

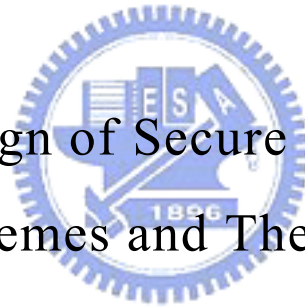
國立交通大學

資訊工程學系

博士論文

安全數位盲簽章機制之設計與應用

The Design of Secure Digital Blind
Signature Schemes and Their Applications



研究生：吳林全 Student: Lin-Chuan Wu
指導教授：葉義雄 Advisor: Yi-Shiung Yeh

中華民國九十四年十一月

安全數位盲簽章機制之設計與應用

學生：吳林全

指導教授：葉義雄

國立交通大學資訊工程學系博士班

摘要

近年來由於網際網路應用快速地發展，使得網路購物和網路競標等電子交易服務日漸普及。目前這些服務所採用的認證方式大多為身份-密碼 (ID-Password) 機制，因其不具備不可否認 (non-repudiation) 的性質。因此，植基於公開金鑰基礎建設 (PKI) 之數位簽章機制能夠達到交易上之不可否認性，建立電子商務應用和服務之穩定基礎。

然而，在電子現金或電子投票等應用中，須額外滿足使用者對匿名性 (anonymity) 的要求，以保障使用者的隱私權。因此，數位盲簽章機制的設計即是要解決此一問題，以提供使用者達到不可追蹤性 (untraceability) 目的，使得在計算上簽章之簽署者事後無法識別所簽署之簽章是由何人所持有；換句話說就是要追蹤出該簽章的持有人在計算上是不可行的。

本論文主要是提出偽造即停盲簽章機制 (fail-stop blind signature scheme) 來解決傳統盲簽章機制在面對擁有無限計算能力的偽造者總是能夠成功地偽造簽章，且對偽造即停盲簽章機制所須具備的安全性質加以定義，並證明所提出之簽章機制是安全的。

本論文亦針對現有各種植基於整數分解、二次剩餘以及離散對數之盲簽章機制，提出一些在安全上和效率上的改善方法。同時也探討代理盲簽章機制之不可偽造性 (unforgeability) 和不可追蹤性 (untraceability) 等安全議題。最後，提出具備偽造即停盲簽章機制之電子現金系統和具備資訊隱藏和不可追蹤性之電子票卷協定，期能建構更安全的電子交易系統之理論基礎和應用服務。



關鍵字：不可追蹤性，偽造即停盲簽章機制，盲簽章機制，密碼學，資訊安全

The Design of Secure Digital Blind Signature Schemes and Their Applications

Student: Lin-Chuan Wu

Advisor: Yi-Shiung Yeh

Institute of Computer Science and Information Engineering

National Chiao Tung University

The logo of National Chiao Tung University is a circular emblem with a gear-like border. Inside the circle, there is a stylized building and the year '1896' at the bottom. The word 'Abstract' is overlaid on the logo.

Abstract

Recently, Internet applications are developed rapidly, such that electronic transaction services like purchasing and bidding on Internet are more popular. The ID-Password mechanism is mainly used for authentication, but it cannot achieve the non-repudiation property. Therefore, the digital signature scheme based on PKI can achieve the non-repudiation property in electronic transactions. It can be the well-constructed basis for electronic commerce services and applications.

However, in electronic cash or electronic ticket applications, the anonymity property must be satisfied for the participants to preserve their

privacy. Thus, the digital blind signature scheme is proposed for this purpose. The untraceability property is an important property in digital blind signature scheme, it makes the signer computationally cannot identify the signature which is owned by someone. In the other words, the signer is computationally infeasible to trace the signature.

In this dissertation, a fail-stop blind signature scheme is proposed to solve the problem that a forger with more powerful computational capability can always forge a signature successfully. A secure fail-stop blind signature scheme is also defined. Moreover, our proposed signature scheme is proved secure.



Some improved digital blind signature schemes, in security and efficiency, based on integer factorization, quadratic residue, and discrete logarithm cryptosystems are also be presented in this dissertation. Furthermore, the unforgeability and untraceability properties of proxy blind signature schemes are discussed. Finally, an electronic cash system based on fail-stop blind signature scheme and an electronic ticket protocol with information hiding are proposed. They can be established for more secure electronic transaction systems in theoretical basis and applications.

關鍵字: *Untraceability, Fail-stop Blind Signature Scheme, Blind Signature Scheme, Cryptography, Information Security.*



Contents

CHAPTER 1 INTRODUCTION	1
1.1 MOTIVATIONS.....	1
1.2 RESEARCH OBJECTIVES AND CONTRIBUTIONS	2
CHAPTER 2 DIGITAL SIGNATURE SCHEMES	4
2.1 RIVEST-SHAMIR-ADLEMAN SIGNATURE SCHEME.....	4
2.2 ELGAMAL SIGNATURE SCHEME	7
2.3 RABIN SIGNATURE SCHEME.....	9
2.4 CHAUM BLIND SIGNATURE SCHEME	11
2.5 SUSILO-SAFAVI-PIEPRZYK FAIL-STOP SIGNATURE SCHEME	13
2.6 MAMBO-USUDA-OKAMOTO PROXY SIGNATURE SCHEME	16
CHAPTER 3 ANALYSIS OF SOME BLIND SIGNATURE SCHEMES	20
3.1 CRYPTANALYSIS ON A NEW RABIN-LIKE BLIND SIGNATURE SCHEME.....	20
3.1.1 <i>Chen et al.'s Blind Signature Scheme</i>	23
3.1.2 <i>Cryptanalysis on Chen et al.'s Scheme</i>	25
3.2 RSA-BASED PARTIALLY BLIND SIGNATURE SCHEME	27
3.2.1 <i>Chien et al.'s scheme</i>	28
3.2.2 <i>Hwang et al.'s Traceability Attack</i>	30
3.2.3 <i>Analysis of Hwang et al.'s Attack</i>	32
3.3 UNTRACEABLE ELGAMAL BLIND SIGNATURE SCHEME	34
3.3.1 <i>Camenisch et al.'s scheme</i>	35
3.3.2 <i>Lee et al.'s Traceability Attack</i>	36
3.3.3 <i>Analysis of Lee et al.'s Attack</i>	37
3.4 THE SECURE PROXY BLIND SIGNATURE SCHEMES	39

3.4.1	<i>The Proxy Blind Signature Schemes</i>	42
3.4.1.1	Tan et al.'s proxy blind signature schemes	42
3.4.1.2	Lal and Awasthi's proxy blind signature scheme	44
3.4.2	<i>Sun et al.'s Traceability Attack</i>	46
3.4.2.1	Sun et al.'s attack on Tan et al.'s schemes	46
3.4.2.2	Sun et al.'s attack on Lal-Awasthi's scheme.....	47
3.4.3	<i>Analysis of Sun et al.'s Attack</i>	48
3.4.3.1	Analysis of Sun et al.'s attack on Tan et al.'s schemes.....	48
3.4.3.2	Analysis of Sun et al.'s attack on Lal-Awasthi's scheme.....	50

CHAPTER 4 THE PROPOSED DIGITAL BLIND SIGNATURE SCHEMES 52

4.1	A FAIL-STOP BLIND SIGNATURE SCHEME	52
4.1.1	<i>The Proposed Blind Signature Scheme</i>	53
4.1.2	<i>Security Analysis</i>	55
4.2	THE ENHANCED GENERIC BLIND SIGNATURE SCHEME	61
4.3	THE ENHANCED BLIND SIGNATURE SCHEME BASED ON THE ELLIPTIC CURVE CRYPTOSYSTEM	63
4.3.1	<i>Yeh-Chang's Blind Signature Scheme</i>	64
4.3.2	<i>The Enhanced Signature Scheme</i>	66
4.3.3	<i>Security Analysis</i>	68
4.3.4	<i>Performance Comparison</i>	71

CHAPTER 5 APPLICATIONS OF SOME BLIND SIGNATURE SCHEMES 73

5.1	THE UNTRACEABLE FAIL-STOP ELECTRONIC CASH SCHEME	74
5.1.1	<i>Chaum's Untraceable Electronic Cash Scheme</i>	75
5.1.2	<i>The Proposed Electronic Cash Scheme</i>	77
5.1.3	<i>Security Analysis</i>	80

5.2 AN UNTRACEABLE ELECTRONIC TICKET SCHEME FOR INFORMATION HIDING 82

 5.2.1 *The Proposed Electronic Ticket Scheme* 83

 5.2.2 *Security Analysis* 85

CHAPTER 6 CONCLUSIONS **87**

Bibliography **88**



List of Figures

FIGURE 2.1	BLOCK DIAGRAM OF RSA SIGNATURE SCHEME	5
FIGURE 2.2	PROTOCOL DIAGRAM OF RSA SIGNATURE SCHEME	7
FIGURE 2.3	BLOCK DIAGRAM OF ELGAMAL SIGNATURE SCHEME	8
FIGURE 2.4	PROTOCOL DIAGRAM OF ELGAMAL SIGNATURE SCHEME	9
FIGURE 2.5	BLOCK DIAGRAM OF RABIN SCHEME.....	10
FIGURE 2.6	PROTOCOL DIAGRAM OF RABIN SCHEME	11
FIGURE 2.7	PROTOCOL DIAGRAM OF CHAUM SIGNATURE SCHEME.....	13
FIGURE 2.8	PROTOCOL DIAGRAM OF PARTIAL DELEGATION PROXY SIGNATURE SCHEME	19



List of Tables

TABLE 4.1	THE COMPARISON OF REQUIRED STORAGE REQUIREMENTS	72
TABLE 4.2	THE COMPARISON OF REQUIRED OPERATIONS	72



Chapter 1 Introduction

1.1 Motivations

Due to Internet applications are developed rapidly, such that electronic transaction services like that purchasing and bidding on Internet are more popular. These applications are mainly using the ID-Password mechanism for authentication, but this mechanism cannot achieve the non-repudiation property. For protect the users against malicious parties, some advanced techniques to enhance the security of the electronic transaction services are required. Therefore, the digital signature scheme based on Public Key Infrastructure (PKI) can achieve the non-repudiation property. It is also the key component for electronic commerce services and applications.

Although the digital signature scheme can achieve the non-repudiation property, it cannot provide the privacy for the users. In some applications like electronic cash or electronic ticket systems, the anonymity property is very important and should be satisfied. Thus, the digital blind signature scheme is proposed to ensure the unforgeability for the signer and achieve the untraceability for the users. The untraceability property makes the

signer computationally cannot identify the signature which is owned by someone. Hence, the signer is computationally infeasible to trace the signature.

However, the traditional digital blind signature schemes cannot protect the signer against a forger with more powerful computational capability to forge a signature. This means that there is no mechanism to protect the signer against a forged signature which has succeeded in signature verification. Namely, if a signed message succeeds in signature verification it is assumed to be generated by the owner of the private key. Thus, a fail-stop blind signature scheme is proposed to solve this problem in this dissertation.



Recently, a lot of misunderstandings on digital blind signature schemes and proxy blind signature schemes are submitted. They claim that some blind signature schemes cannot satisfy the untraceability property. However, these claims are incorrect and they will be analyzed and corrected.

1.2 Research Objectives and Contributions

In this dissertation, a secure fail-stop blind signature scheme based on the integer factorization is defined, proposed and proved. It can be applied in more critical system like electronic payment systems which need higher security against more powerful forger and can preserve the users' privacy. Furthermore, some misunderstanding claims on digital blind signature schemes are discussed and corrected in detail. The untraceability property of the proxy blind signature schemes is also analyzed in this dissertation. Finally, some more secure electronic transaction systems are designed by using our proposed schemes.



Chapter 2 Digital Signature Schemes

The ordinary handwritten signature is used to specify the responsibility of the person and can achieve the non-repudiation property. A digital signature scheme is a method to sign the message in electronic form and can provide analogous to the ordinary handwritten signature. Any digital information including digital signatures can be copied easily, so digital signatures cannot be the digitalized version of handwritten signatures. To overcome this problem, digital signature schemes are designed by using mathematical functions and interactive protocols. The following sections describe the various digital signature schemes in detail.

2.1 Rivest-Shamir-Adleman Signature Scheme

The concept of digital signature scheme was introduced by Diffie and Hellman [12] in 1976. Generally, a digital signature scheme has the signing algorithm and the verification algorithm. The fundamental idea is that everyone has pair of keys: a signing/private key and a verification/public key. The signing key is to sign the message by using

the signing algorithm and the verification key is to verify the correctness of the signature by using the public verification algorithm. Especially, the verification key can be published and the signing key must be kept secretly.

In 1978, Rivest, Shamir, and Adleman [43] proposed the first digital signature scheme based on the integer factorization problem. The signer and the requester are two kinds of participants in RSA signature scheme. The four phases in RSA signature scheme are: (1) Initialization, (2) Requesting, (3) Signing, (4) Verification. Initially, the signer publishes the necessary information for the participants. In the requesting phase, the requester sends the message to the signer. The signer signs on that message in the signing phase. Finally, anyone can verify the correctness of the signature using the message-signature pair in the verification phase.

Figure 2.1 shows the block diagram of RSA signature scheme for signing and verification. The detailed signature scheme is described as follows.

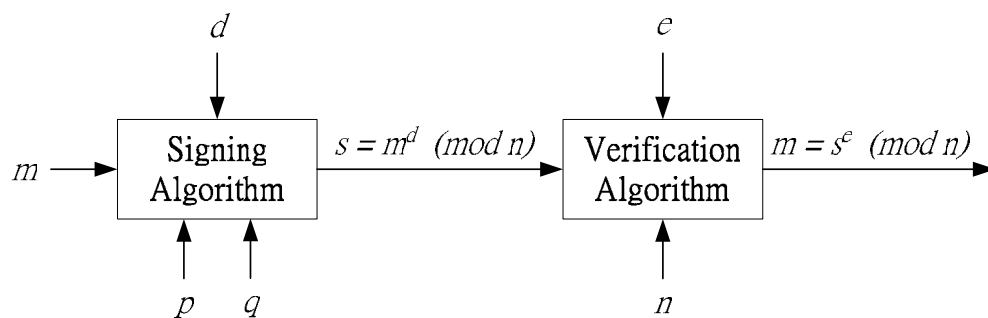


Figure 2.1 Block diagram of RSA signature scheme

(1) Initialization: The signer randomly selects two large primes p and q , and calculates $n = p \cdot q$ and $\phi(n) = (p-1) \cdot (q-1)$. Next, the signer selects a large random number $1 < d < \phi(n)$ such that $e \cdot d \equiv 1 \pmod{\phi(n)}$. Thus, d is the private key of the signer and e is the public key.

(2) Requesting: The requester prepares the message m and sends it to the signer.

(3) Signing: The signer calculates the signature $s = m^d \pmod{n}$ on the message m and sends s to the requester.

(4) Verification: Anyone can verify the correctness of the signature s received from the requester by checking whether $s^e = m \pmod{n}$ because e is public.

The protocol diagram of RSA signature scheme is illustrated in Figure 2.2.

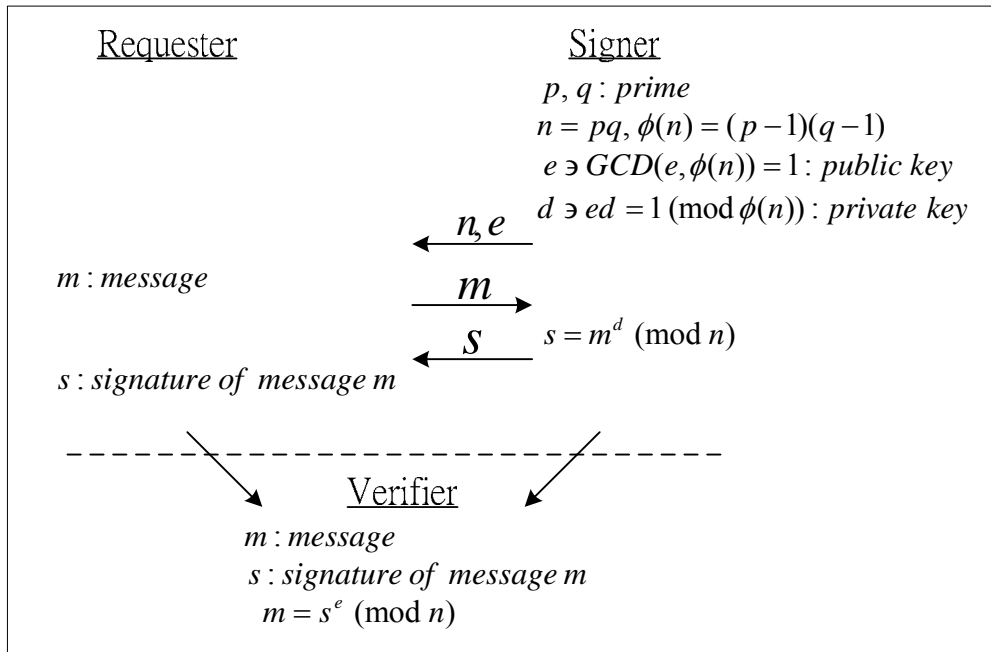


Figure 2.2 Protocol diagram of RSA signature scheme

Anyone can forge a signature by using multiplication attack in RSA signature scheme. To prevent this attack, hash function can be used within the signature scheme to reduce the problem.



2.2 ElGamal Signature Scheme

ElGamal [13] presented another digital signature scheme in 1985. The security of ElGamal scheme is based on the difficulty of computing discrete logarithm. There are many valid signatures for any given message in ElGamal scheme, and any of these valid signatures are authentic by the verification algorithm. Thus, ElGamal is called the

non-deterministic signature scheme. The major shortcoming in ElGamal scheme is the double length of any message. The block diagram of ElGamal signature scheme for signing and verification is shown by Figure 2.3.

