THRESHOLD SIGNATURE SCHEME BASED ON FACTORING AND DISCRETE LOGARITHMS PROBLEMS

MOHD SAIFUL ADLI MOHAMAD

HASLINDA IBRAHIM

SCHOOL OF QUANTITATIVE SCIENCES
UUM COLLEGE OF ARTS AND SCIENCES
UNIVERSITI UTARA MALAYSIA

2012

PENGAKUAN TANGGUNGJAWAB
(DISCLAIMER)

Kami, dengan ini, mengaku bertanggungjawab diatas ketepatan semua pandangan, komen teknikal, laporan fakta, data, gambarajah, ilustrasi, dan gambar foto yang telah diutarakan dalam laporan ini. Kami bertanggungjawab sepenuhnya bahawa bahan yang diserahkan ini telah disemak daripada aspek hakcipta dan hak keempunyaan. Universiti Utara Malaysia tidak bertanggungjawab terhadap ketepatan mana-mana komen, laporan, dan maklumat teknikal dan fakta lain, dan terhadap tuntutan hakcipta dan juga hak keempunyaan.

We are responsible for the accuracy of all opinions, technical comments, factual reports, data, figures, illustrations and photographs in the article. We bear full responsibility for checking whether material submitted is subject to copy right or ownership right. Universiti Utara Malaysia does not accept any liability for the accuracy of such comment, report and other technical and factual information and the copy right or ownership right claims.

Ketua Penyelidik:

Mohd Saiful Adli Mohamad

Ahli:

Haslinda Ibrahim

ACKNOWLEDGEMENT

First and foremost, Praise to Allah for all His blessings and guidance who has given us all we need to complete this research. We are immensely pleased to place on record our profound gratitude and heartfelt thanks to Universiti Utara Malaysia for providing grant for us to study the research problem at hand.

We would like to acknowledge and extend our appreciation to the faculty members who have helped supported us directly or indirectly, throughout the research process. Also a lot of thanks to Dr Eddie Shahril Ismail from Universiti Kebangsaan Malaysia for his help and guidance in completing this research. Last but not least, we also would like to express our deepest gratitude to the officials and other staff members of Research and Innovation Management Center (RIMC) of UUM who rendered their help during the period of our research work.

ABSTRACT

Digital signature scheme is a mathematical scheme for demonstrating the authenticity of a digital message or document. The ordinary digital signature scheme allows single user to sign an online transaction. For the reason of integrity, nowadays many documents and online transactions need to be signed by more than one person in an organization. Threshold digital signature scheme are developed to overcome this problem. In a threshold signature scheme, t out of n members are required to sign an online transaction. In this research, we developed a new threshold signature scheme based on two number theory problems, namely factoring and discrete logarithms. The advantage of our scheme is based on the fact that it is very hard to solve both problems simultaneously. The security analysis of our scheme shows that our scheme is invulnerable against several securities threat, while performance evaluation shows that our scheme requires reasonable number of operations in signing and verifying.

ABSTRAK

Skema tandatangan digital ialah suatu skema matematik untuk menunjukkan kesahihan sesuatu mesej atau dokumen digital. Skema tandatangan digital biasa membenarkan seorang pengguna untuk menandatangani sesuatu transaksi atas talian. Walau bagaimanapun, diatas faktor integriti, kebanyakan dokumen dan transaksi atas talian pada masa kini memerlukan lebih daripada seorang penandatangan. Oleh sebab itu, skema tandatangan digital berkumpulan dibangunkan untuk mengatasi masalah ini. Dalam skema tandatangan digital berkumpulan, t daripada n pengguna diperlukan untuk menandatangani sesuatu transaksi atas talian. Dalam kajian ini, suatu skema tandatangan digital berkumpulan berasaskan masalah pemfaktoran dan logaritma diskret dibangunkan. Kekuatan skema baharu ini adalah berdasarkan fakta bahawa ianya adalah terlalu sukar untuk penggodam untuk menyelesaikan kedua-dua masalah pemfaktoran dan logaritma diskret secara serentak. Analisis keselamatan pula menunjukkan skema baharu ini kebal terhadap beberapa ancaman keselamatan, manakala penilaian efisiensi pula menunjukkan skema baharu ini memerlukan bilangan operasi yang munasabah dalam kedua-dua langkah menandatangan mesej dan mengesahkan tandatangan.

TABLE OF CONTENTS

CONTENTS

PAGE

CHAPTER 1

INTRODUCTION

1.1 RESEARCH BACKGROUND

Cryptography is a science or study of methods for sending messages in secret (encipher or disguise messages) so that only the intended recipient can remove the disguise and read the messages (decipher it). The term cryptography itself comes from the Greek words; *kryptos* (hidden) and *graphein* (write). In cryptography, mathematicians and computer scientists play important role in designing cryptography systems. Mathematicians play the role to develop the algorithms/schemes of cryptography (cryptosystem, digital signature, authenticated encryption, etc.), while computer scientists develop the computer systems from the mathematical schemes.

