. Then, the showing security of a cryptographic scheme turns out to verify that if there exists an attacker who can successfully attack the scheme, then one can construct an attacker who can break the presumed hardness. In other words, one may state that: if the hardness assumption is correct, the cryptographic scheme is secure.

3.1.2 The random oracle model

The random oracle model was first introduced by Bellare and Rogaway [1]. To analyze a cryptographic scheme in the random oracle model, one assumes that a cryptographic hash function behaves like a random function (random oracle), that is, this function always outputs the same random bit string for the same input value. All parties, including participants and adversaries, have access the random oracle. Using this assumption, one should prove the security of the scheme. Although it is less preferable to the standard model, a proof of security in the random oracle model is still acceptable when it is difficult to achieve the proof in the standard model. In this thesis, we have also proved security of our threshold blind digital signature scheme in the random oracle model.

3.2 Chosen-target CDH problem and assumption

To construct and prove security of cryptographic schemes, one normally has to base on a reasonable computational assumption. As in [25], the assumptions can be categorized into three types. The first type is the intractability of inverting problem such as factoring a composite number, inverting of RSA function, computing the discrete logarithm problem and computing the Diffie-Hellman problem. The second one is the intractability of the decision problem such as the decision Diffie-Hellman problem. And the last one is a new class of problems called Gap problems proposed by Okamoto and Pointcheval [25]. An example is Gap Diffie-Hellman problem mentioned in Chapter 2. In this thesis, our proposed signature scheme works on a *GDH* group where the DDH problem is easy, therefore we cannot use the DDH problem to build the security proof. Instead of that, the security of our proposed scheme is based on a variant of the CDH problem, namely the chosen-target CDH (**CT-CDH**)

problem.

The **CT-CDH** problem is analogous to the chosen-target RSA inversion problem [3] based on which the security proof of Chaum's blind RSA digital signature scheme is performed. The difference is only the computational assumption. The chosen-target RSA inversion problem is as follows:

Let a randomly-generated RSA key pair be $\mathtt{pk} = (N, e), \mathtt{sk} = (N, d)$ where $N = pq$ for p and q are two large primes, e is a random element of $\mathbb{Z}^*_{\phi(N)}$, $ed \equiv 1 \mod \phi(N)$ with $\phi(\cdot)$ is the Euler's phi-function. An adversary, who is given $\mathtt{pk}$, has access to the "target" oracle which returns random targets in $\mathbb{Z}^*_N$ and the RSA inversion oracle $(\cdot)^d \mod N$ (i.e., can compute $(\cdot)^d \mod N$). The assumption states that there is no polynomial-time adversary who can invert any subset of targets such that the number of queries of the adversary to the RSA inversion oracle is strictly less than the number of queries he has made to the target oracle.

By the similar technique in [7], we propose a chosen-target problem and define our assumption as follows:

Definition 3.1 (CT-CDH) *Let $\mathbb{G}_1$ be* GDH *group of prime order q and P be a generator of $\mathbb{G}_1$. Let s be a random element of $\mathbb{Z}^*_q$ and $Q = sP$. Let $H : \{0,1\}^* \rightarrow \mathbb{G}_1$ be a random hash function. The adversary $\mathcal{B}$ is given input (q, P, Q, H) and has access to the target oracle $T_{\mathbb{G}_1}$ that returns a random point U_i in $\mathbb{G}_1$ and the helper oracle $\mathtt{cdh}$-$s(\cdot)$. Let q_T and q_H be the number of queries $\mathcal{B}$ made to the target oracle and the helper oracle, respectively. The advantage of the adversary attacking the chosen-target CDH problem $\mathsf{Adv}^{ct-cdh}_{\mathbb{G}_1}(\mathcal{B})$ is defined as the probability of $\mathcal{B}$ to output a set of l pairs $((V_1, j_1), (V_2, j_2), \ldots, (V_l, j_l))$, for all $i = 1, 2, \ldots, l$ $\exists$ $j_i = 1, 2, \ldots, q_T$ such that $V_i = sU_{j_i}$ where all V_i are distinct and $q_H < q_T$.*
The chosen-target CDH assumption states that there is no polynomial-time adversary $\mathcal{B}$ with non-negligible $\mathsf{Adv}^{ct-cdh}_{\mathbb{G}_1}(\mathcal{B})$.

3.3 Secure threshold blind signature scheme

In this section, we introduce a communication model as well as the detail security definitions of the secure threshold blind signature scheme.

3.3.1 Communication Model

We assume that there are n players $\{L_1, L_2, \ldots L_n\}$ participating our protocol. Each player connects to other by secure point-to-point channels. All players are connected by a broadcast channel in which when a player sends a broadcast message, all other players can receive the message and know exactly from whom the message was sent.

In our communication model, there exists an adversary who can corrupt up to $t-1$ of the n players. An adversary can be *static* or *adaptive*. A static adversary corrupts players at the beginning of the protocol and an adaptive adversary can choose players to corrupt during protocol execution. We only consider the static adversary in this thesis.

3.3.2 Definitions

Similar to the constructions of other threshold signature schemes [12], [19], [20], our proposed threshold blind signature scheme is constructed from an underlying signature scheme, namely the blind signature scheme $\mathsf{BGS} = (\mathcal{BK}, \mathcal{BS}, \mathcal{BV})$. Therefore, the security notion of corresponding (t, n)-threshold blind digital signature scheme is also changed a bit.