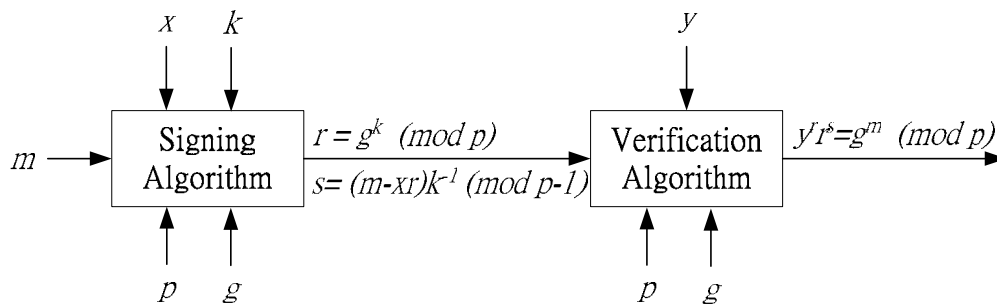


Figure 2.3 Block diagram of ElGamal signature scheme

The four phases of ElGamal scheme are described in the following.

(1) **Initialization:** The signer randomly chooses a prime number p such that discrete logarithm problem in Z_p^* is intractable. Let $g \in Z_p^*$ be a primitive root and x be the private key of the signer. The public key of the signer is defined by $y = g^x \pmod{p}$.

(2) **Requesting:** The requester sends the message m to the signer.

(3) **Signing:** The signer selects a random number k . Then s/he can compute $r = g^k \pmod{p}$ and $s = k^{-1}(m - xr) \pmod{p-1}$. The (r, s) is the signature on the message m .

(4) **Verification:** Anyone can verify the correctness of the signature (r, s) by checking whether $y^r r^s = g^m \pmod{p}$ is true.

Figure 2.4 illustrates the protocol diagram of ElGamal signature scheme.

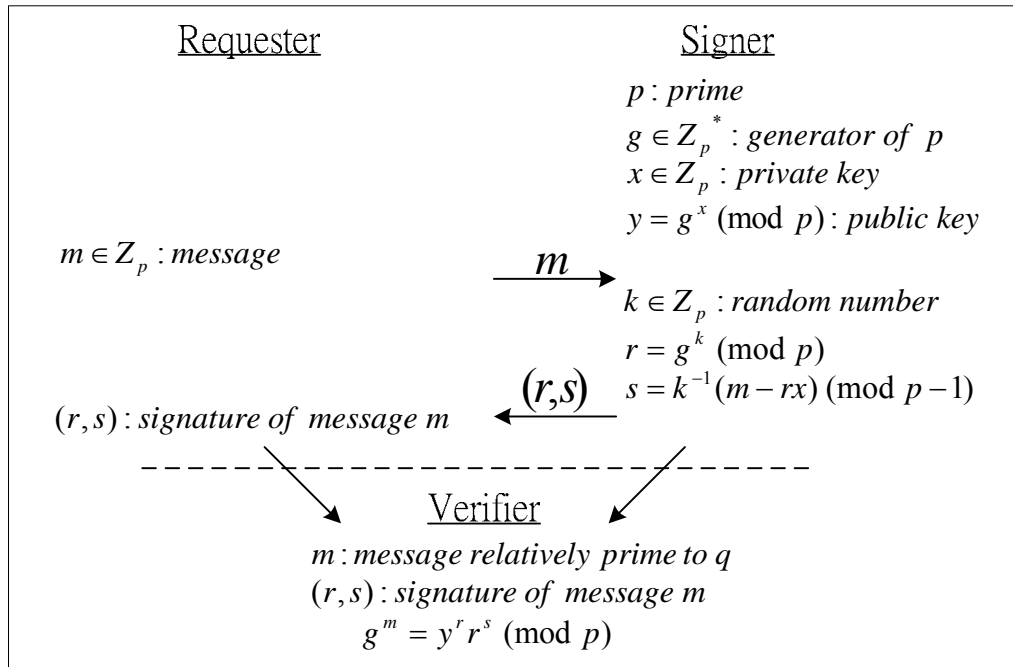


Figure 2.4 Protocol diagram of ElGamal signature scheme

2.3 Rabin Signature Scheme

In 1979, Rabin [40] proposed a signature scheme based on the quadratic residue problem. The security of Rabin scheme is based on the difficulty of computing square root modulo a composite number. Rabin scheme is computationally secure against chosen-plaintext attack. Figure 2.5 shows the block diagram of Rabin signature scheme for signing and verification and the details are described as follows.

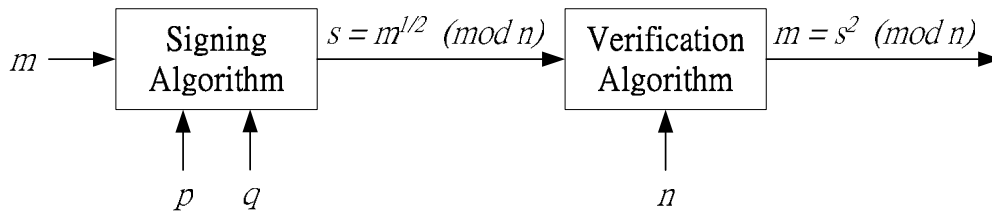


Figure 2.5 Block diagram of Rabin scheme

(1) **Initialization:** The signer can select two random prime numbers p and q , where $p \equiv 3 \pmod{4}$ and $q \equiv 3 \pmod{4}$. Then s/he calculates $n = p \cdot q$ and $\phi(n) = (p-1) \cdot (q-1)$. Thus, p and q are the private key and n is the public key.

(2) **Requesting:** The requester prepares the message m and sends it to the signer.



(3) **Signing:** The signature is the square root of the message m . Thus, the signer can calculate the signature $s = m^{1/2} \pmod{n}$ and sends s to the requester.

(4) **Verification:** Anyone can verify the signature s by checking whether $s^2 = m \pmod{n}$ is true.

The protocol diagram of Rabin signature scheme is illustrated detailed in Figure 2.6.

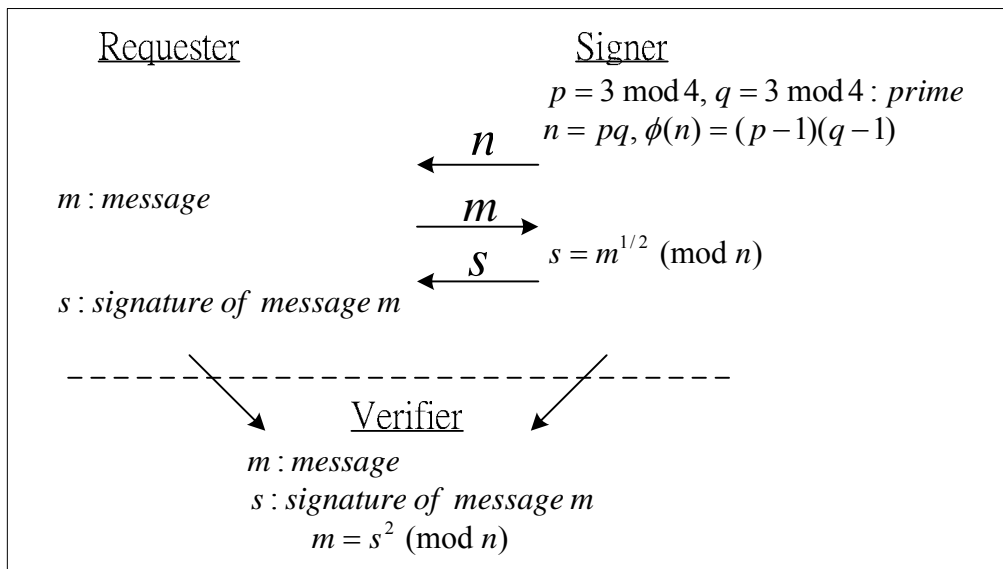


Figure 2.6 Protocol diagram of Rabin scheme

2.4 Chaum Blind Signature Scheme

Chaum [8] presented the first blind signature scheme based on RSA digital signature scheme in 1982. The blind signature scheme allows a requester to obtain a message signed by the signer without revealing message. Therefore, the signer cannot link/trace any message-signature pair practically later. The blind signature scheme can be used in electronic payment systems or electronic voting systems to preserve the participants' anonymity. The detailed scheme is described in the following.

(1) Initialization: The signer chooses two large primes p and q

randomly, and computes $n = pq$ and $\phi(n) = (p-1)(q-1)$. Then, the signer selects two random numbers e and d such that $ed \equiv 1 \pmod{\phi(n)}$, where $1 < e < \phi(n)$ and $1 < d < \phi(n)$. Finally, the signer publishes (n, e) as his public key and a one-way hash function H like SHA-1.

(2) Blinding and requesting: The requester selects a random number r as the blinding factor, where $r \in Z_n^*$. Then, the requester sends the blinded message $\tilde{m} = r^e H(m) \pmod{n}$ to the signer.

(3) Signing: After the signer receives the blind message \tilde{m} , s/he calculates $\tilde{s} = \tilde{m}^d$ and sends it to the requester.

(4) Unblinding: The requester can compute the signature $s = r^{-1} \tilde{s} \pmod{n}$ from the blinded signature \tilde{s} .

(5) Verification: Anyone can easily verify the message-signature pair (m, s) by checking that $s^e = H(m) \pmod{n}$ is true.

The signer cannot recognize which messages was actually signed and know which blind signatures was actually generated due to the blinding factor r . Therefore, Chaum blind signature scheme can achieve the unlinkability/untraceability property.

Figure 2.7 illustrates the protocol diagram of Chaum blind signature

scheme.

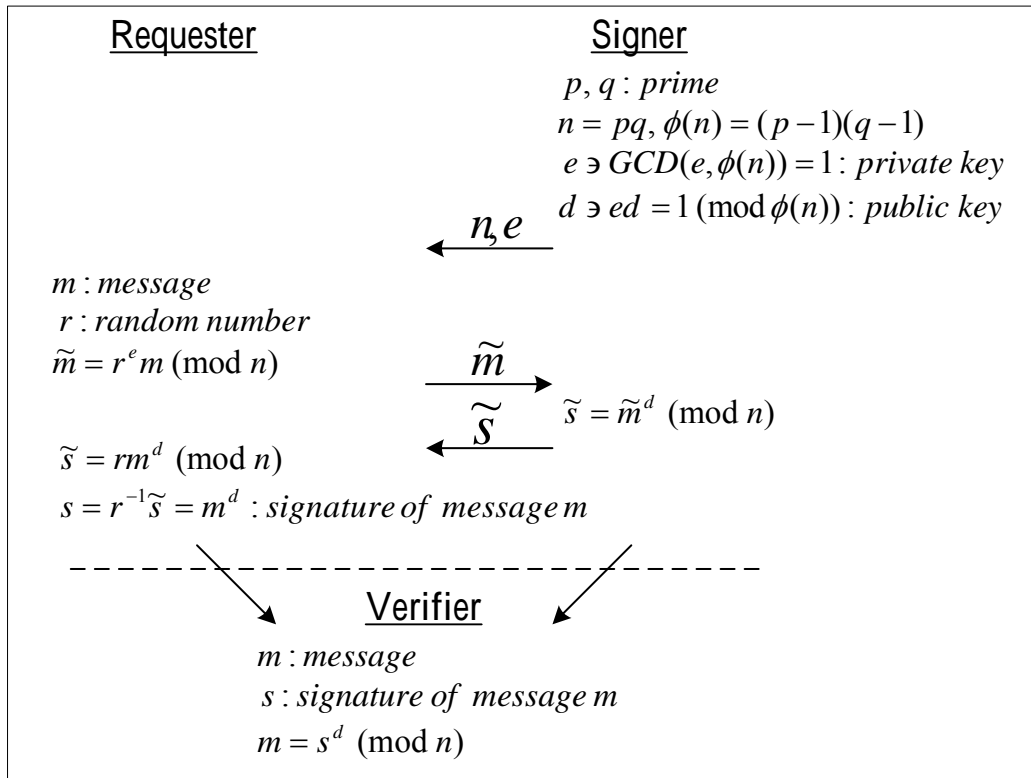


Figure 2.7 Protocol diagram of Chaum signature scheme

2.5 Susilo-Safavi-Pieprzyk Fail-stop Signature Scheme

The traditional digital signature schemes cannot protect the signer against a forger with more powerful computational capability to forge a signature. This means that there is no mechanism to protect the signer against a forged signature which has succeeded in signature verification. Namely, if a signed message succeeds in signature verification it is assumed

to be generated by the owner of the private key.

To overcome this kind of attack, Waidner and Pfitzmann [50, 38] proposed the first fail-stop signature scheme. Fail-stop signature can protect a signer against a forger even with more powerful computational capability because the possibility of finding the signer's right private key in the fail-stop signature is negligible. The signer can use "proof of forgery" algorithm to prove the signature is forgery. It achieves "proof of forgery" by showing that the underlying computational assumption has been broken. The signer can stop the system if a forgery occurs – hence named fail-stop signature scheme. The signer is unconditionally secure and the requester is cryptographically secure in the fail-stop signature scheme.



In 1992, van Heyst and Pedersen constructed a fail-stop signature scheme based on the discrete logarithm problem [46] and their scheme is a Lamport-like one-time signature [26]. Susilo, Safavi-Naini and Pieprzyk [48] presented two RSA-based fail-stop signature schemes with and without a trusted dealer in 1999. We only consider the scheme with trusted dealer here for simplicity. Actually, the signer and the receiver can instead of trusted dealer to perform the initialization phase by using Boneh-Franklin's algorithm [3]. There are three kinds of participants, which are the trusted

dealer, the sender and the receiver in the Susilo et al.'s scheme with trusted dealer. A forged signature can be proved by using Miller's [33] and Bach's [2] methods to reveal non-trivial factors for the signer. The detailed scheme is described as follows.

(1) Initialization: The two large prime numbers p and q are chosen by the trusted dealer D , such that $p = 2p'+1$ and $q = 2q'+1$, where p' and q' are also prime. Then, D computes $n = pq$ and $\phi(n) = (p-1)(q-1)$. Next, D chooses d_D as her/his private key and computes $e_D = d_D^{-1} \bmod \phi(n)$, where $GCD(d_D, \phi(n)) = 1$. Then, D selects a random number $\alpha \in Z_n^*$ and computes $\beta = \alpha^{d_D} \bmod n$. Finally, D publishes her/his public key (α, n) and sends (e_D, β) to the signer S securely.

(2) Key generation: The signer S selects four random numbers, which are k_1, k_2, k_3 and k_4 as the private key, where $k_i \in Z_n^*$, $1 \leq i \leq 4$. Next, S computes $\beta_1 = \alpha^{k_4} \beta^{k_3} \bmod n$, $\alpha_1 = \alpha^{k_3} \beta_1^{k_1} \bmod n$ and $\alpha_2 = \alpha^{k_4} \beta_1^{k_2} \bmod n$. Finally, s/he publishes her/his public key $(\beta_1, \alpha_1, \alpha_2)$.

(3) Signature generation: The signer S computes $y_1 = k_1x + k_2$ and $y_2 = k_3x + k_4$, where $x \in Z_n^*$ is a message. Then, s/he publishes the signature (y_1, y_2) on message x .

(4) Signature verification: The receiver R can verify the signature by checking the formula $\alpha^{y_2} \beta_1^{y_1} = \alpha_1^x \alpha_2 \pmod{n}$. If it is true, this signature is a valid one.

(5) Proof of forgery: If a forged signature (y_1', y_2') on message x succeeds in signature verification phase, S can prove that a forgery has occurred by executing the following steps.

1. To construct the right signature (y_1, y_2) on message x .
2. To compute $Z_1 = (y_1' - y_1)$ and $Z_2 = (y_2 - y_2')$.
3. To compute $\gamma = e_D(Z_2 - k_4 Z_1) - k_3 Z_1 = c\phi(n)$
4. To find non-trivial factors of n by using Miller's [33] and Bach's [2] methods.
5. The non-trivial factors of n is the proof of forgery.

2.6 Mambo-Usuda-Okamoto Proxy Signature Scheme

The proxy signature scheme based on the discrete logarithm problem was presented by Mambo et al. [30] in 1996. It can allow the designated proxy signer to sign messages on behalf of the original signer. For

example, when a manager is going on a vacation, s/he can delegate her/his secretary to sign the messages on behalf of her/him. There are three types of delegation: full delegation, delegation by warrant and partial delegation in the proxy signature scheme.

In full delegation, the original signer gives her/his private key to the designated proxy signer and then the original signer and the proxy signer can both generate the same signatures. However, the signatures generated by the original signer and the proxy signer are not distinguishable. Thus, the dispute between the original signer and the proxy signer on the signature cannot be settled.



The warrant is used to show that the proxy signer is legal and to describe the needed information between the original signer and the proxy signer in delegation by warrant. It can be implemented by using ordinary signature scheme. However, it needs to execute the proxy signature verification process and then the ordinary signature verification process.

In partial delegation, the original signer uses her/his private key to generate the proxy secret key and sends it to the proxy signer securely. The signatures can be distinguished from the original signer and the proxy signer. Thus, partial delegation scheme is more practical than full

delegation scheme and more efficient than delegation by warrant scheme.

We describe Mambo et al.'s partial delegation proxy signature scheme in detail as follows.

(1) Initialization: The original signer randomly chooses a large prime number p and a generator $g \in Z_p^*$. Let x be the private key of the original signer and y be the corresponding public key such that $y = g^x \text{ mod } p$.

(2) Proxy delegation: The original signer randomly selects a number k_o , and calculates $r_o = g^{k_o} \text{ mod } p$ and $s_o = x + k_o r_o \text{ mod } (p-1)$. Next, the original signer sends (r_o, s_o) to the proxy signer in a secure manner. After the proxy signer receives (r_o, s_o) , s/he can verify it by checking the correctness of the equation $g^{s_o} = y r_o^{r_o} \text{ mod } p$. If (r_o, s_o) satisfies that equation, s/he can accept it as a valid proxy. Finally, the proxy signer computes her/his proxy secret key $s_{pr} = s_o + x_p \text{ mod } q$.