Digital signature, which is one of the areas in cryptography, is a mathematical scheme for demonstrating the authenticity of a digital message or document. The concept of digital signature was introduced by Diffie and Hellman (1976). However, in their outstanding paper "New Direction in Cryptography", they just provide the idea of digital signature but not a scheme that could be implemented in practice. It was Rivest et. al. (1978) who introduced the first digital signature scheme based on the hardness of finding the prime factors of a large composite integer (factoring problem). Then,

ElGamal (1985) developed a new digital signature scheme based on discrete logarithms problem.

Currently, digital signatures are widely used for software distribution, internet-based transactions, e-commerce, online file movement system, etc. In general, a digital signature scheme must satisfy the following properties:

a) **Authentic**

   The signature convinces the document's recipient that the signer truly signed the document.

b) **Not forgeable**

   Nobody else except the signer can deliberately sign the document.

c) **Not reusable**

   The signature is a part of document. No one can transfer the signature to other document.

d) **Unalterable**

   After the document is signed, it cannot be altered.

e) **Non-repudiated**

   After signing the document, the signer cannot later claim that he or she did not sign it.

Nowadays, many electronic documents need to be signed by more than one person. This problem brings the idea of society oriented cryptography, which is known as threshold cryptography (Desmedt, 1988; Desmedt & Frankel, 1989). The idea of the threshold cryptosystems introduced Desmedt used the concept of Shamir's secret sharing (Shamir, 1979), which is based on Lagrange interpolation technique. However, the first threshold digital signature scheme was proposed by Desmedt and Frankel

(1991), where they applied the factoring problem in their scheme. Later, Harn (1994) proposed another threshold digital signature scheme from modified ElGamal scheme, which is based on discrete logarithm problem. In his paper, he stated the following properties of a (t, n) threshold signature:

a) Any group signature is mutually generated by at least t group members.
b) The size of the group signature is equivalent to the size of an individual signature.
c) The signature verification process is simplified because there is only one group public key required.
d) The group signature can be verified by any outsider.
e) The group holds the responsibility to the signed message.

1.2 PROBLEM STATEMENT

In the last three decades, many digital signature schemes have being developed based on various number theoretic problems such as factoring, discrete logarithm, and elliptic curve. Although the single-problem schemes remain unsolved today, but it is understood that one day such problems could be solved. When this happens, the signature based on those single problems no longer secured. That's the reason why recent ordinary digital signatures were developed based on multiple hard number theoretic problems (Lee & Hwang, 1996; Laih & Kuo, 1997; He, 2001; Wang & Chang, 2003). However, in threshold digital signature, no scheme based on multiple problems has being applied. Due to this situation, we would like to apply the concept of multiple hard number theoretic problems in threshold digital signature. It is a strong assumption that the signature with multiple hard number theoretic problems will provide more security than the signature with single problem (Ismail, 2009).

1.3 OBJECTIVES

The objectives of this study are:

i. To develop a new threshold signature scheme based on two hard problems, namely factoring and discrete logarithm.
ii. To analyze the security of the new scheme.
iii. To evaluate the performance of the developed scheme.

1.4 SIGNIFICANCE OF THE STUDY

With the new threshold signature scheme, it is expected that computer scientist can develop a more secure system in online services and transactions. Besides that, it will also support the environmentalists to reduce the usage of paper and encourage paperless documentation. Moreover, hopefully with this effort, mathematicians and cryptographer will do more research in developing more secure and efficient threshold signature schemes in future.

1.5 ORGANIZATION OF THE REPORT

This report is organized and presented in five chapters. In Chapter 1, the research background, problem statements, objectives, and significance of the study are stated. In Chapter 2, some tools in cryptography are reviewed. Some groups in algebra that we need in our study are reviewed and then some topics in number theory also will be discussed. In Chapter 3, the new threshold signature scheme will be introduced. In this chapter, we will discuss all steps involve in the scheme and show an example of the scheme. The security analysis and performance evaluation will be discussed in Chapter

4. Finally, we will present the summary and conclusions of the findings and provide suggestions for future research.

CHAPTER 2

MATHEMATICAL TOOLS IN CRYPTOGRAPHY

In this chapter, we present mathematical definitions and results that will be needed for the scheme development. Then, we briefly discuss some cryptographic functions that widely used in developing cryptographic schemes.

2.1 ALGORITHMS

In general, an algorithm is used as a tool for solving a well-specified computational problem. The problem statement specifies in general terms the desired input-output relationship. The algorithm describes a specific computational procedure for achieving the input-output relationship. It is usually of interest to find the most efficient algorithm for solving a given computational problem. The time that an algorithm takes to halt depends on the size of the problem instance. Also, the unit of time used should be made precise, especially when comparing the performance of two algorithms. The formal definition of algorithm is given as follows:

Definition 2.1 An algorithm is any well-defined computational procedure that takes a variable input and halts with an outputs. An algorithm is thus a sequence of computational steps that transform the input into output.