Firstly, we consider the blind signature BGS. Unlike the standard digital signatures, the notion of security of the blind signatures differs in two properties. The first property of the blind signatures is "*blindness*", which means during the blind signing protocol, the signer learns nothing about the messages that the user wants to get signature on. The second property is "*against one-more-forgery*" [29], [28], meaning that after the user has inter-

acted with the signer ℓ times to get the blind signatures, the user should not obtain more than ℓ signatures. We can understand this property intuitively in the electronic cash scenario: a buyer (a user) could not get more money than the amount that a bank (a signer) gave him.

Definition 3.2 (Secure BGS) *Let* $\mathsf{BGS} = (\mathcal{BK}, \mathcal{BS}, \mathcal{BV})$ *be a blind digital signature scheme. An adversary* $\mathcal{A}$ *is given the public key* $\mathtt{pk}$ *output from* $\mathcal{BK}$ *and other public information.* $\mathcal{A}$ *acts as a user to run the blind signing protocol. After* ℓ *interactions with the signer,* $\mathcal{A}$ *outputs a set of message-signature pairs. The advantage of* $\mathcal{A}$ *attacking the blind signing protocol* $\mathsf{Adv}^{blind}_{\mathsf{BGS}}(\mathcal{A})$ *is defined as the probability that* $\mathcal{A}$ *can output a set* Σ *of valid message-signature pairs, such that* $\ell < |\Sigma|$.

We say that the blind signature scheme BGS *is secure against one-more-forgery under chosen message attack if there does not exist a polynomial-time adversary* $\mathcal{A}$ *with non-negligible advantage* $\mathsf{Adv}^{blind}_{\mathsf{BGS}}(\mathcal{A})$.

A (t, n)-threshold blind signature scheme $\mathsf{TBS} = (\mathcal{TBK}, \mathcal{TBS}, \mathcal{TBV})$ for BGS consists of three protocols with the set of players $\{L_1, L_2, \ldots, L_n\}$, which are described as follows:

- $\mathcal{TBK}$ is a distributed key generation protocol performed by n players $\{L_1, L_2, \ldots, L_n\}$. In this protocol, each player takes the public input I, returns the public key Q. The private output of each player L_i is s_i, the implicit secret key corresponding to the public key Q of n players is s. If (s, Q) has same distribution with the output of $\mathcal{BK}$, $\mathcal{TBK}$ is said to be *complete successfully.*

- $\mathcal{TBS}$ is a distributed signature generation protocol performed by a subset of players, taking input a blind message M', a public input I and a private input s_i. The output is a message-signature pair (M, σ). $\mathcal{TBS}$ *completes successfully* if the output (M, σ) is the same as the output of $\mathcal{BS}$ when the same message M is given, for all $M \in \{0,1\}^*$

- $\mathcal{TBV}$ is identical to $\mathcal{BV}$ since the outputs from $\mathcal{TBS}$ and $\mathcal{BS}$ are the same.

Now we come up with the definition of the secure threshold blind digital signature. The definition includes both unforgeability (against one-more-forgery attack) and robustness. However, an attacker can corrupt up to $t-1$ of players, where $2t-1 \leq n$.

Definition 3.3 (Secure TBS) *Let* $\mathsf{TBS} = (\mathcal{TBK}, \mathcal{TBS}, \mathcal{TBV})$ *be the* (t, n)*-threshold blind digital signature scheme, where* $2t-1 \leq n$. TBS *is the robust secure threshold blind digital signature scheme if:*

1. Unforgeability. *No polynomial-time adversary who corrupts at most* $t-1$ *players, with non-negligible probability, can do one-more forgery under chosen message attack, that is an adversary cannot produce more than* ℓ *signatures after executing* $\mathcal{TBS}$ *protocol* ℓ *times with messages of his choices.*

2. Robustness. *Even there exists an polynomial-time adversary who can corrupt up to* $t-1$ *players, both* $\mathcal{TBK}$ *and* $\mathcal{TBS}$ *protocols complete successfully.*

■ **Method of proving security of the threshold digital signature scheme**. Normally, to show the security of the threshold digital signature scheme, one can show that the underlying signature scheme is secure and the *corresponding* threshold digital signature scheme is simulatatble [12]. Even the proving method originally is used to prove the security of the threshold signature scheme where the underlying signature scheme is a standard one, it still can be applied to our proof. We explain how to utilize this method to prove security of our threshold blind digital signature scheme TBS, where the blind digital signature scheme BGS presented in Section 2.3 used as the underlying signature scheme. Intuitively, we can understand this proving

method as follows. If the TBS scheme is simulatable, then given an adversary $\mathcal{A}_{\mathsf{TBS}}$ who makes one-more-forgery attack on TBS successfully, we then are able to construct an adversary $\mathcal{A}_{\mathsf{BGS}}$ who can make one-more-forgery attack on BGS successfully. We can see that when the underlying signature scheme is changed, the attacking method is also changed. However the basic idea of the proving method is unchanged: we can construct an adversary attacking the underlying signature scheme given an attacker of the corresponding threshold signature scheme if the threshold scheme is simulatable. We have the definition of simulatable condition for the TBS is as follows:

Definition 3.4 (Simulatable) *A threshold blind digital signature scheme* $\mathsf{TBS} = (\mathcal{TBK}, \mathcal{TBS}, \mathcal{TBV})$ *is simulatable if:*

1. *The* $\mathcal{TBK}$ *protocol simulatable: there exists a simulator that on the input the public key* Q*, the public output from* $\mathcal{TBK}$*, can simulate the view of the adversary on execution of* $\mathcal{TBK}$ *protocol.*

2. *The* $\mathcal{TBS}$ *protocol simulatable: there exists a simulator that on the input the public key* Q*, the message* M*, the information of* $t-1$ *corrupted players, and the signature on* M*, can simulate the view of the adversary on execution of* $\mathcal{TBS}$ *protocol which outputs the signature on* M*.*

Chapter 4
The proposed scheme

In this chapter, we present in detail our propose threshold blind signature scheme $\mathsf{TBS} = (\mathcal{TBK}, \mathcal{TBS}, \mathcal{TBV})$.