(3) Requesting: The requester sends the prepared message m to the signer.

(4) Signing: The proxy signer chooses k randomly, and computes $r = g^k \text{ mod } p$ and $s = k^{-1}(m - xr) \text{ mod } (p-1)$, where m is the message to be signed.

(5) **Verification:** Anyone can verify the correctness of the signature (r, s) by checking that the equation $y^r r^s = g^m \pmod p$ holds.

Figure 2.8 illustrates the protocol diagram of Mambo et al.'s proxy signature scheme.

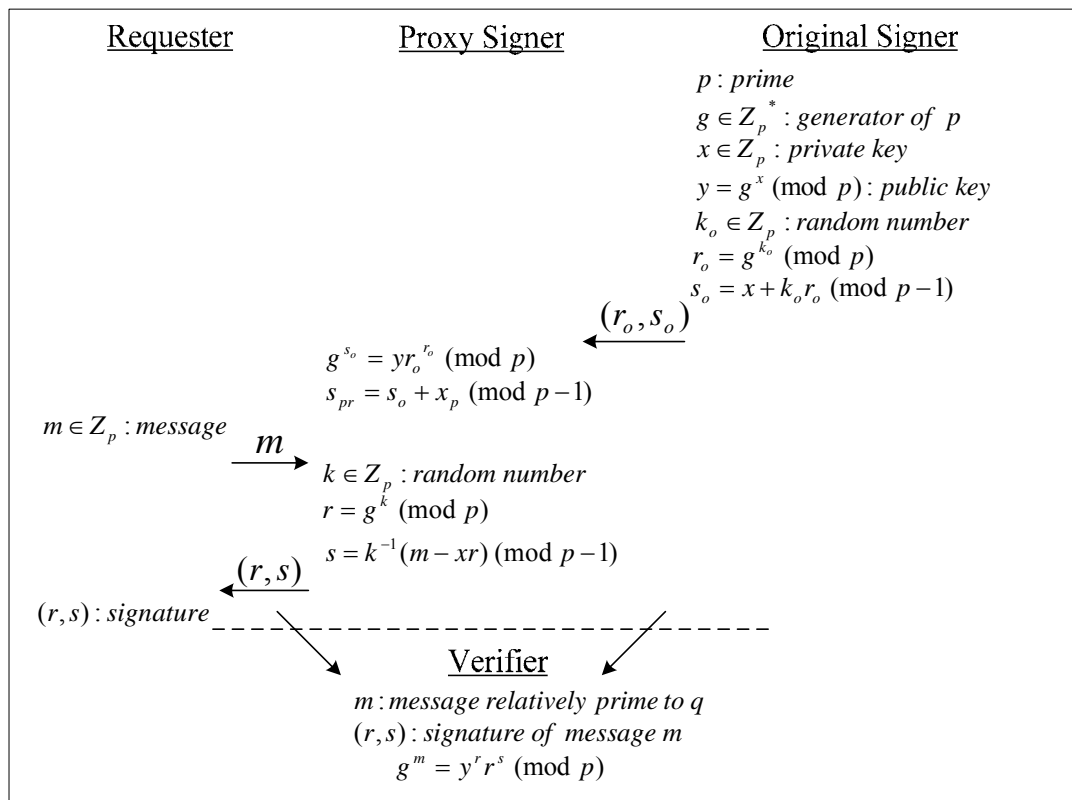


Figure 2.8 Protocol diagram of Partial delegation proxy signature scheme

Chapter 3 Analysis of Some Blind Signature Schemes

In **Section 3.1**, we introduce the cryptanalysis on a new Rabin-like blind signature scheme based on the quadratic residue problem. A traceability attack on RSA-Based partially signature with low computation is analyzed and corrected in detail In **Section 3.2**. Lee et al. claimed that ElGamal blind signature scheme is traceable but we show that their claims are incorrect in **Section 3.3**. Finally, we analyze Sun et al.'s traceability attack on proxy blind signature scheme in **Section 3.4**.



3.1 Cryptanalysis on a New Rabin-like Blind Signature Scheme

The Rabin digital signature scheme [40] is based on the square-root problem. Its security is relying on the difficulty of finding the square roots of a quadratic residue under a modulus n and it has been proved to be as hard as factoring n [40]. Compared to the RSA cryptosystem [43], the signature verification only requires one modular multiplication.

The blind signature scheme was proposed by Chaum [7] and it is based on the RSA cryptosystem [43]. In addition to the unforgeability of the signatures, it must satisfy two requirements: (1) the contents of messages are unknown to the signer when signing and (2) the signer cannot trace the signed messages after the senders have revealed the signatures publicly. Because of the unlinkability property, blind signature can protect the senders' privacy in digital transactions and it can be applied in electronic voting systems and electronic cash systems.

Recently, Chen et al. [10] proposed a new Rabin-like blind signature scheme, which is based on the square-root problem. Although their scheme is simple and efficient, it can be compromised when choosing some particular blinding factors. In this section, we propose an attack on Chen et al.'s scheme and demonstrate that their scheme is not secure.

Let $Z_n^* = \{k \in Z_n \mid GCD(k, n) = 1\}$ be the multiplicative group under modulus n , where n is a positive integer. An integer a is called a quadratic residue (QR) in Z_n^* , if there exists an integer $x \in Z_n^*$ such that $x^2 \equiv_n a$. If no such x exists, a is called a quadratic non-residue (QNR) in Z_n^* . The set of all quadratic residues under modulus n is denoted by Q_n and the set of all quadratic non-residues under modulus n is denoted

by $\overline{Q_n}$. That is, $Q_n = \{a \in Z_n^* \mid \exists x \in Z_n^*, x^2 \equiv_n a\}$ and $\overline{Q_n} = Z_n^* - Q_n$ [32, 44, 46].

Let p be an odd prime and let α be a generator in Z_p^* . An integer $a \in Z_p^*$ is a quadratic residue modulo p if and only if $a \equiv_p \alpha^i$ where i is an even integer. It follows that $|Q_p| = |\overline{Q_p}| = (p-1)/2$, i.e. half of the elements in Z_p^* are QR's and the other half are QNR's.

Let p be an odd prime and a be an integer. The Legendre symbol

$\left(\frac{a}{p}\right)$ is defined below.

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & , \text{ if } p \mid a \\ 1 & , \text{ if } a \in Q_p \\ -1 & , \text{ if } a \in \overline{Q_p} \end{cases}$$



Let n be a product of two distinct odd primes p and q , i.e., $n = p \cdot q$. An integer $a \in Z_n^*$ is a quadratic residue under modulo n if and only if $a \in Z_p^*$ and $a \in Z_q^*$. Therefore, $|Q_n| = |Q_p| \parallel |Q_q| = (p-1)(q-1)/4$ and $|\overline{Q_n}| = 3(p-1)(q-1)/4$.

Let $n \geq 3$ be an odd integer with prime factorization $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ and let a be an integer. The Jacobi symbol [32] is defined below.

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)^{e_1} \left(\frac{a}{p_2}\right)^{e_2} \cdots \left(\frac{a}{p_k}\right)^{e_k}$$

Let $n \geq 3$ be an odd integer and $J_n = \left\{ a \in Z_n^* \mid \left(\frac{a}{n}\right) = 1 \right\}$.

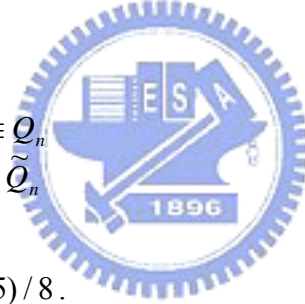
$\tilde{Q}_n = J_n - Q_n$ is defined to be the set of pseudo-squares under modulus n .

Thus the Jacobi symbol is a generalization of the Legendre symbol [32]

where n is not necessarily to be prime.

Let $n = p \cdot q$ be a Blum integer, i.e., p and q are distinct primes and $p \equiv_4 3$ $q \equiv_4 3$. If $x \in Q_n$, then $x^{(n-p-q+5)/8} \pmod n$ is a square root of x . If $x \in J_n$, then

$$x^{2d} = \begin{cases} x & , \text{ if } x \in Q_n \\ n-x & , \text{ if } x \in \tilde{Q}_n \end{cases}$$



where $d = (n-p-q+5)/8$.

Let $n = p \cdot q$ be a Williams integer [32], i.e., p and q are distinct primes and $p \equiv_8 3$ and $q \equiv_8 7$. Thus, 2 is a quadratic non-residue under modulus n with Jacobi symbol $\left(\frac{2}{n}\right) = -1$. Hence, multiplication of any integer x by 2 or $2^{-1} \pmod n$ reverses the Jacobi symbol of x .

3.1.1 Chen et al.'s Blind Signature Scheme

Chen, Qiu and Zheng presented a new blind signature scheme [10]

based on Rabin's signature scheme. There are two kinds of participants, senders and the signer in the blind signature scheme. The sender requests signatures from the signer, and the signer issues signatures on the blinded messages to the sender. The protocol consists of three phases: (1) requesting, (2) signing and (3) extraction. A sender submits a blinded message to the signer in the requesting phase to obtain a signature. In the signing phase, the signer computes the signature on the blinded message and returns the result to the sender. Finally, the sender extracts the signature from the result that he received in the extraction phase.

Let $n = p \cdot q$ be a Williams integer and (p, q) be kept secret by the signer. Let H be a one-way hash function. The details of the scheme are described as follows.

(1) Requesting: To request the signature of the message m , the sender computes $H(m)$. Then s/he randomly chooses the blinding factor $r \in Z_n^*$. The sender chooses appropriate bits a and b .

$$a = \begin{cases} 0 & , \text{ if } \left(\frac{H(m)}{n} \right) = 1 \\ 1 & , \text{ if } \left(\frac{H(m)}{n} \right) = -1 \end{cases}$$

such that $2^{-a} H(m) \bmod n \in J_n$, and sends the blinded message

$\tilde{m} = 2^{-a} r^4 H(m) \bmod n$ to the signer.

(2) **Signing:** After the signer receives \tilde{m} , s/he computes $\tilde{s} = (2^{-a} r^4 H(m))^d \bmod n$ where $d = (n-p-q+5)/8$ is the private key of the signer, and sends \tilde{s} back to the sender.

(3) **Extraction:** The sender computes $s = (\tilde{s} r^{-2}) \bmod n$ and forms (s, a, b) such that $s^2 (-1)^b 2^a \equiv_n H(m)$. One can verify the correctness of the signature (s, a, b) on the message m by checking the formula $s^2 (-1)^b 2^a \equiv_n H(m)$.



3.1.2 Cryptanalysis on Chen et al.'s Scheme

In this section, we demonstrate that Chen-Qiu-Zheng scheme [10] is not secure against the chosen-ciphertext attack.

Theorem 3.1: Given two integers x and y in Z_n^* , where $n = p \cdot q$ is a Blum integer. If $x^2 \equiv_n y^2$ and $x \not\equiv \pm y \pmod n$, then $GCD(x+y \bmod n, n) = p$ or q .

Proof: By the Chinese remainder theorem, an integer w in Z_n^* can be represented by $\langle w_1, w_2 \rangle$, where $w_1 = (w \bmod p)$ and $w_2 = (w \bmod q)$.

For each $k = \langle k_1, k_2 \rangle$ and $w = \langle w_1, w_2 \rangle$ in Z_n^* ,

$$\langle k + w \bmod n \rangle = \langle k_1 + w_1 \bmod p, k_2 + w_2 \bmod q \rangle$$

$$\langle k \cdot w \bmod n \rangle = \langle k_1 \cdot w_1 \bmod p, k_2 \cdot w_2 \bmod q \rangle$$

$$\langle k^{-1} \bmod n \rangle = \langle k_1 \bmod p, k_2 \bmod q \rangle$$

$$\langle -k \bmod n \rangle = \langle -k_1 \bmod p, -k_2 \bmod q \rangle$$

Besides, for every $\langle k_1, k_2 \rangle$ and $\langle w_1, w_2 \rangle$ in Z_n^* ,
 $\langle k_1, k_2 \rangle = \langle w_1, w_2 \rangle$ if and only if $k_1 = w_1 \bmod p$ and $k_2 = w_2 \bmod q$.

Let $x = \langle x_1, x_2 \rangle$ in Z_n^* , where $x_1 = (x \bmod p)$ and $x_2 = (x \bmod q)$,
and let $t = (x^2 \bmod n)$. The integer t has four square roots
 $\{\langle x_1, x_2 \rangle, \langle x_1, -x_2 \rangle, \langle -x_1, x_2 \rangle, \langle -x_1, -x_2 \rangle\}$, where $y = \langle -x_1, x_2 \rangle$ or
 $y = \langle x_1, -x_2 \rangle$ since $x \neq (\pm y \bmod n)$. If $y = \langle -x_1, x_2 \rangle$, then
 $(x + y \bmod n) = \langle 0, 2x_2 \bmod q \rangle$. Hence, $(x + y \bmod n)$ can be divided by
 p and $GCD(x + y \bmod n, n) = p$. If $y = \langle x_1, -x_2 \rangle$, then
 $(x + y \bmod n) = \langle 2x_1 \bmod p, 0 \rangle = \langle 2x_1 \bmod p, 0 \rangle$. Thus, $(x + y \bmod n)$
can be divided by q and $GCD(x + y \bmod n, n) = q$. \square

In Chen-Qiu-Zheng scheme, someone tries to compromise this
scheme, s/he can send $(2^{-a} r^2 h(m) \bmod n)$, instead of $(2^{-a} r^4 h(m) \bmod n)$ to
the signer without being detected by the signer since it is blinded, and then
obtains $\tilde{s} = ((2^{-a} r^2 h(m))^d \bmod n)$. The integer \tilde{s} is a square root of

$(2^{-a} r^2 h(m) \bmod n)$ with probability $1/2$, and $(\tilde{s} r^{-1} \bmod n)$ is a square root of $(2^{-a} h(m) \bmod n)$ with probability $1/2$, too. Then, the sender randomly selects another \hat{r} , and sends $(2^{-a} \hat{r}^2 h(m) \bmod n)$ to the signer, so that he can receive $\hat{s} = ((2^{-a} \hat{r}^2 h(m))^d \bmod n)$. If the integer $(\hat{s} \hat{r}^{-1} \bmod n)$ is a square root of $(2^{-a} h(m) \bmod n)$ and different from $(\pm \tilde{s} r^{-1} \bmod n)$ where the probability is $\frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4}$, then $GCD(\tilde{s} r^{-1} + \hat{s} \hat{r}^{-1} \bmod n, n)$ is one of the prime factors of n by **Theorem 3.1**. This kind of chosen-ciphertext attack can compromise Chen-Qiu-Zheng scheme.



3.2 RSA-Based Partially Blind Signature Scheme

In AsiaCrypt'96, Abe and Fujisaki [1] submitted the first partially blind signature scheme to inject the common information, like the date, on the signature. Chien et al. [11] proposed more efficient RSA-based partially blind signature scheme than Abe-Fujisaki's scheme later. Recently, Hwang et al. [21] claimed Chien et al.'s scheme cannot meet the untraceability property of the blind signature. In this section, we show that Hwang et al.'s claim is incorrect and Chien et al.'s scheme is still the untraceable scheme.

Recently, Chien et al. proposed RSA-based partially blind signature with low computation for mobile and smart-card applications. Hwang et al. claimed that Chien et al.'s scheme cannot meet the untraceability property of the blind signature later. In this section, we show that Hwang et al.'s claim is incorrect and Chien et al.'s scheme is still satisfy the untraceability property.

3.2.1 Chien et al.'s scheme

In 2001, Chien et al. proposed an efficient partially blind signature based on RSA cryptosystem. To compare with Abe-Fujisaki's scheme, Chien et al.'s scheme can reduce the amount of computations by almost 98% for the requester. Therefore, Chien et al.'s scheme is suitable for mobile client and smart-card applications.

The signer and the requester are two kinds of participants in the Chien's partially blind signature. The requester obtains a partially blind signature from the signer and the signer cannot link any message-signature pair later. The four phases in Chien et al.'s scheme are (1) Initialization, (2) Requesting, (3) Signing, (4) Extraction and verification. Initially, the signer initially publishes the necessary information for participants. In

the requesting phase, the requester sends a blinded message and the agreed common information to the signer. The signer signs on the blinded message with the common information in the signing phase. Finally, the requester obtains the signature from the blinded signature without removing the injected common information in the extraction and verification phase. Anyone can verify the correctness of the signature using the message-signature pair and the agreed common information. The detailed scheme is describe as follows.

(1) Initialization: The signer randomly selects two large primes p and q , and calculates $n = p \cdot q$ and $\phi(n) = (p-1) \cdot (q-1)$. Then, the signer selects large integers d such that $e \cdot d \equiv 1 \pmod{\phi(n)}$, where $e = 3$. Thus, d is the private key of the signer and the signer publishes his public key (e, n) and a secure one-way hash function $h(\cdot)$ like SHA-1.

(2) Requesting: The requester prepares the common information a according to the predefined format. Then, s/he randomly selects two integers $r \in Z_n$ and $u \in Z_n$. The requester calculates $\alpha = r^e h(m)(u^2 + 1) \pmod{n}$ and sends (a, α) to the signer. After the signer verifying the agreed common information a , s/he randomly chooses a integer $x \in Z_n^+$, where $x < n$, and sends it to the requester. After the

requester receives x , s/he selects a random number k and computes $b = rk$. Finally, the requester computes $\beta = b^e(u - x) \bmod n$ and sends β to the signer.

(3) Signing: The signer calculates $\beta^{-1} \bmod n$ and $t = h(a)^d (\alpha(x^2 + 1)\beta^{-2})^{2d} \bmod n$ then s/he sends (β^{-1}, t) to the requester.