2.2 GROUP THEORY

Group theory plays an important role in digital signatures. In this section, we will review some topics in group theory that will be used in our threshold signature scheme.

2.2.1 Groups

Let S be a nonempty set and $*$ be a binary operation that maps elements in $S \times S$ to S; mathematically we write as $*\colon (a, b) \mapsto a * b$.

Definition 2.2 A binary operation $*$ is called commutative and associative if

$$a * b = b * a \text{ and } (a * b) * c = a * (b * c)$$

holds for all $a, b, c \in S$ respectively.

Definition 2.3 A group, denoted as $\langle G,* \rangle$, is a set G together with a binary operation $*$ on elements of G such that the following are satisfied:

i. The operation $*$ is associative.

ii. G contains an element e, called as identity element such that

$$e * a = a * e = a$$

holds for all $a \in G$.

iii. For every a in G, there exists an element b such that

$$a * b = b * a = e.$$

The element b is called as inverse of a under the binary operation $*$.

If the operation $*$ is commutative, then the group is called abelian group. The group is called finite if $|G|$ (cardinality of G) is finite and the number of elements of a finite group is called as its order. In a group, the identity element is unique as is the inverse of any element.

Definition 2.4 A group G is called cyclic group if there exists an element $a \in G$ such that for every element $b \in G$ can be written in the form of $b = a^x$ for some $x \in \mathbb{Z}$. Such an element a is called a generator of G and can be written as $\langle a \rangle = G$ to indicate that a generates G.

Definition 2.5 The order of an element $b \in G$, denoted by $\mathrm{ord}(b)$, is the smallest positive integer n such that $b^n = 1$.

The order of any element of a finite group divides the order of the group. If a is a generator of a cyclic group of order m, then the element $b = a^i$ has order $m|\gcd(m, i)$. In particular, b is a generator of G if and only if $\gcd(m, i) = 1$. Hence, if m is a prime, then every element different from 1 is a generator of G.

In our scheme, we consider the following two important groups:

i. The set of integers modulo n together with addition modulo n constitutes an abelian group of order n. This group is denoted by $\mathbb{Z}_n = \{0,1,2, \dots, n-1\}$.
ii. The set formed by the positive integers smaller than n and relatively prime to n together with the multiplication modulo n. This group is called as the multiplicative group and denoted by $\mathbb{Z}_n^* = \{k | 1 \leq k < m, \gcd(k,m) = 1\}$.

2.3 NUMBER THEORY

Number theory plays important role in cryptography. In modern cryptography, messages are represented in numerical value and the steps or processes in cryptography, either in cryptosystem or digital signature, are being done by mathematical operation. In this section, we discussed some important topics in number theory that will be used in our scheme.

2.3.1 Divisibility

Definition 2.6 An integer b is said to be divisible by an integer $a \neq 0$, denoted by $a|b$, if there exists some integer c such that $b = ac$. We write $a \nmid b$ to indicate that b is not divisible by a.

Definition 2.7 Let a and b be given integers, with at least one of them different from zero. The greatest common divisor of a and b, denoted by $\gcd(a, b)$, is the positive integer d satisfying the following conditions:

i. $d|a$ and $d|b$.
ii. If $c|a$ and $c|b$, then $c \leq d$.

2.3.2 Congruences

One of the most basic in number theory is congruences, or also known as modular arithmetic. Here, we review some important definitions and theorems that are related with our threshold signature scheme.

Definition 2.8 Let $a, b,$ and n be integers, with $n \neq 0$. We say that a is congruent to b modulo n, denoted by $a \equiv b \pmod{n}$, if $a - b$ is a multiple of n.

Another formulation is that $a \equiv b \pmod{n}$ if a and b differ by a multiple of n. This can be written as $a = b + nk$ for some integer k (positive or negative).

Theorem 2.1 Let $a, b, c,$ and n be integers with $n \neq 0$.

i. $a \equiv 0 \pmod{n}$ if and only if $n|a$.
ii. $a \equiv a \pmod{n}$.
iii. $a \equiv b \pmod{n}$ if and only if $b \equiv a \pmod{n}$.
iv. If $a \equiv b \pmod{n}$ and $b \equiv c \pmod{n}$, then $a \equiv c \pmod{n}$.

Proof:

i. $a \equiv 0 \pmod{n}$ means that $a = a - 0$ is a multiple of n, which is the same as $n|a$.
ii. $a - a = 0 \cdot n$, so $a \equiv a \pmod{n}$.
iii. If $a \equiv b \pmod{n}$, we can write it as $a - b = nk$. Then, $b - a = n(-k)$, so $b \equiv a \pmod{n}$. Reversing the roles of a and b gives the reverse implication.
iv. Write $a = b + nk$ and $c = b + nl$. Then, $a - c = nk - nl = n(k - l)$. So, $a \equiv c \pmod{n}$.

□

Definition 2.9 Suppose that $a \in \mathbb{Z}_n$. The multiplicative inverse of a is an element $b \in \mathbb{Z}_n$ such that $ab \equiv 1 \pmod{n}$, or $b \equiv a^{-1} \pmod{n}$.