4.1 Key Generation Protocol $\mathcal{TBK}$

The Key Generation protocol $\mathcal{TBK}$ makes use of DKG proposed in [13], but we use elliptic curve notions for the discrete logarithm problem, as described in Section 2.1.

Let $\mathbb{G}_1$ be *GDH* group and P and P' be the generators of $\mathbb{G}_1$ (i.e., $P' = \alpha P$ for some $\alpha \in \mathbb{Z}_q$, P and P' have same order, and the computing α given P and P' is infeasible). Denote n players involving in the $\mathcal{TBK}$ protocol as $\{L_1, L_2, \ldots, L_n\}$. The public key and the secret key of this group of players are Q and s, respectively. The public share of the player L_i is Q_i and the corresponding secret share is s_i, for $i = 1, 2, \ldots, n$.

Each player L_i behaves as follows to generate a shared secret.

G1. At first, L_i sends its information.

- Select randomly (uniformly distributed as in [27]) a_{i0} and $b_{i0} \in \mathbb{Z}_q^*$, keep them secret.
- Pick up randomly two polynomials $f_i(x)$ and $f_i'(x)$ over $\mathbb{Z}_q$ of degree at most $t-1$ such that $f_i(0) = a_{i0}$ and $f_i'(0) = b_{i0}$. Let

$$f_i(x) = a_{i0} + a_{i1}x + \cdots + a_{i,t-1}x^{t-1}$$

and

$$f'_i(x) = b_{i0} + b_{i1}x + \cdots + b_{i,t-1}x^{t-1}.$$

The above polynomials are kept secret by each player.

– Compute and broadcast $C_{ik} = a_{ik}P + b_{ik}P'$ for $k = 0, 1, \ldots, t-1$; send $f_i(j)$ and $f'_i(j)$ *secretly* to each player L_j for $j = 1, 2, \ldots, n;\ j \neq i$.

G2. L_i receives information from other players.

(a) After receiving $f_j(i)$ and $f'_j(i)$ from L_j for $j = 1, 2, \ldots, n;\ j \neq i$, the player L_i verifies $f_j(i)$ and $f'_j(i)$ by checking

$$f_j(i)P + f'_j(i)P' \stackrel{?}{=} \sum_{k=0}^{t-1} i^k \cdot C_{jk} \tag{4.1}$$

If Eq.(4.1) is verified to be false, L_i broadcasts a *complaint* against L_j.

(b) Each player L_j, who received a complaint from player L_i, broadcasts the values $f_j(i)$ and $f'_j(i)$ satisfying Eq.(4.1).

(c) Each player marks as *disqualified* any player that either:

– received more than $t-1$ complaints at (a), or,

– answered to a complaint in (b) with values that make invalid Eq.(4.1).

G3. Build the set of non-disqualified players by denoting this by HP which means a set of honest players.

G4. Computes the secret share $s_i = \sum_{k \in HP} f_k(i)$.

G5. Each player $L_i \in HP$ broadcasts $a_{ik}P$ for $k = 0, 1, \ldots t-1$.

– Player L_i verifies the value broadcasted by other players in HP, for each $j \in HP$, verify:

$$f_j(i)P \stackrel{?}{=} \sum_{k=0}^{t-1} i^k \cdot a_{jk}P \tag{4.2}$$

If the check fails for an index j, player L_i sends a complaint against L_j by broadcasting values $f_j(i)$ and $f'_j(i)$ which satisfies Eq.(4.1) but does not satisfy Eq.(4.2).

– For player L_j, who receives at least one valid complaint as above, the other players will use Pedersen's VSS to reconstruct the values of $a_{j0}, f_j(x)$ and $a_{jk}P$ for $k = 0, 1, \ldots, t-1$. Each player in HP sets the public key of group as $Q = \sum_{i \in HP} a_{i0}P$.

After execution of the Key Generation protocol, the public key of group of players is $Q = sP$. The corresponding secret key $s = \sum_{i=1}^{n} a_{i0}$ is distributed to n players but does not appear explicitly in the protocol. Each player has the secret share s_i with the corresponding public share $Q_i = s_iP$. For the sake of convenience, we assume that there are n players in the set HP of the honest players.

4.2 Signature Generation Protocol $\mathcal{TBS}$

Let M be a message to be signed, and $H : \{0,1\}^* \rightarrow \mathbb{G}_1$ be an one-way hash function. The public key from the $\mathcal{TBK}$ protocol is $Q = sP$, where s is the implicit secret key constructed by n signers via the threshold scheme. Suppose that a user A wants to get a signature on the message M blindly from t signers. Denote t signers as S.

S1. User A chooses randomly (uniformly distributed) $r \in \mathbb{Z}_q^*$ and blinds the message M by computing $M' = rH(M)$. A broadcasts M' to all players.

S2. Signer L_i, after receiving M', computes a partial signature σ_i and broadcasts it, where $\sigma_i = s_i \omega_i M'$ and $\omega_i = \prod_{\substack{j \in S \\ j \neq i}} \frac{j}{j-i}$.

Any subset of players or a combiner can verify σ_i by computing

$$\hat{e}(\sigma_i, P) \stackrel{?}{=} \hat{e}(\omega_i M', Q_i) \tag{4.3}$$

If Eq.(4.3) does not hold for the player L_i, L_i is requested to send the correct σ_i. Otherwise, the combiner computes $\sigma' = \sum_{i \in S} \sigma_i$ and sends back σ' to the user A.