(4) Extraction and verification: After the requester receives (β^{-1}, t) , s/he obtains the signature by calculating $c = (ux + 1)\beta^{-1}b^e \bmod n$ and $s = tr^2k^4 \bmod n$. The 3-tuple (a, c, s) is a signature on the message m , and anyone can verify the correctness of (a, c, s) by checking whether $s^e = h(a)h(m)(c^2 + 1)^2 \bmod n$.



If (a, c, s) is a signature of the message m generated by Chien et al.'s partially blind signature scheme, then $s^e = h(a)h(m)(c^2 + 1)^2 \bmod n$ must be held. The detailed proof can be found in [11].

3.2.2 Hwang et al.'s Traceability Attack

In Hwang et al.'s claim [21], the signer can keep a set of record for all blinded messages and use them to trace back the blind signature. Thus, Hwang et al. claimed that Chien et al.'s scheme cannot meet the

untraceability of the blind signature. The detailed procedures of Hwang et al.'s cryptanalysis are described as follows.

1. The signer can keep a set of records $(\alpha_i, x_i, \beta_i, t_i, \beta_i^{-1})$ for each instance i in Chien et al.'s scheme.
2. When the requester reveals (a, c, s, m) to the public, the signer can compute $\tilde{u}_i = (1 + cx_i)(c - x_i)^{-1} \bmod n$ for each instance i since $c = (u_i x_i + 1)\beta_i^{-1} b_i^e = (u_i x_i + 1)(u_i - x_i)^{-1} \bmod n$.

3. The signer can obtain $\tilde{b}_i = \beta_i^d (\tilde{u}_i - x_i)^{-d} \bmod n$ for each instance i since $\beta = b^e (u - x) \bmod n$.

Note: $\tilde{b}_i = \beta_i^d (\tilde{u}_i - x_i)^e \bmod n$ is wrong in Hwang et al. [21].

4. The signer can then compute $\tilde{r}_i = \alpha_i^d h(m)^{-d} (\tilde{u}_i^2 + 1)^{-d} \bmod n$ for each instance i since $\alpha_i = r_i^e h(m)(u_i^2 + 1) \bmod n$.

Note: $\tilde{r}_i = \alpha_i^d h(m)^e (\tilde{u}_i^2 + 1)^d \bmod n$ is also wrong in Hwang et al. [21].

5. The signer can obtain $\tilde{k}_i = \tilde{b}_i \tilde{r}_i^{-1} \bmod n$ for each instance i since $b_i = r_i k_i \bmod n$.

6. Finally, the signer can check if $s = t_i \cdot \tilde{r}_i^2 \cdot \tilde{k}_i^4 \bmod n$. If it is true, the signer can trace back the blind signature.

Therefore, Hwang et al. claimed that Chien et al.'s scheme cannot meet

the untraceability property of the blind signature.

3.2.3 Analysis of Hwang et al.'s Attack

In 1995, Harn [19] claimed that Camenisch et. al.'s blind signature scheme [5] is traceable. Horster et al. [20] proved that Harn's cryptanalysis is incorrect later. However, **Theorem 3.2** shows that Hwang et al.'s claim on Chien et al.'s scheme is incorrect.

Theorem 3.2: For given a message-signature pair (a, c, s, m) , the signer can derive 4-tuple $(\tilde{u}_i, \tilde{b}_i, \tilde{r}_i, \tilde{k}_i)$ such that $s = t_i \cdot \tilde{r}_i^2 \cdot \tilde{k}_i^4 \pmod n$ is always satisfied for each $(\alpha_i, x_i, \beta_i, t_i, \beta_i^{-1})$.

Proof: According to Hwang et al.'s claim, the signer can keep $(\alpha_i, x_i, \beta_i, t_i, \beta_i^{-1})$ for each instance i in Chien et al.'s scheme. When the requester reveals (a, c, s, m) to the public, the signer can compute $\tilde{u}_i = (1 + cx_i)(c - x_i)^{-1} \pmod n$ for each instance i . Then s/he can obtain $\tilde{b}_i = \beta_i^d (\tilde{u}_i - x_i)^{-d} \pmod n$. The signer can compute $\tilde{r}_i = \alpha_i^d h(m)^{-d} (\tilde{u}_i^2 + 1)^{-d} \pmod n$ and $\tilde{k}_i = \tilde{b}_i \tilde{r}_i^{-1} \pmod n$. Finally, the signer can check whether the formula $s = t_i \cdot \tilde{r}_i^2 \cdot \tilde{k}_i^4 \pmod n$ is true or not. However, the formula $s = t_i \cdot \tilde{r}_i^2 \cdot \tilde{k}_i^4 \pmod n$ is always true for each instance

i in the following derivations.

$$\begin{aligned}
& (t_i \cdot \tilde{r}_i^2 \cdot \tilde{k}_i^4) \\
& \equiv h(a)^d \cdot (\alpha_i(x_i^2 + 1)\beta_i^{-2})^{2d} \cdot \tilde{r}_i^2 (\tilde{b}_i \cdot \tilde{r}_i^{-1})^4 \pmod{n} \\
& \equiv h(a)^d \cdot (\alpha_i(x_i^2 + 1)\beta_i^{-2})^{2d} \cdot \tilde{b}_i^4 \cdot \tilde{r}_i^{-2} \pmod{n} \\
& \equiv h(a)^d \cdot (\alpha_i(x_i^2 + 1)\beta_i^{-2})^{2d} \cdot (\beta_i^d (\tilde{u}_i - x_i)^{-d})^4 \cdot (\alpha_i^d h(m)^{-d} (\tilde{u}_i^2 + 1)^{-d})^{-2} \pmod{n} \\
& \equiv h(a)^d \cdot (\alpha_i^{2d} (x_i^2 + 1)^{2d} \beta_i^{-4d}) \cdot (\beta_i^{4d} (\tilde{u}_i - x_i)^{-4d}) \cdot (\alpha_i^{-2d} h(m)^{2d} (\tilde{u}_i^2 + 1)^{2d}) \pmod{n} \\
& \equiv h(a)^d \cdot (x_i^2 + 1)^{2d} \cdot (\tilde{u}_i - x_i)^{-4d} \cdot (h(m)^{2d} (\tilde{u}_i^2 + 1)^{2d}) \pmod{n} \\
& \equiv h(a)^d \cdot h(m)^{2d} \cdot [(x_i^2 + 1) \cdot (\tilde{u}_i - x_i)^{-2} \cdot (\tilde{u}_i^2 + 1)]^{2d} \pmod{n} \\
& \equiv [h(a) \cdot h(m)^2 \cdot (x_i^2 + 1) \cdot (\tilde{u}_i - x_i)^{-2} \cdot (\tilde{u}_i^2 + 1)]^d \pmod{n} \\
& \equiv [h(a) \cdot h(m)^2 \cdot (x_i^2 + 1) \cdot (\tilde{u}_i - x_i)^{-2} \cdot (\tilde{u}_i^2 + 1)]^d \pmod{n} \\
& \equiv [h(a) \cdot h(m)^2 \cdot [(\tilde{u}_i - x_i)^{-2} \cdot (x_i^2 \tilde{u}_i^2 + x_i^2 + \tilde{u}_i^2 + 1)]]^d \pmod{n} \\
& \equiv [h(a) \cdot h(m)^2 \cdot [(\tilde{u}_i - x_i)^{-2} \cdot (x_i^2 \tilde{u}_i^2 + x_i^2 + \tilde{u}_i^2 + 1 + 2\tilde{u}_i x_i - 2\tilde{u}_i x_i)]]^d \pmod{n} \\
& \equiv [h(a) \cdot h(m)^2 \cdot [(\tilde{u}_i - x_i)^{-2} \cdot ((x_i \tilde{u}_i + 1)^2 + (\tilde{u}_i - x_i)^2)]]^d \pmod{n} \\
& \equiv [h(a) \cdot h(m)^2 \cdot [c^2 + 1]]^d \pmod{n} \\
& \equiv s \pmod{n} \quad \square
\end{aligned}$$

Thus, Hwang et al.'s cryptanalysis on Chien et al.'s scheme is incorrect.

Chien et al.'s partially blind signature scheme is still obtain the

untraceability property and it is an untraceable scheme.

3.3 Untraceable ElGamal Blind Signature Scheme

In Eurocrypt'94, Camenisch et al. presented the blind signature schemes based on the discrete logarithm problem. Recently, Lee et al. asserted that Camenisch et al.'s schemes cannot satisfy the untraceability property of the blind signature scheme. We will analyze that Lee et al.'s traceability attack is failed and Camenisch et al.'s schemes are still untraceable here. Although Lee et al. presented an untraceable scheme, it needs more computations and storages than Camenisch et al.'s schemes. Hence, Lee et al.'s scheme is unnecessary.

A blind signature scheme is a protocol to allow the requester to obtain a signature without revealing message and the signer cannot trace any message-signature pair later. It can achieve the unforgeability property for the signer and the untraceability for the requester. The first blind signature scheme was presented by Chaum [8] and it is based on the integer factoring problem. Camenisch et al. [5] proposed DSA [34] and Nyberg-Rueppel [35] blind signature schemes based on the discrete

logarithm problem in Eurocrypt'94. Harn [19] pointed out that Camenisch et al.'s schemes are traceable in 1995. Horster et al. [20] showed that Harn's cryptanalysis is incorrect later. Recently, Lee et al. [27] claimed Horster et al.'s comment is improper and asserted Camenisch et al.'s schemes cannot satisfy the untraceability property of the blind signature scheme. However, we show that Lee et al.'s traceability attack on Camenisch et al.'s schemes is failed in this section.

3.3.1 Camenisch et al.'s scheme

There are two kinds of participant: the signer and the requester in Camenisch et al.'s blind signature scheme. Initialization, requesting, signing, and verification are four phases in their schemes and the details of DSA blind signature scheme are described in the following. (The concept of Nyberg-Rueppel blind signature scheme is similar to DSA blind signature scheme and its details are omitted here.)

(1) Initialization: Two large primes p and q are randomly chosen by the signer such that $q|(p-1)$. Next, s/he selects $g \in Z_p^*$ of order q and a random number $x \in Z_q$, and computes $y = g^x \pmod{p}$. Thus, the signer's secret key is x and the corresponding public key is y . Finally,

the signer randomly selects $\hat{k} \in Z_q$ and calculates $\hat{r} = g^{\hat{k}} \pmod{p}$, and sends \hat{r} to the requester.

(2) Requesting: To sign a message m which is relatively prime to q , the requester selects two random numbers $a, b \in Z_q$ and computes $r = \hat{r}^a g^b \pmod{p}$. Then, s/he calculates the blinded message $\hat{m} = am\hat{r}^{-1} \pmod{q}$ and sends \hat{m} to the signer.

(3) Signing: After the signer receives \hat{m} , s/he computes $\hat{s} = x\hat{r} + k\hat{m} \pmod{q}$ and sends \hat{s} back to the requester.

(4) Verification: The requester can calculate the signature s by the equation $s = \hat{s}\hat{r}^{-1} + bm \pmod{q}$. Thus, (r, s) is the signature on the message m . Anyone can verify the signature by checking whether $g^s = y^r r^m \pmod{p}$ holds.

3.3.2 Lee et al.'s Traceability Attack

Recently, Lee et al. [27] asserted that Camenisch et al.'s schemes [5] cannot satisfy the untraceability property of the blind signature scheme. The detailed procedures of Lee et al.'s traceability attack on Camenisch et al.'s DSA blind signature scheme are described as follows. (The

traceability attack on Nyberg-Rueppel blind signature scheme is similar to DSA blind signature scheme and its description is omitted for concise.)

1. The signer can record all instances $(\hat{k}_i, \hat{r}_i, \hat{m}_i, \hat{s}_i)$ in Camenisch et al.'s scheme.

2. After the requester publishes (r, s, m) , the signer can calculate

$$b_i = m^{-1}(s - \hat{s}_i r \hat{r}_i^{-1}) \pmod{q} \quad \text{for all instances because of}$$

$$s = \hat{s}_i r \hat{r}_i^{-1} + b m \pmod{q}.$$

3. Next, the signer can compute $a_i = \hat{m}_i m^{-1} \hat{r}_i^{-1} r \pmod{q}$ for all instances because of $\hat{m} = a m r r^{-1} \pmod{q}$.

4. Finally, the signer can check whether $r = \hat{r}_i^{a_i} g^{b_i} \pmod{p}$ holds. If it is true, the signer can trace the blind signature.

Thus, Lee et al. asserted that Camenisch et al.'s schemes cannot satisfy the untraceability property of the blind signature.

3.3.3 Analysis of Lee et al.'s Attack

Recently, Hwang et al. [22] asserted that Chaum's blind signature scheme [8] is traceable and presented an untraceable blind signature scheme based on integer factoring problem. Lee and Wu [29] showed that Hwang

et al.'s claim is invalid later. There are several papers [23, 24] claimed that many blind signature schemes incurred the traceability attack. However, many cryptanalysts [28, 14] have showed the traceability attack is failed later. We analyze that Lee et al.'s traceability attack is failed in the following.

Based on Lee et al.'s traceability attack, the signer can keep $(\hat{k}_i, \hat{r}_i, \hat{m}_i, \hat{s}_i)$ for all instances in Camenisch et al.'s schemes. After the requester publishes (r, s, m) , the signer can calculate $b_i = m^{-1}(s - \hat{s}_i r \hat{r}_i^{-1}) \pmod{q}$ and $a_i = \hat{m}_i m^{-1} \hat{r}_i^{-1} r \pmod{q}$ for all instances. Then, the signer can check whether $r = \hat{r}_i^{a_i} g^{b_i} \pmod{p}$ holds. If the result is true, Lee et al. asserted that the signer can trace the blind signature in Camenisch et al.'s schemes. Indeed, we analyze that $r = \hat{r}_i^{a_i} g^{b_i} \pmod{p}$ is always true for all instances in the following.

$$\begin{aligned}
& \hat{r}_i^{a_i} g^{b_i} \pmod{p} \\
&= g^{\hat{k}_i a_i} g^{m^{-1}(s - \hat{s}_i r \hat{r}_i^{-1})} \pmod{p} \\
&= g^{\hat{k}_i (\hat{m}_i m^{-1} \hat{r}_i^{-1} r)} g^{m^{-1}(s - \hat{s}_i r \hat{r}_i^{-1})} \pmod{p} \\
&= g^{\hat{k}_i (\hat{m}_i m^{-1} \hat{r}_i^{-1} r) + m^{-1}(s - \hat{s}_i r \hat{r}_i^{-1})} \pmod{p} \\
&= g^{m^{-1}(\hat{k}_i \hat{m}_i \hat{r}_i^{-1} r + s - \hat{s}_i r \hat{r}_i^{-1})} \pmod{p}
\end{aligned}$$

$$= g^{m^{-1}(\hat{k}_i \hat{m}_i \hat{r}_i^{-1} r + s - (x \hat{r}_i + \hat{k}_i \hat{m}_i) r \hat{r}_i^{-1})} \pmod{p}$$

$$= g^{m^{-1}(\hat{k}_i \hat{m}_i \hat{r}_i^{-1} r + s - (x r + \hat{k}_i \hat{m}_i r \hat{r}_i^{-1})} \pmod{p}$$

$$= g^{m^{-1}(\hat{k}_i \hat{m}_i \hat{r}_i^{-1} r + s - x r - \hat{k}_i \hat{m}_i r \hat{r}_i^{-1})} \pmod{p}$$

$$= g^{m^{-1}(s - x r)} \pmod{p}$$

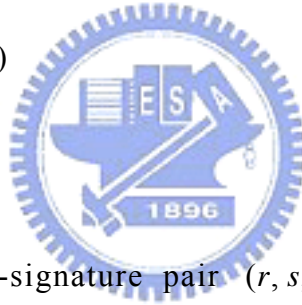
$$= g^{m^{-1} s} g^{-m^{-1} x r} \pmod{p}$$

$$= (y^r r^m)^{m^{-1}} g^{-m^{-1} x r} \pmod{p}$$

$$= (y^{r m^{-1}} r) g^{-m^{-1} x r} \pmod{p}$$

$$= (g^{x m^{-1}} r) g^{-m^{-1} x r} \pmod{p}$$

$$= r \pmod{p}$$



For a given message-signature pair (r, s, m) , the signer can derive (a_i, b_i) such that $r = \hat{r}_i^{a_i} g^{b_i} \pmod{p}$ is always held for all instances $(\hat{k}_i, \hat{r}_i, \hat{m}_i, \hat{s}_i)$. Hence, Lee et al.'s traceability attack on Camenisch et al.'s schemes is failed. Although Lee et al.'s scheme satisfies the untraceability property, it needs more computations and storages than Camenisch et al.'s schemes. Thus, Lee et al.'s scheme is unnecessary.

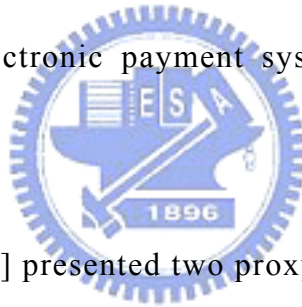
3.4 The Secure Proxy Blind Signature Schemes

The proxy blind signature scheme allows the designated proxy signer using the proxy secret key to generate a blind signature on behalf of the original signer. Tan et al. presented the DLP-based and ECDLP based blind signature schemes. Lal and Awasthi proposed a improved DLP-based scheme later. Recently, Sun et al. presented linkability attack on Tan et al.'s and Lal-Awasthi's proxy blind signature schemes respectively. In this section, we show that Sun et al.'s attack is invalid and these schemes are still satisfy the unlinkability property.