2.3.3 Prime Numbers

In number theory, the concept of prime numbers is widely used in mathematical computation of cryptography. Many hard problems in cryptography, such as factoring and discrete logarithm, use prime numbers in the difficulty of certain computation. In this section, we recall the definitions and theorems relate to the prime numbers.

Definition 2.10 A number $p > 1$ that is divisible only by 1 and itself is called a prime number. An integer $m > 2$ that is not a prime is called a composite number.

Theorem 2.2 Every positive integer n can be expressed as a product of primes, symbolically written as $n = {p_1}^{a_1}{p_2}^{a_2} \dots {p_l}^{a_l}$, where p_i for $i = \{1, 2, \dots, l\}$ are distinct primes. This factorization is unique, up to reordering the factors.

Proof:

There is a small technicality that must be dealt with before we begin. When dealing with products, it is convenient to make the convention that an empty product equals 1. This is similar to the convention that $x^0 = 1$. Therefore, the positive integer 1 is a product of primes, namely the empty product. Also, each prime is regarded as a one factor product of primes.

Suppose there exist positive integers that are not product of primes. Let n be the smallest such integer. Then, n cannot be 1 (the empty product), or a prime (a one factor product), so n must be composite. Therefore, $n = ab$ with $1 < a, b < n$. Since n is the smallest positive integer that is not a product of primes, both a and b are products of primes. But a product of primes times a product of primes is a product of primes, so $n = ab$ is a product of primes. This contradiction shows that the set of integers that are not products of primes must be the empty set. Therefore, every positive integer is a product of primes.

□

Definition 2.11 Suppose that $a, b \in \mathbb{Z}$. We say that a and b are relatively prime if $\gcd(a, b) = 1$. (Note that a and b not necessarily primes).

Theorem 2.3 Let g be a primitive root for the prime p.

i. If n is an integer, then $g^n \equiv 1 \pmod{p}$ if and only if $n \equiv 0 \pmod{p-1}$.
ii. If $j, k \in \mathbb{Z}$, then $g^j \equiv g^k \pmod{p}$ if and only if $j \equiv k \pmod{p-1}$.

Proof:

i. If $n \equiv 0 \pmod{p-1}$, then $n = (p-1)m$ for some m. Therefore,

$$g^n \equiv (g^m)^{p-1} \equiv 1 \pmod{p}$$

by Fermat's theorem.

ii. Suppose that $g^j \equiv g^k \pmod{p}$. Dividing both side by g^k yields $g^{j-k} \equiv 1 \pmod{p}$. By (i), $j - k \equiv 0 \pmod{p-1}$ so $j \equiv k \pmod{p-1}$.

□

Definition 2.12 Let n be a positive integer. The Euler-phi function, $\phi(n)$ is defined as follows:

$$\phi(n) = |\{k | 1 \leq k < n, \gcd(k, n) = 1\}|$$

Equivalently, for an integer $n = \prod_{i=1}^{l} {p_i}^{a_i}$,, where p_i's are distinct primes, we have that

$$\phi(n) = n \prod_{i=1}^{l} \left(1 - \frac{1}{p_i}\right).$$

Two of the most basic theorems in number theory are Fermat's little theorem and Euler's theorem. These two theorems recently proved to have important cryptographic applications and will be used in our scheme.

Theorem 2.4 (*Fermat's Little Theorem*) If p is a prime and p does not divide a, then

$$a^{p-1} \equiv 1 \pmod{p}$$

Proof:

We begin by considering the first $p-1$ positive multiples of a; that is the integers $a, 2a, 3a, \dots, (p-1)a$. None of these numbers is congruent modulo p to any other, nor is any congruent to zero. Indeed, if it happened that

$$ra \equiv sa \pmod{p}, \text{for } 1 \leq r < s \leq p-1$$

then a could be cancelled to give $r \equiv s \pmod{p}$, which is impossible. Therefore, the previous set of integers must be congruent modulo p to $1, 2, 3, \dots, p-1$, taken in some order. Multiplying all these congruences together, we find that

$$a \cdot 2a \cdot 3a \cdots (p-1)a \equiv 1 \cdot 2 \cdot 3 \cdots (p-1) \pmod{p}$$

whence

$$a^{p-1}(p-1)! \equiv (p-1)! \pmod{p}$$

Once $(p-1)!$ is cancelled from both side of the preceding congruence (this is possible because $p \nmid (p-1)!$), our line of reasoning culminates in the statement that $a^{p-1} \equiv 1 \pmod{p}$, which is Fermat's little theorem.