S3. User A unblinds σ' to get the signature on M.

$$\sigma = r^{-1} \sigma' \tag{4.4}$$

Figure 4.1 depicts the signature generation protocol.

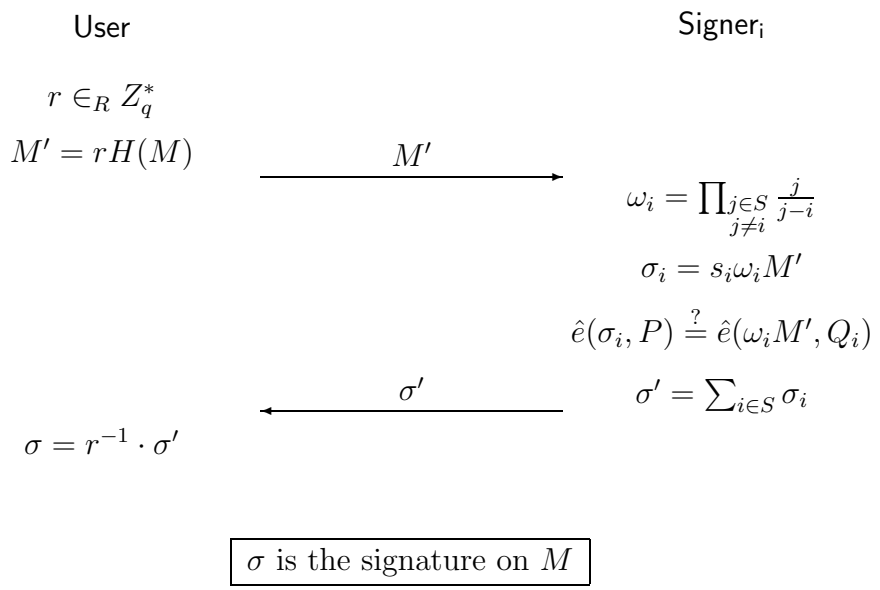

Figure 4.1: Threshold blind signature scheme - $\mathcal{TBS}$ protocol

4.3 Signature Verification Protocol $\mathcal{TBV}$

The signature σ on a message M is accepted if and only if:

$$\hat{e}(\sigma, P) = \hat{e}(H(M), Q) \tag{4.5}$$

4.4 Correctness

We now present the correctness of our signature scheme. Firstly, the correctness of the signature scheme must involve the correctness of the verification of Eq.(4.3) in the $\mathcal{TBS}$ protocol. That means the partial signature σ_i is valid if the i-th signer is honest. We have:

$$\begin{aligned} \hat{e}(\sigma_i, P) &= \hat{e}(\omega_i M', Q_i) \\ &= \hat{e}(\omega_i M', s_i P) \\ &= \hat{e}(\omega_i s_i M', P) \end{aligned}$$

Secondly, we verify the correctness of the threshold blind signature scheme. The scheme signature σ has a form:

$$\begin{aligned} \sigma &= r^{-1} \sum_{i \in S} \sigma_i \\ &= r^{-1} \sum_{i \in S} s_i \prod_{\substack{j \in S \\ j \neq i}} \frac{j}{j-i} \cdot M' \qquad (4.6) \\ &= r^{-1} r s H(M) \qquad (4.7) \\ &= s H(M) \end{aligned}$$

We can derive Eq.(4.7) from Eq.(4.6) by Lagrange interpolation.

The verification using Eq.(4.5) gives us:

$$\begin{aligned} \hat{e}(\sigma, P) &= \hat{e}(H(M), Q) \\ &= \hat{e}(H(M), sP) \\ &= \hat{e}(sH(M), P) \end{aligned}$$

Hence, if σ is the valid signature on M, the verification always holds.

Chapter 5
Security analysis

In this chapter, we discuss about the security aspects of our proposed scheme by describing the blindness, the robustness and the unforgeability of the signature scheme. We give complete security proof of our proposed threshold blind signature scheme using the definitions introduced in the previous chapter.

5.1 Blindness

The blindness of the proposed threshold blind signature scheme is shown by the following theorem:

Theorem 5.1 (Blindness) *The threshold blind digital signature scheme* TBS *exhibits the blind property.*

Proof: By the similar method in [7], we can show that the proposed signature scheme is blind. Since r is chosen randomly from $\mathbb{Z}_q^*$, therefore $M' = rH(M)$ is also a random element in the group $\mathbb{G}_1$. Thus signers only receive the random information from the user and there is no way to know the original message. The signers also cannot link between the information they received and the message-signature pair which is output by the user. ■

5.2 Robustness

The robustness of the proposed signature scheme is shown by the following theorem:

Theorem 5.2 *The threshold blind signature scheme* TBS *is robust for an adversary who can corrupt* $t-1$ *signers among* n *signers such that* $n \geq 2t-1$ *signers.*

Proof: As in [27], every signer chooses randomly a secret a_{i0} uniformly distributed in $\mathbb{Z}_q^*$ during $\mathcal{TBK}$ protocol. Therefore, even there exists an adversary who can corrupt up to $t-1$ signers among $n \geq 2t-1$ signers, any subset of t signers constructs the unique secret key s uniformly distributed in $\mathbb{Z}_q^*$, thus the public key Q is uniformly distributed in $\mathbb{G}_1$ too. That means $\mathcal{TBK}$ completes successfully in case at most $t-1$ signers are corrupted.