Mambo et al. [30] presented the proxy signature scheme to allow the designated proxy signer to sign messages on behalf of the original signer. For example, when a manager is going on a vocation, s/he can delegate her/his secretary to sign messages on behalf of her/him. The defined three types of delegation in the proxy signature scheme are full delegation, partial delegation and delegation by warrant. In full delegation, the original signer gives her/his private key to the designated proxy signer and then the original signer and the proxy signer can both generate the same signatures. The original signer uses her/his private key to generate the proxy secret key and sends it to the proxy signer securely in partial delegation. In delegation by warrant, the warrant is used to show that the

proxy signer is legal and to describe the information between the original signer and the proxy signer.

The blind signature scheme was first proposed by Chaum [8] in Crypto'83. The security of Chaum's scheme is based on the difficulty of integer factoring. The blind signature scheme can allow the receiver to obtain a signature signed by the signer without revealing message and the signer cannot link any message-signature pair later. It can achieve the unforgeability property for the signer and the unlinkability for the receiver. Hence, it is useful in electronic payment systems and electronic voting systems.



In 2002, Tan et al. [49] presented two proxy blind signature schemes to allow the proxy signer to generate a blind signature on behalf of the original signer. Lal and Awasthi [25] showed a forgery attack on Tan et al.'s schemes and proposed a more secure proxy blind signature scheme later. Recently, Sun et al. [47] pointed out that neither Tan et al.'s schemes nor Lal-Awasthi's scheme can satisfy the unlinkability property of the proxy blind signature scheme. In this section, we show that Sun et al.'s attack is invalid and these schemes are still satisfy the unlinkability property.

3.4.1 The Proxy Blind Signature Schemes

The system parameters in the following proxy blind signature schemes are defined as follows.

System Parameters:

p, q : two large prime numbers, where $q | (p-1)$.

g : element of Z_p^* of order q .

x_o, y_o : secret key and public key of the original signer respectively, where

$$y_o = g^{x_o} \text{ mod } p.$$

x_p, y_p : secret key and public key of the proxy signer respectively, where

$$y_p = g^{x_p} \text{ mod } p.$$

$h()$: a secure and public one way hash function.

$\|$: the concatenation of strings.

3.4.1.1 Tan et al.'s proxy blind signature schemes

Tan et al. [49] presented two proxy blind signature schemes based on the discrete logarithm problem (DLP) and elliptic curve discrete logarithm problem (ECDLP) in 2002. They also defined the required security

properties of proxy blind signature scheme. There are three kinds of participants: original signer, the proxy signer and the receiver in their schemes. The three phases in their schemes are (1) Proxy delegation, (2) Signing and (3) Verification. The details of Tan et al.'s DLP-based scheme are described as follows.

(1) Proxy delegation: The original signer randomly selects a number k_o , and calculates $r_o = g^{k_o} \bmod p$ and $s_o = k_o + x_o r_o \bmod q$. Then, the original signer sends (r_o, s_o) to the proxy signer in a secure way. After the proxy signer receives it, s/he can verify it by checking the correctness of the equation $g^{s_o} = y_o^{r_o} r_o \bmod p$. Finally, the proxy signer computes her/his proxy secret key $s_{pr} = s_o + x_p \bmod q$.

(2) Signing: The proxy signer chooses a random number k , computes $t = g^k \bmod p$ and sends (r_o, t) to the receiver. After receiving it, the receiver randomly chooses two numbers a and b and calculates $r = t g^b y_p^{-a-b} (y_o^{r_o} r_o)^{-a} \bmod p$, $e = h(r || m) \bmod q$, $u = (y_o^{r_o} r_o)^{-e+b} y_o^{-e} \bmod p$ and $e' = (e - a - b) \bmod q$. Then, the receiver sends e' to the proxy signer. Next, the proxy signer calculates the blinded signature $s' = e' s_{pr} + k \bmod q$ and sends s' back to the receiver. Finally, the receiver computes $s = s' + b \bmod q$. The signature of the message m is (m, u, s, e) .

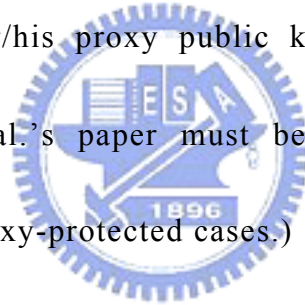
(3) Verification: Anyone can verify the correctness of the proxy blind signature (m, u, s, e) by checking that $e = h(g^s y_p^{-e} y_o^e u \text{ mod } p \parallel m) \text{ mod } q$ holds.

The descriptions of Tan et al.'s ECDLP-based proxy blind signature scheme is omitted here because it is similar to DLP-based scheme except to replace discrete logarithm cryptosystem parameters by elliptic curve cryptosystem parameters.

3.4.1.2 Lal and Awasthi's proxy blind signature scheme

Lal and Awasthi [25] showed a forgery attack on Tan et al.'s schemes and proposed a more secure and efficient proxy blind signature scheme later. Proxy-unprotected and proxy-protected are two kinds of schemes according to whether the original signer can generate the same proxy signature as the proxy signer. In proxy-protected schemes, the proxy signer and the original signer both can generate valid proxy signatures. Only the proxy signer can generate valid proxy signatures that s/he cannot repudiate it later in proxy-protected schemes. The participants, phases and system parameters are the same as Tan et al.'s schemes. The detailed scheme is described in the following.

(1) Proxy delegation: The original signer chooses a random number k_o , and computes $r_o = g^{k_o} \bmod p$ and $s_o = x_o + k_o r_o \bmod q$. Next, the original signer sends (r_o, s_o) to the proxy signer via a secure channel. After the proxy signer receives it, s/he can verify it by checking whether the equation $g^{s_o} = y_o r_o^{r_o} \bmod p$ holds. In proxy-unprotected case, the proxy signer uses $s_{pr} = s_o$ as her/his proxy secret key and $y_{pr} = y_o r_o^{r_o} \bmod p$ as her/his proxy public key. In proxy-protected case, the proxy signer computes $s_{pr} = s_o + x_{pr} \bmod q$ as her/his proxy secret key and $y_{pr} = y_o r_o^{r_o} y_p \bmod p$ as her/his proxy public key. (Note that the proxy public keys in Sun et al.'s paper must be exchanged each other in proxy-unprotected and proxy-protected cases.)

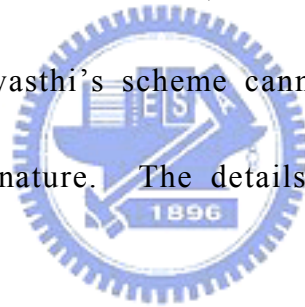


(2) Signing: The proxy signer randomly chooses a number k and computes $t = g^k \bmod p$ and sends (r_o, t) to the receiver. After receiving it, the receiver selects two random numbers a and b . Then s/he calculates $r = t g^{-a} y_{pr}^{-b} \bmod p$, $e' = h(r || m) \bmod q$, and $e = (e' + b) \bmod q$. The receiver sends e to the proxy signer. Next, the proxy signer calculates the blinded signature $s' = k - e s_{pr} \bmod q$ and sends s' back to the receiver. Finally, the receiver computes $s = s' - a \bmod q$ from the blind signature s' . The signature of the message m is (m, s, e') .

(3) Verification: Anyone can verify the correctness of the proxy blind signature (m, s, e') by checking whether $e' = h(g^s y_{pr}^{e'} \bmod p \parallel m) \bmod q$ holds.

3.4.2 Sun et al.'s Traceability Attack

In Sun et al.'s [47] linkability attack, they pointed out that the proxy signer can record all blinded messages and use them to trace back the corresponding blind signatures. Hence, Sun et al. claimed that all Tan et al.'s schemes and Lal-Awasthi's scheme cannot satisfy the unlinkability property of the blind signature. The details of Sun et al.'s attack are described as follows.



3.4.2.1 Sun et al.'s attack on Tan et al.'s schemes

We only describe the detailed Sun et al.'s attack on Tan et al.'s DLP-based proxy blind signature scheme because Tan et al.'s ECDLP-based scheme is similar to it.

1. The proxy signer can keep all set of records (t_i, e_i', s_i') for each instance i in Tan et al.'s DLP-based scheme, where $t_i = g^{k_i} \bmod p$.
2. When the receiver reveals (m, u, s, e) to the public, the proxy signer can

compute $b_i' = s - s_i' \bmod q$ for each instance i since $s = s' + b \bmod q$.

3. The proxy signer can calculate $a_i' = (e - b_i' - e_i') \bmod q$ for each instance i since $e' = (e - a - b) \bmod q$.
4. Then the proxy signer can compute $r_i' = t_i g^{b_i'} y_p^{-a_i' - b_i'} (y_o^{r_o} r_o)^{-a_i'} \bmod p$ for each instance i since $r = t g^b y_p^{-a-b} (y_o^{r_o} r_o)^{-a} \bmod p$.
5. Finally, the proxy signer can check that $r_i' = g^s y_p^{-e} y_o^e u \bmod p$ holds. If it is true, the proxy signer can trace back the blind signature.

Hence, Sun et al. claimed that Tan et al.'s schemes cannot satisfy the unlinkability property of the blind signature.



3.4.2.2 Sun et al.'s attack on Lal-Awasthi's scheme

1. The proxy signer can keep all set of records (t_i, e_i', \tilde{s}_i) for each instance i , where $t_i = g^{k_i} \bmod p$.
2. After the receiver reveals (m, s, e) to the public, the proxy signer can calculate $a_i' = \tilde{s}_i - s \bmod q$ for each instance i since $s = \tilde{s} - a \bmod q$.
3. The proxy signer can calculate $b_i' = (e_i' - e) \bmod q$ for each instance i since $e' = (e + b) \bmod q$.
4. The proxy signer then can compute $r_i' = t_i g^{-a_i'} y_{pr}^{-b_i'} \bmod p$ for each instance i since $r = t g^{-a} y_{pr}^{-b} \bmod p$.

5. Finally, the proxy signer can check whether $r_i' = g^s y_{pr}^e \text{ mod } p$ holds. If the equation is true, the proxy signer can trace back the blind signature.

Thus, Sun et al. claimed that Lal-Awasthi's scheme cannot satisfy the unlinkability property of the blind signature.

3.4.3 Analysis of Sun et al.'s Attack

In this section, we show that Sun et al.'s linkability attack is failed and Tan et al.'s [49] and Lal-Awasthi's [25] proxy blind signature schemes are still unlinkable.



3.4.3.1 Analysis of Sun et al.'s attack on Tan et al.'s schemes

According to Sun et al.'s linkability attack, the proxy signer can keep all set of records (t_i, e_i', s_i') for each instance i in Tan et al.'s DLP-based scheme. After the receiver reveals (m, u, s, e) to the public, the proxy signer can calculate $b_i' = s - s_i' \text{ mod } q$ for each instance i . Next, s/he can obtain $a_i' = (e - b_i' - e_i') \text{ mod } q$. Then the proxy signer can calculate $r_i' = t_i g^{b_i'} y_p^{-a_i' - b_i'} (y_o^{r_o} r_o)^{-a_i'} \text{ mod } p$. Finally, the proxy signer can check whether the equation $r_i' = g^s y_p^{-e} y_o^e u \text{ mod } p$ holds. However, we show that the equation is always true for each instance i in the following.

$$\begin{aligned}
& t_i g^{b_i'} y_p^{-a_i'-b_i'} (y_o^{r_o} r_o)^{-a_i'} \pmod p \\
& \equiv t_i g^{s-s_i'} y_p^{-e+b_i'+e_i'+s_i'-s} (y_o^{r_o} r_o)^{b_i'+e_i'-e} \pmod p \\
& \equiv g^s (t_i g^{-s_i'}) y_p^{-e} (y_p^{b_i'+e_i'+s_i'-s}) (y_o^{r_o} r_o)^{b_i'+e_i'-e} \pmod p \\
& \equiv g^s (t_i g^{-s_i'}) y_p^{-e} (y_p^{s-s_i'+e_i'+s_i'-s}) (y_o^{r_o} r_o)^{b_i'+e_i'-e} \pmod p \\
& \equiv g^s (t_i g^{-s_i'}) y_p^{-e} (y_p^{e_i'}) (y_o^{r_o} r_o)^{b_i'+e_i'-e} \pmod p \\
& \equiv g^s (t_i g^{-s_i'}) y_p^{-e} (y_p^{e_i'}) (y_o^{r_o} r_o)^{b_i'-e} (y_o^{r_o} r_o)^{e_i'} \pmod p \\
& \equiv g^s (t_i g^{-s_i'}) y_p^{-e} (y_p^{e_i'}) (y_o^{r_o} r_o)^{b_i'-e} (y_o^{r_o} r_o)^{e_i'} (y_o^e y_o^{-e}) \pmod p \\
& \equiv (g^s y_p^{-e} y_o^e) (t_i g^{-s_i'} y_p^{e_i'}) (y_o^{r_o} r_o)^{b_i'-e} (y_o^{r_o} r_o)^{e_i'} (y_o^{-e}) \pmod p \\
& \equiv (g^s y_p^{-e} y_o^e) (g^{k_i} g^{-e_i' s_{pr}-k_i} y_p^{e_i'}) (y_o^{r_o} r_o)^{b_i'-e} (y_o^{r_o} r_o)^{e_i'} (y_o^{-e}) \pmod p \\
& \equiv (g^s y_p^{-e} y_o^e) (g^{k_i-k_i} g^{-e_i'(s_o+x_p)} y_p^{e_i'}) (y_o^{r_o} r_o)^{b_i'-e} (y_o^{r_o} r_o)^{e_i'} (y_o^{-e}) \pmod p \\
& \equiv (g^s y_p^{-e} y_o^e) (g^{-e_i'(s_o+x_p)} g^{e_i' x_p}) (y_o^{r_o} r_o)^{b_i'-e} (y_o^{r_o} r_o)^{e_i'} (y_o^{-e}) \pmod p \\
& \equiv (g^s y_p^{-e} y_o^e) (g^{-e_i' s_o}) (y_o^{r_o} r_o)^{b_i'-e} (y_o^{r_o} r_o)^{e_i'} (y_o^{-e}) \pmod p \\
& \equiv (g^s y_p^{-e} y_o^e) (y_o^{r_o} r_o)^{-e_i'} (y_o^{r_o} r_o)^{e_i'} (y_o^{r_o} r_o)^{b_i'-e} (y_o^{-e}) \pmod p \\
& \equiv (g^s y_p^{-e} y_o^e) (y_o^{r_o} r_o)^{b_i'-e} (y_o^{-e}) \pmod p \\
& \equiv g^s y_p^{-e} y_o^e u \pmod p \\
& \equiv r_i' \pmod p
\end{aligned}$$

For a given message-signature pair (a, c, s, m) , the proxy signer can derive 3-tuple (b'_i, a'_i, r'_i) such that $r'_i = g^s y_p^{-e} y_o^e u \bmod p$ is always held for each (t_i, e'_i, s'_i) . Hence, Sun et al.'s claim is incorrect and Tan et al.'s DLP-based scheme is still satisfy the unlinkability property. The cryptanalysis of Sun et al.'s linkability attack on Tan et al.'s ECDLP-based scheme is similar to above description.

3.4.3.2 Analysis of Sun et al.'s attack on Lal-Awasthi's scheme

Based on Sun et al.'s linkability attack, the proxy signer can records all set of (t_i, e_i, s_i') for each instance i in Lal-Awasthi's scheme. After the receiver reveals (m, s, e') to the public, the proxy signer can compute $a'_i = (s'_i - s) \bmod q$ for each instance i . Then s/he can calculate $b'_i = (e_i - e') \bmod q$. Next, the proxy signer can compute $r'_i = t_i g^{-a'_i} y_{pr}^{-b'_i} \bmod p$. Finally, the proxy signer can check if the equation $e' = h(g^s y_{pr}^{-e'} \bmod p \parallel m) \bmod q$ holds. We show that the equation is always true for each instance i in the following.

$$\begin{aligned}
& h(t_i g^{-a'_i} y_{pr}^{-b'_i} \bmod p \parallel m) \bmod q \\
& \equiv h(t_i g^{s-s'_i} y_{pr}^{e'-e_i} \bmod p \parallel m) \bmod q \\
& \equiv h(g^s t_i g^{-s'_i} y_{pr}^{e'-e_i} \bmod p \parallel m) \bmod q
\end{aligned}$$

$$\begin{aligned}
&\equiv h(g^s t_i g^{e_i s_{pr} - k_i} y_{pr}^{e' - e_i} \bmod p \parallel m) \bmod q \\
&\equiv h(g^s g^{k_i - k_i} g^{e_i s_{pr}} y_{pr}^{e' - e_i} \bmod p \parallel m) \bmod q \\
&\equiv h(g^s g^{e_i s_{pr}} y_{pr}^{e' - e_i} \bmod p \parallel m) \bmod q \\
&\equiv h(g^s y_{pr}^{e_i} y_{pr}^{e' - e_i} \bmod p \parallel m) \bmod q \\
&\equiv h(g^s y_{pr}^{e'} \bmod p \parallel m) \bmod q \\
&\equiv e'
\end{aligned}$$

For a given message-signature pair (m, s, e') , the proxy signer can derive 3-tuple (b'_i, a'_i, r'_i) such that $e' = h(g^s y_{pr}^{-e'} \bmod p \parallel m) \bmod q$ is always held for each (t_i, e_i, s'_i) . Hence, Sun et al.'s linkability attack is failed again on Lal-Awasthi's scheme. Lal-Awasthi's scheme is still satisfy the unlinkability property of the proxy blind signature scheme.

Chapter 4 The Proposed Digital Blind Signature Schemes

To establish the basis of electronic transaction services, we will present several secure digital blind signature schemes in the following sections. The fail-stop blind signature scheme that can obtain unforgeability property for the signer and anonymity property for the participants is described in **Section 4.1**. Then, an improved blind signature scheme based on the elliptic curve cryptosystem is presented in **Section 4.2**. It can reduce 50% storage requirements for each signature and speed up performance more than 36% compared with Yeh-Chang's scheme.