□

Theorem 2.5 (*Euler's Theorem*) If a and n are positive integers, where $\gcd(a,n) = 1$, then

$$a^{\phi(n)} \equiv 1 \pmod{n}$$

Proof:

Take $n > 1$. Let $a_1, a_2, \dots, a_{\phi(n)}$ be the positive integers less than n that are relatively prime to n. Because $\gcd(a,n) = 1$, it follows that $aa_1, aa_2, \dots, aa_{\phi(n)}$ are congruent, not necessarily in order to appearance, to $a_1, a_2, \dots, a_{\phi(n)}$. Then,

$$aa_1 \equiv a_1' \pmod{n}$$

$$aa_2 \equiv a_2' \pmod{n}$$

$$\vdots$$

$$aa_{\phi(n)} \equiv a_{\phi(n)}' \pmod{n}$$

where $a_1', a_2', \ldots, a_{\phi(n)}'$ are the integers $a_1, a_2, \ldots, a_{\phi(n)}$ in some order. On taking the product of these $\phi(n)$ congruences, we get

$$(aa_1)(aa_2) \ldots \left(aa_{\phi(n)}\right) \equiv a_1' a_2' \ldots a_{\phi(n)}' \pmod{n}$$

$$\equiv a_1 a_2 \ldots a_{\phi(n)} \pmod{n}$$

and so

$$a^{\phi(n)}\left(a_1 a_2 \ldots a_{\phi(n)}\right) \equiv a_1 a_2 \ldots a_{\phi(n)} \pmod{n}$$

Because $\gcd(a_i, n) = 1$ for each i, it implies that $\gcd\left(a_1 a_2 \ldots a_{\phi(n)}, n\right) = 1$. Therefore, we may divide both sides of the foregoing congruence by the common factor $a_1 a_2 \ldots a_{\phi(n)}$, leaving us with

$$a^{\phi(n)} \equiv 1 \pmod{n}$$

□

2.4 HARD PROBLEMS IN CRYPTOGRAPHY

In cryptography, there are many hard number theoretic problems have been used to design a scheme. Some examples of the problems that commonly use in cryptography are factoring, residuosity, discrete logarithm, and elliptic curve. In this section, we

discussed two of the problems, factoring and discrete logarithm, that being used in our threshold signature scheme.

2.4.1 Factoring Problem

Factoring problem is one of the most popular problems in cryptography. Since Rivest et. al. (1978) introduced factoring problem in cryptography, many digital signature schemes and cryptosystems were developed based on this problem. Actually, there are many algorithms that can solve this problem but they need unreasonable amount of time and memory.

Definition 2.13 The integer factorization problem (FAC): Given a positive integer n, find its prime power factorization, i.e., find pair-wise distinct primes p_i and positive integers e_i such that $n = {p_1}^{e_1} \cdot {p_2}^{e_2} \cdot \ldots \cdot {p_k}^{e_k}$.

2.4.2 Discrete Logarithm Problem

Another popular problem in cryptography is discrete logarithms. It is ElGamal (1985), who introduced this problem in cryptography. In his paper, he used discrete logarithm problem to secure his protocol of ElGamal public key encryption and digital signature.

Definition 2.14 Let G be a finite cyclic group and $g \in G$ be a generator of G. The discrete logarithm problem of some element $a \in G$, denoted $log_g a$, is the unique integer x, $0 \leq x < |G|$, such that $a = g^x$.

If g is not a generator, the notion of the discrete logarithm of a to the base g is extended to be the smallest integer x, such that $a = g^x$, if it exists. Here, the discrete logarithm is called the index of the element a.

Definition 2.15 The discrete logarithm problem (DLP): Given a finite cyclic group G, a generator $g \in G$, and an element a, find the integer x, $0 \leq x < |G| - 1$, such that $a = g^x$ holds.

2.5 CRYPTOGRAPHIC FUNCTIONS

Functions in cryptography are important in constructing a cryptographic system. The functions will make sure a cryptographic system or digital signature scheme remain hard to break. Here, we discuss some types of functions that widely used in developing cryptographic system.

Definition 2.16 A function is defined by two set X and Y and a rule f, which assigns to each element in X precisely one element in Y. The set X is called domain of the function while Y is the range of the function. If x is an element of X, the image of x (denoted by $y = f(x)$) is the element in Y, which the rule f associates with x. If $y \in Y$, then the pre-image of y is an element $x \in X$ for which $f(x) = y$.

2.5.1 One-Way Function

Definition 2.17 A function $f: X \to Y$ is called a one-way function if $f(x)$ is easy to compute for all $x \in X$, but it is difficult to obtain x from $f(x)$.

A clear example of this function can be seen in discrete logarithm problem. Let $X = Y = Z_p^*$, and define $f(x) = g^x \pmod{p}$, where g is a generator of the multiplicative group. Obviously, given $x \in X$, it is easy to compute $f(x)$ but given g^x, it is hard to obtain x.

2.5.2 Trapdoor One-Way Function

Definition 2.18 A one-way trapdoor function is a one-way function $f: X \to Y$ with the additional property that given some extra information, called trapdoor information, that is it become easy to find any given $y \in Y$, an $x \in X$ such that $f(x) = y$.