In the $\mathcal{TBS}$ protocol, every partial signature σ_i is verified by the corresponding public key $Q_i = s_i P$. Even at most $t-1$ signers can be corrupted, the adversary still needs partial signatures from other signers to form t valid signature shares. With t valid signature shares, the signature $\sigma = sH(M)$ can be produced by Eq.(4.4) at step S3 of $\mathcal{TBS}$, and its correctness was shown in Section 4.4. Therefore, the $\mathcal{TBS}$ protocol completes successfully too. Thus we prove TBS to be robust. ■

5.3 Unforgeability

To show the unforgeability of the threshold digital signature, one can show that the underlying signature scheme is unforgeable and the corresponding threshold digital signature scheme is simulatatble [12]. We utilize this method to prove the unforgeability of our threshold blind digital signature scheme TBS as mentioned in Section 3.3.2. The proving process consists two steps:

showing the simulatable condition of TBS as given in Definition 3.4 and showing that BGS signature scheme is unforgeable.

First, we consider the simulatable condition. The similatable condition means that there exists a probabilistic polynomial-time simulator which can simulate the view for every probabilistic polynomial-time adversary $\mathcal{A}$. In other word, it is polynomially indistinguishable the view of $\mathcal{A}$ in running of $\mathcal{TBK}$ and $\mathcal{TBS}$ protocols from the output produced by the simulator. Since $\mathcal{TBK}$ protocol has utilized DKG [13] which is simulatable and proved secure in presence of an adversary who can corrupt up to $t-1$ players, we only consider the simulator for $\mathcal{TBS}$ protocol. Denote the view of the adversary $\mathcal{A}$ during $\mathcal{TBS}$ as $\mathcal{VIEW}_{\mathcal{A}}(\mathcal{TBS}(s_1, s_2, \ldots, s_n, (M, Q)), \sigma)$. We construct a simulator *SIM* which interacts with $\mathcal{A}$ to generate the signature shares of the honest players. Without loss of generality [12], we can assume that $\mathcal{A}$ corrupts the first $t-1$ players, and *SIM* knows all the secret shares but the last one (i.e., *SIM* knows $s_t, s_{t+1}, \ldots, s_{n-1}$ and does not know s_n). The input to *SIM* is a public key Q, a message M, a signature σ on M and secret shares $s_1, s_2, \ldots, s_{t-1}$ of the corrupted signers. The *SIM* works as follows:

1. *SIM* chooses $r' \in \mathbb{Z}_q^*$ randomly (uniformly distributed).
2. *SIM* interacts with $\mathcal{A}$ and computes the partial signature on behalf of the honest players $\sigma_i' = r' s_i \omega_i H(M)$ for $t \leq i \leq n-1$.
3. For the player L_n, whose share *SIM* does not know, *SIM* computes the partial signature as $\sigma_n' = r'\sigma - \sum_{i \in H_1} \sigma_i'$ where H_1 is any subset of $t-1$ players.

Let $\mathcal{SIM}(M, Q, s_1, s_2 \ldots, s_{t-1}, \sigma)$ be the information produced by the above simulator *SIM*. The following theorem shows the simulatable condition of the $\mathcal{TBS}$ protocol:

Theorem 5.3 *$\mathcal{VIEW}_{\mathcal{A}}(\mathcal{TBS}(s_1, s_2 \ldots, s_n, (M, Q), \sigma))$ and $\mathcal{SIM}(M, Q, s_1, s_2, \ldots, s_{t-1}, \sigma)$ have the same probability distribution.*

Proof: By comparing the information produced by *SIM* and $\mathcal{TBS}$ protocol we have:

1. Both the protocol and the simulator choose a blind factor randomly from $\mathbb{Z}_q^*$, r in $\mathcal{TBS}$ and r' in *SIM*. The probability distribution of r and r' are the same.

2. All partial signatures produced by $\mathcal{TBS}$ contain the blind factor r and the shared secret s_i, $1 \leq i \leq n$. The simulator *SIM* also produces the partial signatures σ'_i for $t \leq i \leq n$ and can verified $t-1$ partial signatures of the corrupted signers (controlled by $\mathcal{A}$). The correctness of the partial signatures produced by the corrupted signers can be verified using their public information output from $\mathcal{TBK}$ protocol. All partial signatures produced by *SIM* except σ'_n contain the blind factor r' and the shared secrets which are uniformly distributed. The partial signature σ'_n is computed from any set of the $t-1$ partial signatures which are embedded the blind factor and the shared secrets. Hence, σ_n also has the right distribution. Therefore, the view of $\mathcal{A}$ in the running of $\mathcal{TBS}$ and the one in the interaction with *SIM* are polynomially indistinguishable.

This completes the proof. ■

Now, we consider the blind signature presented in Section 2.3.

Under the assumption that the **CT-CDH** problem is hard for all groups where the CDH problem is hard, including the *GDH* groups, we will show in the random oracle model the blind signature proposed in Section 2.3 is secure by the following theorem:

Theorem 5.4 *If the chosen-target CDH assumption is true in the group* $\mathbb{G}_1$ *then the blind signature scheme* BGS *is secure against one-more forgery under chosen message attack in the random oracle model.*

Proof: Let $\mathcal{A}$ be a polynomial time adversary attacking BGS against one-more-forgery under chosen message attack. We will construct a polynomial time adversary $\mathcal{B}$ solving the **CT-CDH** problem such that $\mathsf{Adv}^{blind}_{\mathsf{BGS},I}(\mathcal{A}) = \mathsf{Adv}^{ct-cdh}_{\mathbb{G}_1}(\mathcal{B})$.