4.1 A Fail-stop Blind Signature Scheme

The fail-stop signature scheme was proposed by Pfitzmann and Waidner [39, 38, 50]. It can protect the signer against a forger with more powerful computational capability to forge a signature. It is unconditionally secure for the signer and cryptographically secure for the requester. One important application of the fail-stop signature is electronic payment system [39]. The anonymity of participants is very important in electronic payment systems. However, it cannot be achieved

in the fail-stop signature scheme.

Chaum [7, 8] introduced the concept of a blind signature scheme which can protect the anonymity of participants. The blind signature scheme allows a user to obtain a message signed by the signer without revealing message and the signer cannot link any message-signature pair later. The blind signature scheme can be used in electronic payment systems to preserve participants' anonymity.

Thus, a fail-stop blind signature scheme is proposed to solve this problem. The presented fail-stop blind signature scheme is based on Susilo-Safavi-Pieprzyk [48] scheme (mentioned in **Section 2.5**). Our scheme can provide “proof of forgery” for signers and guarantee “anonymity” for the participants/requesters. We will give proof to show that the proposed scheme satisfies the conditions of fail-stop signature and blind signature.

4.1.1 The Proposed Blind Signature Scheme

The fail-stop blind signature scheme combines the advantages of both fail-stop signature and blind signature. Our proposed scheme is a

modification of Susilo et al.'s scheme with trusted dealer. There are seven phases (1) Initialization, (2) Key generation, (3) Blinding, (4) Signing, (5) Unblinding, (6) Verification and (7) Proof of forgery in the fail-stop blind signature scheme. The three kinds of participants in our scheme are the same as the section 2. The detailed scheme is described bellow.

(1) Initialization: Initially, the trusted dealer D chooses two large primes p and q such that $p=2p'+1$ and $q=2q'+1$, where p' and q' are also prime. D computes $n=pq$ and $\phi(n)=(p-1)(q-1)$. Next, e_D and d_D are chosen by the trusted dealer D such that $e_D d_D \equiv 1 \pmod{\phi(n)}$. Then, D chooses a integer $\alpha \in Z_n^*$ randomly and computes $\beta = \alpha^{d_D} \pmod{n}$. Finally, D publishes her/his public key (α, n) , keeps her/his private key d_D secretly and sends (e_D, β) to the signer S via a secure channel.

(2) Key generation: The signer S randomly chooses his(her) private key (k_1, k_2, k_3, k_4) , where $k_i \in Z_n^*$ and computes $\beta_1 = \alpha^{k_4} \beta^{k_3} \pmod{n}$, $\alpha_1 = \alpha^{k_3} \beta_1^{k_1} \pmod{n}$ and $\alpha_2 = \alpha^{k_4} \beta_1^{k_2} \pmod{n}$. Finally, S publishes her/his public key $(\beta_1, \alpha_1, \alpha_2)$ and a one-way hash function H .

(3) Blinding: For a message m , the receiver R selects a random

numbers r in Z_n^* . R computes $\tilde{m} = rH(m) \bmod n$ with a blinding factor r , where $H(m)$ is the hashed value of message m . Then, R sends the blinded message \tilde{m} and $x = H(r) \bmod n$ to S .

(4) Signing: In this phase, the signer S computes $\tilde{s}_1 = \tilde{m}(k_1x + k_2)$ and $\tilde{s}_2 = \tilde{m}(k_3x + k_4)$. S sends the blinded signature $(\tilde{s}_1, \tilde{s}_2)$ on blinded message \tilde{m} to R .

(5) Unblinding: After the receiver R obtains the blinded signature $(\tilde{s}_1, \tilde{s}_2)$, he/she performs the unblinding operation by computing $s_1 = r^{-1}\tilde{s}_1$ and $s_2 = r^{-1}\tilde{s}_2$. Then, (s_1, s_2) is the signature on hashed message $H(m)$.

(6) Verification: Anyone can verify the message-signature $(H(m), x, s_1, s_2)$ by checking if $\alpha^{s_2} \beta_1^{s_1} = \alpha_1^{H(m)} \alpha_2 \bmod n$.

(7) Proof of forgery: This phase is similar to Susilo et al.'s scheme. The signer can prove that a forgery has occurred by revealing the non-trivial factors of n .

4.1.2 Security Analysis

A secure fail-stop blind signature scheme must satisfy four conditions as follows.

- (1) The forger is nearly infeasible to forge a signature with more powerful computational capability.
- (2) The signer can use a polynomial-time algorithm to prove that a forgery has occurred.
- (3) The polynomial-bounded signer cannot forge a signature and prove it a forgery later.
- (4) The signer is computationally infeasible to link the message he actually signed and the corresponding signature for verification later.



Theorem 4.1: *There exists the matching private keys for each public key, such that different private key can generate different signature on the same message.*

Theorem 4.2: *The signer can prove that a forgery has occurred by factorizing n if a forged signature (s_1', s_2') on a message m succeeds in verification phase.*

Theorem 4.3: *The signer can prove that a forgery has occurred by the*

$$\text{probability} \quad \frac{\phi(n)-1}{\phi(n)}.$$

The second condition of a secure fail-stop blind signature is satisfied by **Theorem 4.2**. The following theorem shows that a forger with more powerful computational capability is still existing $\phi(n)$ possible private keys for that signature.

Theorem 4.4: *The forger with more powerful computational capability is still existing $\phi(n)$ possible private keys for that blinded signature $(\tilde{s}_1, \tilde{s}_2)$ on the blinded message \tilde{m} together with corresponding public key.*



Proof: To Assume the forged blinded signature on the blinded message \tilde{m} is $(\tilde{s}_1', \tilde{s}_2')$ and the public key of the signer is $(\beta_1, \alpha_1, \alpha_2)$. If a forger with more powerful computational capability can solve the discrete logarithm and factorization problem successfully, he can obtain these equations as follows.

$$\tilde{s}_1' = (k_1 x + k_2) \tilde{m} \bmod \phi(n)$$

$$\tilde{s}_2' = (k_3 x + k_4) \tilde{m} \bmod \phi(n)$$

$$c_1 = (k_3 + w k_1) \bmod \phi(n)$$

$$c_2 = (k_4 + wk_2) \bmod \phi(n)$$

Where $\tilde{m} = rH(m)$, $x, c_1, c_2 \in Z_n^*$ and $w = \log_\alpha \beta_1 = k_4 + d_D k_3$. Then, a forger can rewrite these equations by using matrix representation.

$$\begin{bmatrix} x\tilde{m} & \tilde{m} & 0 & 0 \\ 0 & 0 & x\tilde{m} & \tilde{m} \\ w & 0 & 1 & 0 \\ 0 & w & 0 & 1 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{bmatrix} = \begin{bmatrix} \tilde{s}_1' \\ \tilde{s}_2' \\ c_1 \\ c_2 \end{bmatrix}$$

The above matrix's rank is 3 because $x\tilde{m}r_3 - wr_1 - r_2 + \tilde{m}r_4 = 0$, where r_i is the i -th row of the matrix. There are $\phi(n)$ possible private keys for that blinded signature since the solutions of equations are $\phi(n)$. \square



Theorem 4.5: *The forger with more powerful computational capability cannot generate the blinded signature on a new message.*

Theorem 4.6: *The polynomial-bounded signer cannot generate a valid signature and prove it a forgery later.*

Proof: The polynomial-bounded signer must have another private key (k_1', k_2', k_3', k_4') which can match the corresponding public key $(\beta_1, \alpha_1, \alpha_2)$ to deny a generated valid signature, such that $\alpha_1 = \alpha^{k_3'} \beta_1^{k_1'} \bmod n$ and

$\alpha_2 = \alpha^{k_4'} \beta_1^{k_2'} \text{ mod } n$. The difficulty to find another private key (k_1', k_2', k_3', k_4') is equivalent to solve the discrete logarithm problem. Moreover, it is difficult to find d_D without knowing $\phi(n)$ since the difficulty of integer factorization. \square

Theorem 4.7: *There exists a correct private key selected by the signer corresponding to the public key, such that $(\tilde{s}_1, \tilde{s}_2)$ is the blind signature on the blinded message \tilde{m} and $(\tilde{s}_1', \tilde{s}_2')$ is also the blind signature on the blinded message \tilde{m}' , where $\tilde{m} \neq \tilde{m}'$.*

Proof: The signer can organize these equations as follows.

$$\tilde{s}_1 = (k_1 x + k_2) \tilde{m} \text{ mod } \phi(n)$$

$$\tilde{s}_2 = (k_3 x + k_4) \tilde{m} \text{ mod } \phi(n)$$

$$\tilde{s}_1' = (k_1 x + k_2) \tilde{m}' \text{ mod } \phi(n)$$

$$\tilde{s}_2' = (k_3 x + k_4) \tilde{m}' \text{ mod } \phi(n)$$

$$c_1 = (k_3 + w k_1) \text{ mod } \phi(n)$$

$$c_2 = (k_4 + w k_2) \text{ mod } \phi(n)$$

Where $\tilde{m} = rH(m)$, $x, c_1, c_2 \in Z_n^*$ and $w = \log_{\alpha} \beta_1 = k_4 + d_D k_3$. The matrix representation of above equations can rewrite as follows.

$$\begin{bmatrix} x\tilde{m} & \tilde{m} & 0 & 0 \\ 0 & 0 & x\tilde{m} & \tilde{m} \\ x\tilde{m}' & \tilde{m}' & 0 & 0 \\ 0 & 0 & x\tilde{m}' & \tilde{m}' \\ w & 0 & 1 & 0 \\ 0 & w & 0 & 1 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ k_3 \\ k_4 \end{bmatrix} = \begin{bmatrix} \tilde{s}_1 \\ \tilde{s}_2 \\ \tilde{s}_1' \\ \tilde{s}_2' \\ c_1 \\ c_2 \end{bmatrix}$$

Since $\tilde{m} \neq \tilde{m}'$, The above coefficient matrix's rank is 4. Hence, the private key is the correct private key selected by the signer corresponding to the public key. \square

Theorem 4.8: *The signer computationally cannot link the blinded message \tilde{m} he actually signed and the corresponding signature (s_1, s_2) for verification later.*



Proof: In the signing phase, the signer can obtain the blinded message $\tilde{m} = rH(m)$ and $x = H(r) \bmod n$. The signer can obtain the signature (s_1, s_2) in the verification phase, where

$$s_1 = r^{-1}\tilde{s}_1 = (k_1x + k_2)H(m)$$

$$s_2 = r^{-1}\tilde{s}_2 = (k_3x + k_4)H(m)$$

The signer is computationally infeasible to link the blinded message and the signature for verification later since a blinding factor is chosen randomly by the receiver. \square

Hence, the fail-stop blind signature scheme can obtain the unforgeability property for the signer and the anonymity property for the requesters. It also can be applied in more critical system like electronic payment systems which need higher security against more powerful forger and can preserve participants' anonymity. Recently, Chang et al. [6] presented a fail-stop blind signature scheme based on pairings. Their scheme can work in any Gap Diffie-Hellman group.



4.2 The Enhanced Generic Blind Signature Scheme

By modifying the generic blind signature scheme presented in [17], we propose an enhanced scheme in the followings. Let M be the underlying set of messages and R be a finite set of random strings. The proposed blind signature scheme consists of five elements (B, H, S, U, V) , where

(1) $B: M \times R \rightarrow M$ is a blinding function. Without r , it is infeasible for the signer to compute m from $B(m, r)$. The integer r is called the blinding factor of the message m , and r is randomly chosen from R and kept secret by some user. Besides, $B(m, r)$ is called the

blinded message.

(2) $H : M \rightarrow M$ is a public one-way hash function.

(3) $S : M \rightarrow MK$ is a signing function. S is kept secret by the signer and K is a positive integer, where $MK = M$ when $K = 1$ and $MK = MK^{-1} \times M$ when $K \geq 2$. Without S , it is computationally infeasible to compute $S(H(m))$, where $S(H(m))$ is called the signer's signature on the message m in the scheme.

(4) $U : MK \times R \rightarrow MK$ is an unblinding function. For every $m \in M$ and $r \in R$, $U(S(B(m, r)), r) = S(m)$, and it is computationally infeasible to derive $S(m)$ from $S(B(m, r))$ through U without r .

(5) $V : MK \times M \rightarrow \{True, False\}$ is a public verification formula. $V(t, m) = True$, where $t \in MK$ if and only if t is the signer's signature on m , i.e., $t = S(H(m))$.

The corresponding protocol is described in detail below.

(1) **Blinding:** A user randomly selects a blinding factor $r \in R$ and chooses a message $m \in M$, where some message may be hidden in m . Then s/he computes the blinded message $u = B(H^2(m), r)$ and submits it to the signer to request the signer's signature on $H^2(m)$, where

$$H^2(m) = H(H(m)).$$

(2) **Signing:** The signer applies S to u , and then sends $S(u)$ to the user.

(3) **Unblinding:** After receiving the signing result $S(u)$, the user computes $U(S(u), r)$ to obtain $S(H^2(m))$.

The user shows the signature-message pair $(S(H^2(m)), H(m))$ for verification and that 2-tuple can be verified by examining whether $V(S(H^2(m)), H(m)) = True$ or not. Later, the user can reveal m for further verification. Besides, given the pair $(S(H^2(m)), m)$, the signer cannot link $(S(H^2(m)), m)$ to the pair $(S(B(H^2(m), r)), B(H^2(m), r))$ since it is computationally infeasible for the signer to derive $H^2(m)$ from $B(H^2(m), r)$ or to convert $S(B(H^2(m), r))$ into $S(H^2(m))$ without r .

4.3 The Enhanced Blind Signature Scheme Based on the Elliptic Curve Cryptosystem

The elliptic curve cryptosystem has more advantages than RSA or DSA such as smaller key length and low bandwidth on equivalent security strength. Recently, Yeh and Chang presented the first blind signature

scheme based on the elliptic curve cryptosystem and it is a modification of Okamoto's signature. In this section, we propose a Schnorr-type blind signature scheme based on the elliptic curve that can reduce 50% storage requirements for each signature and speed up more than 36% performance compared with Yeh-Chang's scheme. We also show that our scheme is a secure blind signature scheme here.

4.3.1 Yeh-Chang's Blind Signature Scheme

Recently, Yeh and Chang [51] presented a blind signature scheme based on the elliptic curve cryptosystem and it is the modification of Okamoto's signature [36]. Yeh-Chang's scheme is a secure blind signature scheme and it can preserve the properties of unforgeability and untraceability. Yeh-Chang's scheme can reduce the storage requirements by 33% and speed up performance ratio more than 6 compared with Okamoto's blind signature.

Let p be a prime and let E be an elliptic curve over Z_p . It satisfies the equations $y^2 = x^3 + ax + b \pmod{p}$ and $(4a^3 + 27b^2 \neq 0) \pmod{p}$ together with a special point at infinity denoted by O , where $a, b \in Z_p$ [31]. The three phases in Yeh-Chang's scheme are

initialization, signature generation and signature verification. The detailed scheme is described as follows.

(1) Initialization: The signer chooses two points $G_1, G_2 \in E$ with prime order q and selects two random numbers $x_1, x_2 \in Z_q$ as private keys. Then s/he computes $Y_1 = -x_1 G_1 \pmod{q}$ and $Y_2 = -x_2 G_2 \pmod{q}$, where $Y_1, Y_2 \in E$.

(2) Signature generation: The signer selects two random numbers $r_1, r_2 \in Z_q$ and calculates $R = r_1 G_1 + r_2 G_2 \pmod{q}$. Then the signer sends Y_1, Y_2 and R to the requester. After the requester receiving R , the requester randomly chooses two numbers $a, b \in Z_q$ and computes $\tilde{R} = R + a(G_1 + G_2) + bY \pmod{q}$, where $Y = Y_1 + Y_2 \pmod{q}$. The requester calculates $t = h(m \parallel \tilde{r}_x)$, where m is the message, $\tilde{R} = (\tilde{r}_x, \tilde{r}_y)$ and $h()$ is one-way hash function. Then the requester computes $\tilde{m} = t + b \pmod{q}$ and sends \tilde{m} to the signer. The signer calculates $\tilde{s}_1 = r_1 + \tilde{m}x_1 \pmod{q}$ and $\tilde{s}_2 = r_2 + \tilde{m}x_2 \pmod{q}$ and sends \tilde{s}_1 and \tilde{s}_2 to the requester. After the requester receiving \tilde{s}_1 and \tilde{s}_2 , s/he calculates $s_1 = \tilde{s}_1 + a \pmod{q}$ and $s_2 = \tilde{s}_2 + a \pmod{q}$. The signature of the message m is (\tilde{R}, s_1, s_2) .

(3) Signature verification: Anyone can verify the correctness of the signature (\tilde{R}, s_1, s_2) by checking $t = h(m \parallel \tilde{r}_x)$, where

$$V = (\tilde{r}_x, \tilde{r}_y) = s_1G_1 + s_2G_2 + tY \pmod{q}.$$

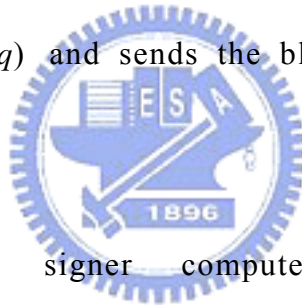
4.3.2 The Enhanced Signature Scheme

To improve the efficiency of Yeh-Chang's scheme, we propose a fast blind signature scheme modified from Schnorr's signature [45] based on the ECDLP. The elliptic curve cryptosystem assumptions are briefly described as follows. Let E be an elliptic curve over Z_p and the set of points (x, y) satisfy the equation $y^2 = x^3 + ax + b$, where $x, y, a, b \in Z_p$ and $4a^3 + 27b^2 \neq 0 \pmod{p}$. O is a special point on E at infinity [31]. The proposed scheme consists of four phases: (1) initialization, (2) blinding, (3) signing and (4) unblinding and verification. The signer will publish system parameters in the initialization phase. The requester sends a blinded message to the signer in the blinding phase. In the signing phase, the signer generates the blind signature and sends it back to the requester. The requester obtains the signature derived from the blind signature and anyone can verify the correctness of the signature in the unblinding and verification phase. The details of proposed scheme are described as follows.