An example for this trapdoor one-way function can be shown in factoring problem. Consider a function defined by $f(m) = m^e \pmod{pq}$, where p and q are two large primes and e is an integer satisfying $\gcd\big(e, (p-1)(q-1)\big) = 1$. Given m, it is easy to compute $f(m)$. However, given $f(m)$, it is hard to derive m unless we know a trapdoor d such that $ed = 1 \pmod{(p-1)(q-1)}$.

2.5.3 Hash Function

Cryptographic hash functions play an important role in signature scheme. They are used to reduce long messages of arbitrary length to a fixed length of bit-strings so that the running of signing algorithm can be made even faster.

Definition 2.19 A has function is a function $H: \{0,1\}^* \rightarrow \{0,1\}^l$ mapping binary strings of arbitrary finite length to binary strings of a fixed length l.

For current security, an output of size 160-bits of hash function is reasonable. In cryptographic requirement, the hash function must efficiently computable and satisfies one of the following properties:

1) Weak collision-resistant: For a given a, it is hard to find $b \neq a$ such that $H(a) = H(b)$.
2) Strong collision-resistant: It is hard to find a pair (a, b) with $b \neq a$ such that $H(a) = H(b)$ if H is chosen randomly from a family of hash function.

CHAPTER 3

THRESHOLD SIGNATURE SCHEME

In previous chapter, we already discussed about tools that regularly used in constructing cryptographic systems and digital signature schemes. In this chapter, we will introduce a new threshold signature scheme based on two hard number theoretical problems, namely factoring and discrete logarithm.

The security of this threshold signature scheme is based on the difficulty of solving both problems, factoring and discrete logarithm, simultaneously. This scheme is modified from the ordinary signature scheme proposed by Ismail et. al. (2009). In this scheme, a trusted dealer (TD) is required to generate the parameters and keys for the users and group. TD also plays the role verify the partial signatures and construct the group signature.

Basically, a digital signature scheme consists of three steps; generating keys, signing message, and verifying signature. In the key generation step, TD will generate the public and secret keys. In ordinary signature scheme, the secret keys will be kept by an individual; while in threshold signature scheme, the secret keys will be shared among n users. In this scheme, we use Shamir's secret sharing technique to distribute the secret keys among the users. We will discuss about the technique in the next subtopic.

For the message signature, t out of n users will collaborate to sign a message by using their pieces of secret keys and produce a partial signature. They send the partial signatures to TD and TD will verify the validity of the partial signatures. After TD verifies that all partial signatures are valid, then TD generates the group signature from the partial signatures. Then, TD produces the group signature along with the hash-function message. Any outsider can verify the signature, as long as he has access to the public keys.

3.1 SHAMIR'S SECRET SHARING

This technique was invested by (Shamir, 1979) by using the Lagrange interpolation technique. The idea of this technique is two point is needed to form a polynomial function of degree one, three point is needed to form a polynomial function of degree two, and so on. Generally, we can write this statement as follows:

Proposition 3.1 t points is needed to form a polynomial function of degree $t - 1$.

Shamir's secret sharing scheme can be defined as follows:

Definition 3.1 Let t, n be positive integers with $t \leq n$. A (t, n) threshold scheme is a method of sharing a message m among a set of n participants such that any subset consisting of t participants can reconstruct the message m, but no subset of smaller size can reconstruct m.

The secret message m is represented as a number mod p, and we want to split it among n participants so t out of n are needed to reconstruct the message. The first thing we do is construct a polynomial function of degree $t-1$ as follows:

$$P(x) = m + s_1 x + s_2 x^2 + \cdots + s_{t-1} x^{t-1} \pmod{p}$$

where $s_1, s_2, \ldots, s_{t-1}$ are any positive integers mod p. The prime p is made public but the polynomial function $P(x)$ is kept secret (notice that, if $P(x)$is known to anybody, then m is not a secret anymore). Then, set a pair $\left(x_i, P(x_i)\right)$ for each participants. x_i's represent the public individual identity, which is known to all, while $P(x_i)$'s are individual secret.

To reconstruct the secret message, (t, n) participants must collaborate to calculate the value of the secret m by using the formula:

$$m = \sum_{i=1}^{t} P(x_i) \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{-x_j}{x_i - x_j} \pmod{p}.$$

In our threshold signature scheme, we apply the Shamir's secret sharing technique to split the secret keys among the participants. Then, each user calculates $v_i = P(x_i) \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{-x_j}{x_i - x_j}$ by using the public individual identities and their $P(x_i)$ and sends it to the TD as partial signature. Lastly, TD calculates $\sum_{i=1}^{t} v_i$ to reconstruct the secret key, that will be used in the group signature generation. By this way, the secret key will be always kept secret from any outsider and also to all participants.

3.2 THE NEW THRESHOLD SIGNATURE SCHEME

We proposed a new threshold signature scheme based on factoring and discrete logarithm. The following parameters and notations will be used throughout this scheme unless otherwise specified:

- $h(m)$ is the one-way hash function of the message m.
- p is a 1024-bits prime number.
- $n = ab$ is a factor of $p - 1$, that is the product of two primes a and b.
- $\phi(n)$ is the Euler's phi function of n, i. e. $\phi(n) = (a - 1)(b - 1)$.
- g is a primitive root mod p, satisfying $g^n \equiv 1 \pmod{p}$.