The adversary $\mathcal{A}$ has access to a blind signing oracle $\mathtt{cdh}$-$s(\cdot)$ and a random hash oracle $H(\cdot)$. Then the adversary $\mathcal{B}$ can solve the chosen-target CDH problem by simulating $\mathcal{A}$. Firstly, $\mathcal{B}$ provides $\mathtt{pk} = (q, P, H, Q)$ to $\mathcal{A}$ and $\mathcal{B}$ has to simulate the random hash oracle and the blind signing oracle for $\mathcal{A}$.

Each time $\mathcal{A}$ makes a new hash oracle query which differs from the previous ones, $\mathcal{B}$ will forward to its target oracle and return the reply to $\mathcal{A}$. $\mathcal{B}$ stores the pair query-reply in the list of those pairs. If $\mathcal{A}$'s query is the same as the previous ones, $\mathcal{B}$ will take and send the corresponding reply which $\mathcal{B}$ has stored before.

If $\mathcal{A}$ makes a query to the blind signing oracle, $\mathcal{B}$ will forward to its helper oracle $\mathtt{cdh}$-$s(\cdot)$ and returns the answer to $\mathcal{A}$.

At some point, the adversary $\mathcal{A}$ produces a list of message-signature pairs $((M_1, \sigma_1), (M_2, \sigma_2), \ldots, (M_l, \sigma_l))$. $\mathcal{B}$ can find M_i in the list stored hash oracle query-reply for $i = 1, 2, \ldots, l$. Let j_i be the index of the found pair, then $\mathcal{B}$ can output its list as $((\sigma_1, j_1), (\sigma_2, j_2), \ldots, (\sigma_l, j_l))$.

In the view of $\mathcal{A}$, the above simulation and the real protocol are indistinguishable and easily we can see that $\mathcal{B}$ is successful only if $\mathcal{A}$ is successful. Therefore, we have $\mathsf{Adv}^{blind}_{\mathsf{BGS},I}(\mathcal{A}) = \mathsf{Adv}^{ct-cdh}_{\mathbb{G}_1}(\mathcal{B})$. ■

Theorem 5.5 *The threshold blind signature scheme* TBS *is secure against one-more forgery under chosen message attack in the random oracle model if the chosen-target CDH assumption is true in the group* $\mathbb{G}_1$.

Proof: The proof can be easily derived from Theorems 5.3 and 5.4. ■

Theorem 5.6 *The threshold blind digital signature scheme* TBS *is a robust secure* (t, n)*-threshold blind digital signature scheme.*

Proof: The proof of the theorem comes immediately from Theorems 5.2 and 5.5. ■

Chapter 6
Comparison with other schemes

This chapter evaluates performance of the proposed scheme. The following tables show the comparison of computation in the Signature Generation protocol with that of other schemes.

Operation	KKL01 scheme	LJY99 scheme	Our scheme
$\mathsf{A_m}$	$2t+1$	$2t+1$	0
M	$t+5$	$2n-t+6$	0
E	6	8	0
I	0	0	1
A	N/A	N/A	0
S	N/A	N/A	2

Table 6.1: Computation in the user side

In Tables 6.1 and 6.2, $\mathsf{A_m}$, M, E and I mean modular addition, multiplication, exponentiation and inversion, respectively. A and S denote point addition and scalar multiplication on an elliptic curve over a finite field. KKL01 and LJY99 schemes are the threshold blind signature schemes in [19] and [20] based on discrete logarithm problems. N/A means Not Applicable.

For the comparison, we assume that the standard binary method is used for the computing of the modular exponentiation and the scalar point multiplication in polynomial basis form. With q is a 140-bit prime and using

Operation	KKL01 scheme	LJY99 scheme	Our scheme
$\mathsf{A_m}$	2	$2(n-t+1)$	0
M	5	$2n-1$	1
E	8	6	0
I	0	0	0
A	N/A	N/A	$t-1$
S	N/A	N/A	1

Table 6.2: Computation in the signer side

the well-known facts in [4], [23], we can roughly estimate the computation cost of the proposed scheme compared to those of the previous ones, where $1\mathsf{S} \approx 1028\mathsf{M}, 1\mathsf{A} \approx 14\mathsf{M}, 1\mathsf{E} \approx 210\mathsf{M}$. And we also assume that the computation of modular additions is negligible.

In our proposed scheme, to produce a signature, a user needs to perform 1 modular inversion and 2 point multiplications. The computational overhead is less efficient than those of KKL01 and LJY99 due to the heavy scalar multiplications.

However, as being shown in Table 6.2, the computation cost in the signer side of our scheme including the signature combination is more efficient compared to those of KKL01 and LJ99. A signer has to compute 1 modular multiplication, $t-1$ point additions, and 1 scalar multiplication. The computation cost in the signer side is lessened because there is only one scalar multiplication performed. The verification of the partial signatures is done in the signer side too.

Any verifier only needs to perform 2 pairing computations to verify the signature σ.

Since our threshold blind signature scheme works on an elliptic curve, the advantage of the scheme Compared to the previous schemes [19] and [20] is the efficiency in the signature size as well as the signature shares.

In fact, this size can be reduced into half if we use the point compression techniques [16]. Moreover, since the underlying signature scheme is different, the communication overhead in our scheme is reduced as well.

Chapter 7
Conclusions and further work

In this thesis, we have studied the design and the analysis of secure digital signatures schemes, in particular, a blind digital signature scheme and a corresponding (t, n)-threshold digital signature scheme. We have reviewed previous works and then presented a new construction.

We have proposed a new robust secure threshold blind digital signature scheme based on bilinear pairings. A threshold blind digital signature scheme combines a threshold digital signature scheme and a blind one to get both signature schemes' properties. That is, a threshold blind digital signature scheme distributes possessions of the secret key in a group of signers while lets a user get signatures on messages without revealing messages' content to signers. Our construction is working on an elliptic curve over a finite field and as far as we know, this is the first construction of a threshold blind digital signature based on bilinear pairings in the literature.