(1) Initialization: The signer selects a point $G \in E$ with prime order

q and chooses a random number $x \in Z_q$ as private key. Then the signer computes $Y = -xG \pmod{q}$. The signer randomly chooses a number $r \in Z_q$ and computes $R = rG \pmod{q}$. Then the signer sends Y and R to the requester.

(2) Blinding: After receiving R , the requester selects two random numbers $a, b \in Z_q$ and calculates $\tilde{R} = R + aG - bY \pmod{q}$. Then the requester computes $t = h(m \parallel \tilde{r}_x) \pmod{q}$, where $h()$ is one-way hash function, $\tilde{R} = (\tilde{r}_x, \tilde{r}_y)$ and m is the message. Finally, the requester calculates $\tilde{m} = (t + b) \pmod{q}$ and sends the blinded message \tilde{m} back to the signer.



(3) Signing: The signer computes the blind signature $\tilde{s} = r + \tilde{m}x \pmod{q}$ and sends \tilde{s} back to the requester.

(4) Unblinding and verification: The requester computes $s = \tilde{s} + a \pmod{q}$ and the signature of the message m is (\tilde{R}, s) . Anyone can examine the correctness of the signature by checking $t = h(m \parallel r'_x) \pmod{q}$, where $(r'_x, r'_y) = sG + tY \pmod{q}$.

4.3.3 Security Analysis

In this section, we discuss the correctness, randomness, unforgeability and unlinkability of the proposed scheme.

The following theorem can ensure the correctness of the proposed scheme.

Theorem 4.5: If (\tilde{R}, s) is a valid signature of the message m , then $t = h(m \parallel r'_x) \pmod{q}$, where $(r'_x, r'_y) = sG + tY \pmod{q}$

Proof : $(r'_x, r'_y) = sG + tY \pmod{q}$

$$= (\tilde{s} + a)G + tY \pmod{q}$$

$$= (r + \tilde{m}x + a)G + tY \pmod{q}$$

$$= (r + (t + b)x + a)G + tY \pmod{q}$$

$$= (r + bx + a)G + txG + tY \pmod{q}$$

$$= (rG + bxG + aG) \pmod{q}$$

$$= (R - bY + aG) \pmod{q}$$

$$= \tilde{R} \pmod{q}$$

$$= (\tilde{r}_x, \tilde{r}_y) \pmod{q}$$

The requester can examine the correctness of the signature by checking

$t = h(m \parallel r'_x) \pmod{q}$ since $r'_x = \tilde{r}_x \pmod{q}$. If $t = h(m \parallel r'_x) \pmod{q}$ is hold, the signature (\tilde{R}, s) on the message m is a valid one. \square

In the proposed scheme, the signer randomized the blinded message using the random number r . The attacker is computationally infeasible to remove r from $\tilde{R} = rG + aG - bY \pmod{q}$, since s/he has to solve the elliptic curve problem and it is hard to be solved. The signature (\tilde{R}, s) of the proposed scheme has the randomness property.

In the blinding phase, the requester randomly selects $a, b \in Z_q$ and computes the blinded message \tilde{m} . Since $\tilde{m} = (h(m \parallel \tilde{r}_x) + b) \pmod{q}$, where $(\tilde{r}_x, \tilde{r}_y) = rG + aG - bxG \pmod{q}$, the signer cannot know the message m . Hence, the blindness property can be obtained in the proposed scheme.

The security of the proposed scheme is based on the difficulty of solving the elliptic curve problem. It is hard to forge a valid signature (\tilde{R}, s) on any message m to pass the signature verification equation $t = h(m \parallel r'_x) \pmod{q}$, where $(r'_x, r'_y) = sG + tY \pmod{q}$.

Unlinkability property means that the signer cannot link any valid signature (\tilde{R}, s) to the corresponding message m . The following theorem shows that the proposed scheme can possess unlinkability

property.

Theorem 4.6: For any valid signature (\tilde{R}, s) of the corresponding message m , the signer can derive a_i' and b_i' for any $(\tilde{m}_i, \tilde{s}_i)$ such that

$$\tilde{m}_i = (t_i + b_i') \pmod{q}$$

$$\tilde{R} = R_i + a_i' G_i - b_i' Y_i \pmod{q}$$

$$s = \tilde{s}_i + a_i' \pmod{q}$$

Proof: If $\tilde{m}_i = (t_i + b_i') \pmod{q}$ then we have $b_i' = (\tilde{m}_i - t_i) \pmod{q}$.

If $s = \tilde{s}_i + a_i' \pmod{q}$ then we have $a_i' = (s - \tilde{s}_i) \pmod{q}$.

$$\begin{aligned} & R_i + a_i' G_i - b_i' Y_i \pmod{q} \\ &= R_i + (s - \tilde{s}_i) G_i - (\tilde{m}_i - t_i) Y_i \pmod{q} \\ &= (r_i G_i + s G_i - \tilde{s}_i G_i - \tilde{m}_i Y_i + t_i Y_i) \pmod{q} \\ &= (r_i G_i + \tilde{R} - \tilde{s}_i G_i - \tilde{m}_i Y_i) \pmod{q} \\ &= (r_i G_i + \tilde{R} - (r_i + \tilde{m}_i x_i) G_i - \tilde{m}_i Y_i) \pmod{q} \\ &= (\tilde{R} - \tilde{m}_i x_i G_i - \tilde{m}_i Y_i) \pmod{q} \\ &= (\tilde{R} + \tilde{m}_i Y_i - \tilde{m}_i Y_i) \pmod{q} \\ &= \tilde{R} \pmod{q} \end{aligned}$$

According to the above derivations, the signer can derive a_i' and b_i'

for any recorded $(\tilde{m}_i, \tilde{s}_i)$. □

Therefore, for any valid signature (\tilde{R}, s) of the corresponding message m , the signer can always derive the blinding factors a_i' and b_i' for any $(\tilde{m}_i, \tilde{s}_i)$. It is demonstrate that all message-signature pairs are indistinguishable for the signer. Hence, it is computationally infeasible to derive the link between the signature and its corresponding instance of signing process.

4.3.4 Performance Comparison

Because of Schnorr's signature scheme is simple than Okamoto's signature scheme, we can reduce some storage requirements directly from Schnorr's scheme. In Yeh-Chang's scheme, the storage requirements of the elliptic curve points are 8 ($G_1, G_2, Y_1, Y_2, Y, R, \tilde{R}$ and V) and are 4 (G, Y, R and \tilde{R}) in our scheme. Yeh-Chang's scheme uses 6 (x_1, x_2, r_1, r_2, a and b) random numbers and our scheme only uses 4 (x, r, a and b) random numbers. The number of signatures are 4 ($\tilde{s}_1, \tilde{s}_2, s_1$ and s_2) in Yeh-Chang's scheme and are 2 (\tilde{s} and s) in our scheme. Hence, the proposed scheme can reduce 50% storage requirements compared with Yeh-Chang's scheme for each signature. The

comparison results of the storage requirements are shown in Table 4.1.

Table 4.1 The comparison of required storage requirements

	Our scheme	Yeh-Chang's scheme	Improvement
Elliptic curve points	4	8	50%
Random numbers	4	6	33.3%
Signatures	2	4	50%

The proposed scheme can reduce some modular multiplication operations in signature signing phase because Schnorr's signature scheme is more efficient than Okamoto's scheme. Comparing with Yeh-Chang's scheme, Table 4.2 shows that the number of multiplication, addition and negative operations can be reduced more than 36%.

Table 4.2 The comparison of required operations

	Our scheme	Yeh-Chang's scheme	Improvement
Multiplication	7	11	36%
Addition	6	12	50%
Negative	1	2	50%

Hence, the proposed scheme can reduce 50% storage requirements for each signature and speed up more than 36% performance compared with Yeh-Chang's scheme. It is efficient and more suitable for applying in thin-client applications to preserve anonymity.

Chapter 5 Applications of Some Blind Signature Schemes

Because of the networking technologies are developed rapidly, the electronic commerce is becoming more practical and important. Based on the proposed schemes, we will present two applications of the secure blind signature schemes.

The electronic cash is a popular electronic payment technique for the electronic commerce. It makes the payer from anywhere to pay his/her electronic cash conveniently through electronic communication channel. First, the untraceable fail-stop electronic cash scheme is proposed based on RSA cryptosystem in **Section 5.1**. The proposed electronic cash scheme has the fail-stop capability for the signer/bank against a forger with powerful computational capability. It also can obtain the unforgeability for the signer and the untraceability property for the participants.

Therefore, the electronic ticket system is another feasible application of electronic commerce. A generic blind signature scheme with double hashed messages is presented in **Section 5.2**. It can provide an easily

implemented solution for untraceable electronic ticket systems. Thus, we design an untraceable electronic ticket protocol based on the generic blind signature scheme for information hiding.

5.1 The Untraceable Fail-stop Electronic Cash Scheme

Chaum [8] introduced the untraceable electronic payment scheme to obtain the untraceability property for the participants. Chaum's scheme is based on RSA public key cryptosystem and its security relies on the difficulty of integer factorization problem. There are three kinds of participants: the bank, a group of payers, and a group of payees in Chaum's scheme. The payer can withdraw the electronic cash from the bank, and then pays it to the payee. The payee can forward the electronic cash to the bank and deposit the electronic cash into his/her account. The untraceability property means that the bank cannot link the electronic cash and the payer after the transactions are completed.

Brand [4] presented the untraceable off-line electronic cash scheme based on the representation problem in 1993. Okamoto [37] proposed the

universal electronic cash scheme to achieve the divisibility property. Fan and Lei [15] proposed a low computation electronic cash scheme based on the quadratic residue problem and it can reduce the amount modular computations for the payer by almost 99%. However, these schemes are only computationally secure for the signer because a forger always can forge a signature with more powerful computational capability. Thus, if a signature passes the signature verification successfully it is assumed to be generated by the owner of the private key.

In this section, we propose a RSA-based [43] electronic cash scheme which has the fail-stop capability for the signer and can obtain the untraceability property for the participants. We will also give sufficient proofs to show that our proposed scheme is secure and untraceable.



5.1.1 Chaum's Untraceable Electronic Cash Scheme

Chaum's untraceable electronic cash scheme [9] contains three kinds of participants: the bank, a group of payers, and a group of payees. There are four phases: initializing, withdrawing, unblinding, and depositing in Chaum's scheme. The details of Chaum's scheme are described as follows.

(1) Initializing: The bank chooses two large prime numbers p and q randomly. Then s/he calculates $n = p \cdot q$ and $\phi(n) = (p-1) \cdot (q-1)$. Next, the signer selects two large random numbers e and d such that $e \cdot d = 1 \pmod{\phi(n)}$ and $\text{GCD}(e, \phi(n)) = 1$. Finally, the signer publishes (n, e) as her/his public key and keeps her/his private key d secretly. Any electronic cash issued by the bank is assumed worth w dollars.

(2) Withdrawing: The payer randomly selects a number $r \in Z_n$ which is related prime to n as blinding factor. Next, s/he calculates $\alpha = r^e \cdot H(m) \pmod{n}$ and sends α to the bank, where m is the message and $H()$ is a one-way hash function. After receiving it, the bank calculates $t = \alpha^d \pmod{n}$, sends t back to the payer, and deducts w dollars from the payer's account.

(3) Unblinding: After receiving t , s/he calculates $s = r^{-1} \cdot t \pmod{n}$. The 2-tuple (m, s) is an electronic cash in Chaum's scheme.

(4) Depositing: When the payer want to pay the electronic cash, s/he can send (m, s) to the payee. The payee checks the correctness of the electronic cash by verifying whether the formula $s^e = H(m) \pmod{n}$ is true. Then s/he requests the bank to check if the electronic cash is fresh or not double-spent. If the electronic cash is correct and fresh, the payee can

accept this electronic cash. Then the bank stores (m, s) in the database for further double-spending checking and deposits w dollars to the payee's account.

Because of the blinding factor r is randomly selected and kept secretly by the payer, it is infeasible for the bank to link the payer and electronic cash. This is the untraceability property in the electronic cash scheme.

5.1.2 The Proposed Electronic Cash Scheme

Because of Chaum's scheme is only computationally secure for the signer, a forger always can forge a signature with more powerful computational capability. We propose a fail-stop electronic cash scheme to provide the fail-stop capability for the signer and the untraceability property for the participants. Three kinds of participants and four phases of the proposed scheme are the same as Chaum's scheme. In addition, a trusted dealer is needed to generate public key-pair in the initializing phase and "Proof of forgery" algorithm is provided for the signer to prove the signature is forgery. If a forgery occurs, the signer can show that the underlying computational assumption has been broken and stop the system.

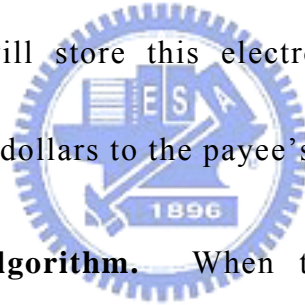
The detailed scheme is described bellow.

(1) Initializing: The trusted dealer randomly selects two large primes p and q such that $p = 2p'+1$ and $q = 2q'+1$, where p' and q' are also primes. Then s/he calculates $n = pq$ and $\phi(n) = (p-1)(q-1)$. The trusted dealer randomly chooses e_D and d_D such that $e_D d_D \equiv 1 \pmod{\phi(n)}$. Then s/he selects a random number $\alpha \in Z_n^*$ and computes $\beta = \alpha^{d_D} \pmod{n}$. The trusted dealer publishes her/his public key (α, n) , keeps his private key d_D secretly and sends (e_D, β) to the bank via secure manner. After receiving it, the bank randomly selects her/his private key (k_1, k_2, k_3, k_4) , where $k_i \in Z_n^*$ and computes $\beta_1 = \alpha^{k_4} \beta^{k_3} \pmod{n}$, $\alpha_1 = \alpha^{k_3} \beta_1^{k_1} \pmod{n}$ and $\alpha_2 = \alpha^{k_4} \beta_1^{k_2} \pmod{n}$. Finally, the bank publishes her/his public key $(\beta_1, \alpha_1, \alpha_2)$ and a one-way hash function $H()$.

(2) Withdrawing: The payer randomly selects a integer r as the blinding factor and calculates $\tilde{m} = rH(m) \pmod{n}$, where m is the message. Then the payer sends the blinded message \tilde{m} and $x = H(r) \pmod{n}$ to the bank. After receiving it, the bank computes $\tilde{s}_1 = \tilde{m}(k_1 x + k_2)$ and $\tilde{s}_2 = \tilde{m}(k_3 x + k_4)$, sends $(\tilde{s}_1, \tilde{s}_2)$ to the payer, and deducts w dollars from the payer's account.

(3) Unblinding: The payer computes $s_1 = r^{-1}\tilde{s}_1$ and $s_2 = r^{-1}\tilde{s}_2$. The 3-tuple (m, s_1, s_2) is the electronic cash in our fail-stop electronic cash scheme.

(4) Depositing: When the payee receives an electronic cash (m, s_1, s_2) from the payer, s/he can check the correctness of the electronic cash by verifying whether the formula $\alpha^{s_2} \beta_1^{s_1} = (\alpha_1^x \alpha_2)^{H(m)} \bmod n$ is true. Then s/he requests the bank to check whether the electronic cash is fresh or not. The payee accepts this electronic cash when the electronic cash is correct and fresh. The bank will store this electronic cash (m, s_1, s_2) in the database and deposits w dollars to the payee's account.



Proof of forgery algorithm. When the forged electronic cash (m, s_1', s_2') satisfies the verification formula, the bank can prove that a forgery has occurred by executing the following steps.

- a. To construct the right signature (s_1, s_2) on the message m .
- b. To compute $Z_1 = (s_1' - s_1)$ and $Z_2 = (s_2 - s_2')$.
- c. To compute $\gamma = e_D(Z_2 - k_4 Z_1) - k_3 Z_1 = c\phi(n)$
- d. To compute $\gamma = 2^h c$, where $h \in Z$ and c is odd. To select a random number $a \in Z_n^*$ and to calculate $c_0 = a^c \bmod n$, where $c_0 \neq 1$.

Next, to compute $c_i = c_{i-1}^2 \bmod n$ until $c_i = 1$ (if $c_{i-1} = -1 \bmod n$, to re-select a). Finally, to calculate $GCD(c_{i-1} + 1, n)$ to obtain the non-trivial factors of n , where i is the minimal indexing such that $c_i = 1$. (This is the Miller-Bach's method [33, 2] to factor the integer).

(5) The non-trivial factors of n is the proof of forgery.

The bank can prove that a forgery has occurred by revealing the non-trivial factors of n and then the bank can stop electronic cash scheme.

5.1.3 Security Analysis

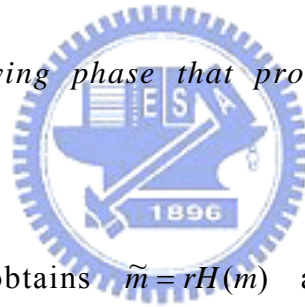
The proposed fail-stop electronic cash scheme is secure and it satisfies the following properties:

- (1) **Correctness:** Any valid electronic cash which produced by the bank can be verified through the verification formula.
- (2) **Unforgeability:** The polynomial-bounded bank cannot forge an electronic cash and the forger with more powerful computational capability is nearly infeasible to forge an electronic cash.
- (3) **Proof of forgery:** The bank can use a polynomial-time algorithm to prove that a forgery has occurred.

(4) Untraceability: Given an electronic cash that produced by the proposed scheme, the bank is computationally infeasible to trace the instance of the withdrawing phase that produces that electronic cash.