All of the parameters above are made public, except the values of a and b. Notice that, the reason for keeping a and b secret is to make sure that computing e-th root modular n cannot be solve by any outsider and the participants due to the factoring problem. Refer to subtopic 2.5.2 for detail explanation.

As we stated earlier, a digital signature scheme consists of three steps; generating keys, signing message, and verifying signature. In the following subtopics, we will show our new scheme and give a numerical example to show how it works.

3.2.1 Generating Keys

Let u_i denote the group members and there are n group members so t from them $(u_1, u_2, \dots, u_t)$ can represent to sign the message.

1) The trusted dealer picks randomly $e \in \mathbb{Z}_n^*$ such that $\gcd(e, n) = 1$ and then calculates d such that $ed \equiv 1 \left(\mathrm{mod}\ \phi(n)\right)$.

2) Then, he constructs a secret (t, n) polynomial threshold function, $P(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1} \pmod{n}$, where a_i are random integers between 1 and $n - 1$, and $i = 0,1,2, \ldots, t - 1$.

3) From the threshold function $P(x)$, he sets the group secret key, $P(0) = a_0$, and calculates the corresponding public key, $V \equiv g^{a_0} \pmod{p}$.

4) He also sets a pair of key $\big(x_i, P(x_i)\big)$ for each group member u_i, where x_i is the public identity and $P(x_i)$ is the secret key for each member.

5) After each member receives their pair of key $\big(x_i, P(x_i)\big)$, each of them computes the corresponding individual public key $y_i \equiv g^{P(x_i)} \pmod{p}$.

The public and secret keys of the individual and group for the scheme are shown in Table 3.1.

TABLE 3.1. The public and secret keys of the scheme.

	Public key	**Secret key**
Individual	y_i	$P(x_i)$
Group	e, V	$d, P(0)$

3.2.2 Signing Message

Suppose that t out of n group members want to sign a message m. They can sign the message simultaneously. Here, the steps of signing the message are described.

1) Each member selects an integer r_i such that $0 < r_i < n$ and $\gcd(r_i, n) = 1$.

2) Computes $k_i \equiv g^{r_i} \pmod{p}$.

3) Each member broadcasts their k_i to other members via a secure channel. After all k_i are received, each of them calculates

$$K \equiv \prod_{i=1}^{t} k_i \pmod{p}.$$

4) By using the information of the public identity x_i for other members, each of them calculates

$$v_i \equiv \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{-x_j}{x_i - x_j} \pmod{n}.$$

5) Calculates

$$s_i \equiv K \cdot r_i + h(m) \cdot P(x_i) \cdot v_i \pmod{n}.$$

6) Each member sends K and v_i along with (k_i, s_i) as the partial signature for the hash-function message $h(m)$ to TD. TD checks the equality $g^{s_i} \equiv {k_i}^{K} \cdot {y_i}^{v_i \cdot h(m)} \pmod{p}$. If the equation holds, then (k_i, s_i) is a valid partial signature of u_i. Otherwise, the partial signature is invalid.

7) After TD shows that all partial signatures are valid, then he computes $S \equiv (\sum_{i=1}^{t} s_i)^d \pmod{n}$. TD produces (K, S) as the group signature for the hash-function message $h(m)$.

3.2.3 Verifying Signature

Any outsider can be the verifier, as long as he has access to the public key. After he receives the group signature from TD, he checks

$$g^{S^e} \equiv K^K \cdot V^{h(m)} \pmod{p}.$$

If the equation holds, then the group signature is valid.

Theorem 1. *Following the applied protocol, then the verification in the signature verification phase is correct.*

Proof:

The equation in signature verification phase is true for valid signature since,

$$
\begin{aligned}
g^{S^e} &\equiv g^{\left(\left(\sum_{i=1}^{t} s_i\right)^d\right)^e} && (\mathrm{mod}\ p)\\
&\equiv g^{\sum_{i=1}^{t} s_i} && (\mathrm{mod}\ p)\\
&\equiv g^{\sum_{i=1}^{t} K\cdot r_i + \sum_{i=1}^{t} h(m)\cdot P(x_i)\cdot v_i} && (\mathrm{mod}\ p)\\
&\equiv \left(g^{\sum_{i=1}^{t} r_i}\right)^K \left(g^{\sum_{i=1}^{t} P(x_i) v_i}\right)^{h(m)} && (\mathrm{mod}\ p)\\
&\equiv \left(\prod_{i=1}^{t} k_i\right)^K (g^{a_0})^{h(m)} && (\mathrm{mod}\ p)\\
&\equiv K^K \cdot V^{h(m)} && (\mathrm{mod}\ p)
\end{aligned}
$$

□

3.3 NUMERICAL EXAMPLE FOR THE PROPOSED SCHEME

In this subtopic, we present a numerical example to show how our threshold signature scheme works. We use software Maple 15 to generate all the parameters and keys and to perform all calculations. However, we only use small numbers to perform our scheme since it's required a processor with high memory to perform the scheme with actual size of numbers.