To guarantee sound security of the construction, first we have identified possible attacks and then established appropriated security model to prove security of the construction. We have used the random oracle model as a tool to show that any attacker, who corrupts up to $t - 1$ players and can breaks the threshold blind digital signature scheme, can be transformed into an efficient algorithm to solve the underlying problem, namely "chosen-target CDH problem". Finally, we have presented the security proof for all well-defined security requirements.

Our construction has also achieved an efficiency in signature size compared to those of the previous schemes.

The proposed threshold blind signature scheme can be applied in any application utilizing blind signatures and the secret key should be distributed to enhance security. The typical example is an Internet voting system with multiple administrators where votes are blind signed by several administrators for validation purpose.

In future work, we can add proactive property to our signature scheme using techniques [15], [14]. This property makes the signature scheme more secure by coping with mobile adversary. Using DKG, we can achieve proactive property more secure, since original techniques used insecure distribution method as pointed out in [13].

References

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

2. P.S.L.M. Barreto, H.Y. Kim, B. Lynn and M. Scott, "Efficient Algorithms for Pairing-Based Cryptosystems", *Advances in Cryptology – Crypto'2002*, LNCS 2442, Springer–Verlag, pp. 354-369, 2002.

3. M. Bellare, C. Namprempre, D. Pointcheval and M. Semanko, "The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme", *Cryptology ePrint Archive – 2001/02*.

4. I. Blake, G. Seroussi and N. Smart, *Elliptic Curves in Cryptography*, Cambridge University Prress, LNS 265, 1999.

5. D. Boneh and M. Franklin, "ID-based Encryption from the Weil-pairing", *Advances in Cryptology – Crypto'2001*, LNCS 2139, Springer–Verlag, pp. 213–229, 2001.

6. D. Boneh, B. Lynn, and H. Shacham, "Short Signatures from the Weil-pairing", *Advances in Cryptology – Asiacrypt'2001*, LNCS 2248, Springer–Verlag, pp. 514–532, 2001.

7. A. Boldyreva, "Threshold Signature, Multisignature and Blind Signature Schemes Based on the Gap-Diffie-Hellman-group Signature Scheme", *Public Key Cryptography - PKC 2003*, LNCS 2567, Springer–Verlag, pp. 31–46, 2003.

8. D. Chaum, "Blind Signatures for Untraceable Payments", *Proc. of Crypto'82*, LNCS 1440, pp. 199–203, Springer–Verlag, 1983.

9. Y. Desmedt and Y. Frankel, "Threshold Cryptosystems", *Advances in Cryptology – Crypto'89*, LNCS 435, pp. 307–315, Springer–Verlag, 1990.

10. W. Diffie and M. Hellman, "New directions in cryptography", *IEEE Trans. on Information Theory* Vol.22, No.6, pp.644–654, 1976.

11. P. Feldman, "A Practical Scheme for Non-interactive Verifiable Secret Sharing", In Proc. 28th FOCS, pp. 427–437, 1987.

12. R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, "Robust Threshold DDS Signatures, *Advances in Cryptology – Eurocrypt'96*, LNCS 1070, Springer–Verlag, pp. 354-371, 1996.

13. R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, "Secure Distributed Key Generation for Discrete-log Based Cryptosystems", *Advances in Cryptology – Eurocrypt'99*, LNCS 1592, Springer–Verlag, pp. 295-310, 1999.

14. A. Herzberg, M. Jakobsson, S. Jarecki, H. Krawczyk, and M. Yung, "Proactive public key and signature systems", *ACM Conference on Computers and Communication Security – CCS'97*, ACM Press, pp. 100–110, 1997.

15. A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung, "Proactive Secret Sharing Or: How to Cope With Perpetual Leakage",*Advances in Cryptology – Crypto'95*, LNCS 963, Springer–Verlag, pp. 339-352, 1999.

16. F. Hess, G. Seroussi and N. Smart, "Two Topics in Hyperelliptic Cryptography", *Selected Areas in Cryptography – SAC'2001*, LNCS 2259, Springer–Verlag, pp. 181-189, 2001.

17. A. Joux and K. Nguyen, "Separating Decision Diffie-Hellman from Diffie-Hellman in Cryptographic Groups", *Cryptology ePrint Archive – 2001/03*.

18. M. Kim, and K. Kim, "A New Identification Scheme Based on the Bilinear Diffie-Hellman Problem", *ACISP2002*, LNCS 2384, Springer-Verlag, pp. 362378, 2002.

19. J. Kim, K. Kim and C. Lee, "An Efficient and Provably Secure Threshold Blind Signature", *International Conference on Information Security and Cryptology – ICISC'2001*, LNCS 2288, Springer–Verlag, pp. 318-327, 2002.

20. C.L. Lei, W.S. Juang and P.L. Yu, "Provably Secure Blind Threshold Signatures Based on Discrete Logarithm", *National Computer Symposium 1999*, pp. C198–C205, 1999.

21. A. J. Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, 1993.

22. A. J. Menezes, T. Okamoto, and S. A. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field", *IEEE Trans. Inform. Theory*, 39(1993), pp. 1639–1646.

23. A. J. Manezes, P. C.van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC press, 1996.

24. M. Ohkubo, F. Miura, M Abe, A. Fujioka and T. Okamoto, "An Improvement on a Practical Secret Voting Scheme", *Information Security Workshop - ISW'99*, LNCS 1729, Springer–Verlag, pp. 225-234, 1999.