The proposed scheme is based on the fail-stop blind signature and its correctness, proof of forgery, and unforgeability properties are similar to the fail-stop blind signature scheme. We show the untraceability property is also satisfied in the proposed scheme as follows.

Theorem 4.1: *The bank is computationally infeasible to trace the instance of the withdrawing phase that produces that electronic cash (m, s_1, s_2) .*




Proof: The bank obtains $\tilde{m} = rH(m)$ and $x = H(r) \bmod n$ in the withdrawing phase. In the unblinding phase, the bank can obtain $s_1 = r^{-1}\tilde{s}_1 = (k_1x + k_2)H(m)$ and $s_2 = r^{-1}\tilde{s}_2 = (k_3x + k_4)H(m)$. Because the blinding factor r is randomly selected by the payer, the bank is computationally infeasible to trace the instance of the withdrawing phase that produces that electronic cash. The proposed electronic cash scheme satisfies the untraceability property that can preserve the anonymity of the payers.

Hence, the traditional electronic cash schemes are only

computationally secure for the bank because a forger always can forge an electronic cash with more powerful computational capability. In this section, we propose a RSA-based electronic cash scheme which has the fail-stop capability for the bank to overcome that weakness and it also can obtain the untraceability property for the participants.

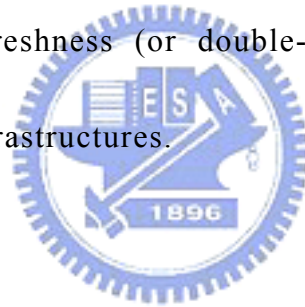
5.2 An Untraceable Electronic Ticket Scheme for Information Hiding



In carrying out electronic commerce over the internet, it is necessary to consider the case where anonymity of participator is concerned. Untraceable electronic ticket (e-ticket) makes it possible for customers to transmit their e-tickets through communication networks during transactions under privacy protection. Because the security and privacy of e-ticket can be guaranteed and the scenario of e-ticket transactions is similar to that of paper-ticket transactions, this kind of advanced digital ticket will be popular in electronic commerce.

In an untraceable electronic ticket protocol, a payer first purchases an e-ticket from the bank and then pays it to a web server for some designated

services such as movie watching, page viewing, and so on. The key point is that it is computationally infeasible for the bank to derive the link between the purchasing stage and the paying stage, i.e., given an e-ticket, the bank cannot derive the instance of the purchasing protocol which producing that e-ticket. It is usually referred to as the untraceability (or unlinkability) property [1, 5, 8, 16, 18, 41, 42] of the e-ticket. This section presents an efficient electronic ticket protocol for information hiding. Furthermore, the method can be applied to the electronic ticket systems which require freshness (or double-used) checking of e-ticket without affecting their infrastructures.



5.2.1 The Proposed Electronic Ticket Scheme

Based on the enhanced generic blind signature scheme with double hashed message described in section 4.2, the proposed electronic ticket scheme is introduced as follows. The identity of the web server is embedded into e-ticket to reduce the overhead of double-used checking. Besides, the identity of a payer is also embedded into her/his e-ticket to make this protocol more flexible. The proposed protocol consists of three parties (a bank, payers, and a group of web servers) and four stages

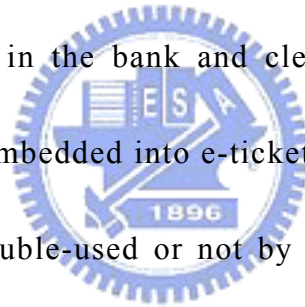
(initializing, purchasing, paying, and depositing). The bank and the payers of the electronic ticket protocol are regarded as the signer and the users of the blind signature scheme respectively. The detailed protocol is described as follows.

(1) Initializing: Initially, every payer performs an account establishment protocol with the bank to open an account in the bank.

(2) Purchasing: To purchase an e-ticket which costs w dollars for a web server with identity ID_w from the bank, a payer with identity ID_p forms a message $m = (ID_p \parallel y) \in M$, where $y \in R$ is chosen at random by the payer and \parallel is the string concatenation operator. The payer randomly chooses a blinding factor $r \in R$, and then computes and submits $B(H(H(m) \parallel ID_w), r)$ to the bank. After verifying the identity of the payer through a secure identification protocol [7, 9], the bank computes $S(B(H(H(m) \parallel ID_w), r))$ and sends it back to the payer. Then, the bank deducts w dollars from the payer's account in the bank. After receives the signing result $S(B(H(H(m) \parallel ID_w), r))$, the payer performs the unblinding operation $U(S(B(H(H(m) \parallel ID_w), r)), r)$ to obtain the signature $S(H(H(m) \parallel ID_w))$. The signature-message pair $t = (S(H(H(m) \parallel ID_w)), H(m) \parallel ID_w)$ is an electronic ticket in the protocol.

(3) Paying: When the payer pays a web server, namely ID_w , the e-ticket, s/he sends t to the web server. After verifying $V(t) = True$, the web server has to check whether the e-ticket is double-used or not. If t is not found in the web server's database which records all used e-ticket, then the web server will accept this payment. Finally, the web server stores the e-ticket in its database for future double-used checking.

(4) Depositing: When the web server's database is full or some event specified by the web server occurs, the server deposits all of the e-tickets in database into its account in the bank and clear its database. Since the identity of web server is embedded into e-ticket, each web server can check whether the e-ticket is double-used or not by itself. In other words, the traffic between the web server and the bank is largely reduced.



5.2.2 Security Analysis

The proposed electronic ticket scheme satisfies the following properties.


(1) Ownership: In some special situations such as to claim the ownership of a lost or stolen e-ticket, the e-ticket owner has to convince the bank or others of the ownership of her/his e-ticket. When a payer decides

to prove that s/he is the owner of her/his e-ticket $(S(H(H(m) \| ID_w)), H(m) \| ID_w)$, then s/he just needs to show m , where $m = (ID_p \| y)$. Due to the uninvertability property of the one-way hash function $H()$, given $H(m)$, no one except the payer knows the value of m . In fact, ID_p can be replaced by any other meaningful messages for other specific purposes.

(2) Untraceability: In the 2-tuple $t = (S(H(H(m) \| ID_w)), H(m) \| ID_w)$ produced by the above protocol, $S(H(H(m) \| ID_w))$ is the signer's signature on $H(m) \| ID_w$. According to S and V defined in **section 4.2.1**, we have that $V(t) = True$, and it is computationally infeasible for any one to compute the signature $S(H(H(m) \| ID_w))$ on $H(m) \| ID_w$ without the signing function S . Besides, due to the blinding factor r , the bank cannot link the e-ticket to the payer. In other words, given the e-ticket $(S(H(H(m) \| ID_w)), H(m) \| ID_w)$, it is computationally infeasible for the bank to derive the instance of the purchasing stage which produces that e-ticket. Thus, the proposed scheme can achieve the untraceability/unlinkability property.

Chapter 6 Conclusions

We have presented cryptanalysis on a new Rabin-like blind signature scheme to show that Chen et al.'s scheme can be compromised when choosing some particular blinding factors. Then, the traceability attack claims on RSA-Based partially blind signature scheme, ElGamal blind signature scheme and proxy blind signature schemes are also analyzed and corrected in this dissertation.



A fail-stop blind signature scheme is proposed to protect the signer against a forger with more powerful computational capability and obtain the anonymity property for the participants. It can be applied in more critical system like electronic payment systems which need higher security against more powerful forger. We also presented an improved blind signature scheme based on the elliptic curve cryptosystem. Comparing with Yeh-Chang's scheme, it can reduce 50% storage requirements for each signature and speed up performance more than 36%.

Based on the proposed fail-stop blind signature, we have constructed a untraceable fail-stop electronic cash scheme. The proposed electronic

cash scheme has the capability for the bank to stop the electronic cash scheme when a signature is forged. It can obtain the unforgeability for the bank and the untraceability property for the participants. We have presented a generic blind signature scheme and design an untraceable electronic ticket protocol for information hiding. The untraceable electronic ticket protocol can check whether the e-ticket is double-used.

In future research, we will consider to design some signature schemes with fail-stop capability for the signer against the forger with more powerful computational capability. Moreover, the untraceable signature scheme is also the important issue to be discussed.



Bibliography

- [1] M. Abe and E. Fujisaki, "How to Date Blind Signatures," *Advances in Cryptology-ASIACRYPT*, Springer-Verlag, pp. 224-251, 1996.
- [2] E. Bach, "Discrete Logarithm and Factoring," *Report no. UCB/CSD 84/186, Comp. Sc. Division (EECS)*, University of California, Berkeley, 1984.
- [3] D. Boneh and M. Franklin, "Efficient Generation of Shared RSA keys," *Advances in Cryptology - CRYPTO*, Springer-Verlag, pp. 425-439, 1997.
- [4] S. Brand, "Untraceable Off-line Cash in Wallets with Observers," *Advances in Cryptology - CRYPTO*, Springer-Verlag, pp. 302-318, 1993.
- [5] J. L. Camenisch, J. M. Piveteau and M. A. Stadler, "Blind Signatures Based on the Discrete Logarithm Problem," *Advances in Cryptology - EUROCRYPT*, pp. 428-432, 1994.
- [6] K. C. Chang, E. H. Lu and P. C. Su, "Fail-stop blind signature scheme design based on pairings," *Applied Mathematics and Computation*, vol. 169, pp. 1324-1331, 2005.
- [7] D. Chaum, "Blind Signature Systems," *Advances in Cryptology -*

CRYPTO, Springer-Verlag, pp. 153, 1983.

- [8] D. Chaum, "Blind Signatures for Untraceable Payments," *Advances in Cryptology - CRYPTO*, Springer-Verlag, pp. 199-203, 1982.
- [9] D. Chaum, "Untraceable Electronic Cash," *Advances in Cryptology – CRYPTO*, Springer-Verlag, pp. 319-327, 1990.
- [10] K. Chen, W. Qiu, and D. Zheng, "New Rabin-like signature scheme," *Workshop Proceedings of the Seventh International Conference on Distributed Multimedia Systems*, Knowledge Systems Institute, pp. 185-188, 2001.
- [11] H. Y. Chien, J. K. Jan and Y. M. Tseng, "RSA-Based Partially Bind Signature with Low Computation," *Parallel and Distributed Systems*, IEEE Computer Society Press, no. 26-29, pp. 385-389, 2001.
- [12] W. Diffie and M. Hellman, "New Directions in Cryptography," *IEEE Trans. Information Theory*, pp. 644-654, 1976.
- [13] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE Trans. Information Theory*, vol. 31, no. 4, pp. 469-472, 1985.
- [14] C. I. Fan, "Comments on Hwang-Lee-Lai Attack upon Fan-Lee Partially Blind Signature Scheme," *IEICE Trans. Fundamentals*,

vol. E86-A, no. 7, pp. 1900-1901, 2003.

- [15] C. I. Fan and C. L. Lei, "Low-computation Partially Blind Signatures for Electronic Cash," *IEICE Trans. Fundamentals*, vol. E81-A, pp. 940-949, 1998.
- [16] C. I. Fan and C. L. Lei, "User efficient blind signatures," *Electronics Letters*, vol. 34, no. 6, pp. 544-546, 1998.
- [17] C. I. Fan, W. K. Chen, and Y. S. Yeh, "Blind signatures with double-hashed messages for fair electronic elections and ownership claimable digital cash," *Proceedings of First International Conference on Enterprise Information Systems*, pp. 612-618, 1999.
- [18] N. Ferguson, "Single term off-line coins," *Advances in Cryptology - EUROCRYPT*, Springer-Verlag, pp. 318-328, 1994.
- [19] L. Harn, "Cryptanalysis of the Blind Signatures Based on the Discrete Logarithm Problem," *Electronics Letters*, vol. 31, no. 14, pp. 1136, 1995.
- [20] P. Hoster, M. Michels and H. Petersen, "Comment: Cryptanalysis of the Blind Signatures Based on the Discrete Logarithm Problem," *Electronics Letters*, vol. 31, no. 21, pp. 1827, 1995.
- [21] M. S. Hwang, C. C. Lee and Y. C. Lai, "Traceability on RSA-Based Partially Signature with Low Computation," *Applied Mathematics and Computation*, vol. 145, no. 2, pp. 465-468, 2003.

- [22] M. S. Hwang, C. C. Lee and Y. C. Lai, "An Untraceable Blind Signature Scheme," *IEICE Trans. Fundamentals*, vol. E86-A, no. 7, pp. 1902-1906, 2003.
- [23] M. S. Hwang, C. C. Lee and Y. C. Lai, "Traceability on Stadler et al.'s Fair Blind Signature Scheme," *IEICE Trans. Fundamentals*, vol. E86-A, no. 2, pp. 513-514, 2003.
- [24] M. S. Hwang, C. C. Lee and Y. C. Lai, "Traceability on Low-Computation Partially Blind Signatures for Electronic Cash," *IEICE Trans. Fundamentals*, vol. E85-A, no. 5, pp. 1181-1182, 2002.
- [25] S. Lal and A. K. Awasthi, "Proxy Blind Signature Scheme," *Cryptology ePrint Archive Report 2003/072*. Available from <http://eprint.iacr.org>.
- [26] L. Lamport, "Constructing Digital Signatures from a One-way Function," *Technical Report CSL-98*, SRI International, 1979.
- [27] C. C. Lee, M. S. Hwang and W. P. Yang, "A New Blind Signature based on the Discrete Logarithm Problem for Untraceability," *Applied Mathematics and Computation*, vol.164, pp. 837-841, 2005.
- [28] N. Y. Lee and M. K. Sun, "Analysis on Traceability on Stadler et al.'s Fair Blind Signature," *IEICE Trans. Fundamentals*, vol. E86-A, no. 11, pp. 2901-2902, 2003.

- [29] N. Y. Lee and C. N. Wu, "Comment on Traceability Analysis on Chaum Blind Signature," *IEICE Trans. Fundamentals*, vol. E87-A, no. 2, pp. 511-512, 2004.
- [30] M. Mambo, K. Usuda and K. Okamoto, "Proxy Signature : Delegation of the Power to Sign Messages," *IEICE Trans. Fundamentals*, vol. E79-A, no. 9, pp. 1338-1353, 1996.
- [31] A. J. Menezes, Elliptic Curve Public Key Cryptosystem, *Kluwer Academic Publishers*, 1993.
- [32] A. J. Menezes, P. C. Van Oorschot and S. A. Vanstone, Handbook of Applied Cryptography, *CRC Press*, 1996.
- [33] G. L. Miller, "Riemann's Hypothesis and Tests for Primality," *Journal of Computer and System Sciences*, vol. 13, pp. 300-317, 1976.
- [34] NIST FIPS PUB 186-2, "Digital Signature Standard (DSS)," *National Institute of Standards and Technology*, U.S. Department of Commerce, 2000.
- [35] K. Nyberg, R. A. Rueppel, "A New Signature Scheme Based on the DSA Giving Message Recovery," *1st ACM Conference on Computer and Communications Security*, pp. 58-61, 1993.
- [36] T. Okamoto, "Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes," *Advances in*

- Cryptology - CRYPTO*, Springer-Verlag, pp. 31-53, 1992.
- [37] T. Okamoto and K. Ohta, "Universal electronic cash," *Advances in Cryptology - CRYPTO*, Springer-Verlag, pp. 324-337, 1992.
- [38] B. Pfitzmann, *Digital Signature Schemes : General Framework and Fail-Stop Signatures*, Springer-Verlag, 1996.
- [39] B. Pfitzmann and M. Waidner, "Fail-Stop Signatures and Their Applications," *SECURICOM P1*, pp. 145-160, 1991.
- [40] M. Rabin, "Digitalized signatures and public key functions as intractable as factorization," *MIT/LCS/TR-212*, MIT Laboratory for Computer Science, 1979.
- [41] D. Pointcheval and J. Stern, "Provably secure blind signature schemes," *Advances in Cryptology - ASIACRYPT*, Springer-Verlag, pp. 252-265, 1996.
- [42] D. Pointcheval and J. Stern, "New blind signatures equivalent to factorization," *Proceedings of the 4th ACM Conference on Computer and Communication Security*, pp. 92-99, 1997.
- [43] R. L. Rivest, A. Shamir, and L. M. Adleman, "A Method for Obtaining Digital Signatures and Public-key Cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120-126, 1978.
- [44] B. Schneier, *Applied Cryptography*, John Wiley & Sons, 2000.

- [45] C. P. Schnorr, "Efficient Identification and Signatures for Smart Cards," *Advances in Cryptography - CRYPTO*, Springer-Verlag, pp. 235-251, 1990.
- [46] D. R. Stinson, *Cryptography: Theory and Practice*, 2 Edition, CRC Press, 2002.
- [47] H. M. Sun, B. T. Hsieh and S. M. Tseng, "On the Security of Some Proxy Blind Signature Scheme," *Journal of Systems and Software*, vol.74, pp. 297-302, 2005.
- [48] W. Susilo, R. Safavi-Naini, and J. Pieprzyk, "RSA-based Fail-Stop Signature Schemes," *International Workshop on Security*, IEEE Computer Society Press), pp. 161-166, 1999.
- [49] Z. Tan, Z. Liu and C. Tang, "Digital Proxy Blind Signature Schemes Based on DLP and ECDLP," *MM Research Preprints*, No. 21, MMRC, AMSS, Academic, pp. 212-217, 2004.
- [50] M. Waidner and B. Pfitzmann, "The Dining Cryptographers in the Disco: unconditional sender and recipient untraceability with computationally secure serviceability," *Advances in Cryptology - EUROCRYPT*, Springer-Verlag, pp. 690, 1989.
- [51] Y. S. Yeh and M. H. Chang, "Schnorr Blind Signature Scheme based on the Elliptic Curves," *Asia Journal of Information Technology*, vol. 2, no. 3, pp.130-134, 2003.