25. T. Okamoto and D. Pointcheval, "The gap-problem: a new class of problems for the security of cryptographic schemes", *PKC 2001*, LNCS 1992, Springer-Verlag, pp. 104–118, 2001.

26. T.P. Pedersen, "A Threshold Cryptosystem without a Trusted Party", *Advances in Cryptology – Eurocrypt'91*, LNCS 547, Springer–Verlag, pp. 522–526, 1991.

27. T.P. Pedersen, "Non-interactive and Information-theoretic Secure Verifiable Secret Sharing", *Advances in Cryptology – Crypto'91*, LNCS 576, Springer–Verlag, pp. 129–140, 1991.

28. D. Pointcheval and J. Stern, "Provably Secure Blind Signature Schemes", *Advances in Cryptology – Asiacrypt'96*, LNCS 1163, Springer–Verlag, pp. 252–265, 1996.

29. D. Pointcheval and J. Stern, "Security Argument for Digital Signatures and Blind Signatures", *Journal of Cryptology*, Springer–Verlag, Vol. 13 No. 3, pp. 361–396, 2000.

30. A. Shamir, "How to Share a Secret", Commmunication of the ACM, Vol. 22, No. 11, pp. 612–613, Nov. 1979.

31. V. Shoup, "Practical Threshold Signatures", *Advances in Cryptology – Eurocrypt'2000*, LNCS 1807, Springer–Verlag, pp. 207–220, 2000.

32. D.R. Stinson, *Cryptography: Theory and Practice*, CRC Press, 1995.

33. David Wagner, "Generalized Birthday Problem", *Advances in Cryptology – Crypto'02*, LNCS 2442, Springer–Verlag, pp. 288-304, 2002.

34. F. Zhang and K. Kim, "ID-Based Blind Signature and Ring Signature from Pairings", *Advances in Cryptology – Asiacrypt'2002*, LNCS 2501, Springer–Verlag, pp. 533–547, 2002.

Acknowledgements

First, I would like to express my gratitude to Prof. Kwangjo Kim, my thesis advisor, for giving me a chance to study in ICU. I would like to thank him for his continuous encouragement, constant direction and support. Without his guidance, I must have been unable to carry out my research. Throughout my days in ICU, he gave me many lessons about how to live, to work and to study. I am specially thankful to Prof. Younghee Lee and Prof. C. Pandu Rangan for their generosity and agreeing to serve as advisory committee members. I also would like to thank Prof. Cheon for his assistance when he was in ICU as well as when he moved.

I would like express my thanks to my seniors in Cryptology and Information Security Laboratory. They were Prof. Byoungcheon Lee, Gookwhan An, Manho Lee, Jinho Kim, Jaegwan Park, Myungsun Kim, Jongseong Kim, Wooseok Ham, Hyunrok Lee, Jungyeun Lee. They have helped me so much in living and studying in ICU. They always give me most valuable advices and encouragements. I would also like to thank my lab members Kyusuk Han, Yan Xie, Dang Nguyen Duc, Sangwon Lee, Jin Kim, Hyungki Choi, Byunggon Kim, Songwon Lee, Hwasun Chang, JaeHyuk Park, Soogil Choi, Divyan Munirathnam, Chooljoon Choi, Joongman Kim, Sungmin, Chen Xiaofeng. Their kind and sincere support lead me to complete this thesis. I also thanks Jeongmi Choi for her helpful support as a staff member.

In addition, I would like to give my thanks to my Vietnamese friends in ICU and Vietnam for their help and encouragement. I also would like to thank Jungsoo Lee, Jeongbae Park in CN lab. for their help and encouragement. I also would like to send my thanks to Academic and Student Department of ICU for their kindly help.

My love and thanks go to my parents who not only gave birth to me but also have had hard lives in bringing me up. I would like to thank them for their endless love and devotion. I also would like to thank family-in-law for their continuous encouragement and soundly moral support when I have to live far from my family.

Last, but not least, I greatly appreciate my lovely wife and daughter. I would like to thank my wife for her love, her understanding and her constant encouragement.

Finally, I would like to dedicate this thesis to my parents, my wife and daughter, especially to my grand-mother whom I could not have any chance to meet. I cherish my dear memories to her.

Curriculum Vitae

Name : Vo Duc Liem

Date of Birth : Sep. 19. 1975

Sex : Male

Nationality : Vietnamese

Education

1992.9–1997.6	IT Engineer Hanoi University of Technology (B.S.)
2001.6–2003.8	Information Security Group, Engineering Information and Communications University (M.S.)

Career

2001.6–	Graduate Research Assistant Cultivation of top level IT security manpower The Ministry of Information and Communications
2001.6–2002.8	Graduate Research Assistant Votopia - Electronic Voting System for World Cup 2002 Information Research center for Information Security

Publications

International Papers (in English)

1. Duc Liem Vo, Fangguo Zhang, and Kwangjo Kim, "A New Threshold Blind Signature Scheme Based on Pairings", *SCIS2003*, vol. 1/2, pp. 233–238, 2003.

2. Duc Liem Vo, and Kwangjo Kim, "Provably Secure Threshold Blind Signature Scheme Based on Pairings", *Submitted to CT-RSA2004*.

Domestic Papers (in English)

1. Hyunrok Lee, Duc Liem Vo, and Kwangjo Kim, "Extension of Votopia to Mobile Voting", *Proceedings of Conference on Information Security and Cryptology – CISC2002*, pp. 371–374, *KIISC*, 2002.

2. Duc Liem Vo, and Kwangjo Kim, "Design of Threshold Blind Signature Scheme", *Proceedings of Conference on Information Security and Cryptology – CISC2003*, pp. 37–42, *KIISC*, 2003.