Suppose that there are 5 members $\{u_1, u_2, u_3, u_4, u_5\}$ and 3 from them can represent to sign a message. The example for this threshold scheme is shown as follows:

3.3.1 Generating Keys and Parameters

1) First, TD sets the following parameters:

   i. A prime $p = 14447$.
   ii. Two primes $a = 31$ and $b = 233$, and $n = ab = 7223$.
   iii. $\phi(n) = (a-1)(b-1) = (31-1)(233-1) = 6960$.
   iv. A primitive root mod p, $g = 8$, satisfies $8^{7223} \equiv 1 \bmod 14447$.

2) Then, TD picks $e = 19$ and $d \equiv e^{-1} \equiv 1099 \bmod 6960$.

3) Constructs a $(3,5)$ threshold function,

$$P(x) = 345 + 123x + 789x^2 \pmod{7223}.$$

   Then, the group secret key is $P(0) = a_0 = 345$ and the corresponding public key, $V \equiv 8^{345} \equiv 4130 \bmod 14447$.

4) Sets a pair of keys for each group members as follows:

   i. $(x_1, P(x_1)) = (1, 1257)$.
   ii. $(x_2, P(x_2)) = (2, 3747)$.
   iii. $(x_3, P(x_3)) = (3, 592)$.
   iv. $(x_4, P(x_4)) = (4, 6238)$.
   v. $(x_5, P(x_5)) = (5, 6239)$.

5) After each member receive their pair of keys $\big(x_i, P(x_i)\big)$, each of them compute the corresponding individual public key y_i as follows:

   i. $y_1 \equiv 8^{1257} \equiv 416 \bmod 14447$.
   ii. $y_2 \equiv 8^{3747} \equiv 8246 \bmod 14447$.
   iii. $y_3 \equiv 8^{592} \equiv 7468 \bmod 14447$.
   iv. $y_4 \equiv 8^{6238} \equiv 6284 \bmod 14447$.
   v. $y_5 \equiv 8^{6239} \equiv 6931 \bmod 14447$.

The public and secret keys of the individual and group for the scheme are shown in Table 3.2.

TABLE 3.2. The public and secret keys of the scheme.

	Public key	**Secret key**
Individual	y_i	$P(x_i)$
Group	$e = 19$	$d = 1099$
	$V = 4130$	$P(0) = 345$

3.3.2 Signing Message

Suppose that u_1, u_3 and u_4 want to sign a hash functioned message m, $h(m) = 805$. They can sign the message simultaneously. Here, the steps of signing the message are described:

1) Each user selects randomly $r_1 = 186$, $r_2 = 407$, and $r_3 = 211$, and then calculates $k_1 = 8^{186} = 9788 \bmod 14447$, $k_2 = 8^{407} = 13107 \bmod 14447$, and $k_3 = 8^{211} = 10188 \bmod 14447$.

2) Broadcasts their k_i to other members through a secure channel. After all k_i's are received, they calculate $K = 186 \times 407 \times 211 = 9187 \bmod 14447$.

3) Then, they calculate $v_1 = \left(\frac{-3}{1-3}\right) \cdot \left(\frac{-4}{1-4}\right) = 2 \bmod 7223$, $v_2 = \left(\frac{-1}{3-1}\right) \cdot \left(\frac{-4}{3-4}\right) = 7221 \bmod 7223$, and $v_3 = \left(\frac{-1}{4-1}\right) \cdot \left(\frac{-3}{4-3}\right) = 1 \bmod 7223$.

4) After that, they compute $s_1 \equiv 9187 \cdot 186 + 805 \cdot 1257 \cdot 2 = 5484 \bmod 7223$, $s_2 \equiv 9187 \cdot 407 + 805 \cdot 592 \cdot 7221 = 5134 \bmod 7223$ and $s_3 \equiv 9187 \cdot 211 + 805 \cdot 6238 \cdot 1 = 4298 \bmod 7223$. Finally, they send (s_i, k_i) to TD as the partial signature for the message.

5) After TD receives $(s_i, k_i) = \{(5484, 9788), (5134, 13107), (4298, 10188)\}$, he checks the validity for each partial signature by showing the equality $g^{s_i} \equiv {k_i}^K \cdot {y_i}^{v_i \cdot h(m)} \pmod{p}$ holds. Otherwise, the partial signature is invalid. For example, he checks the partial signature for u_1:

$$8^{5484} = 9788^{9187} \cdot 416^{2 \cdot 805} = 6823 \bmod 14447.$$

6) After TD checks all partial signature are valid, then he calculate $S = (5484 + 5134 + 4298)^{1099} = 1307 \bmod 7223$. Finally, he produces $(S, K) = (1307,9187)$ as the group signature for the message.

3.3.3 Verifying Signature

After receive the group signature (S, K) for the hash-functioned message $h(m)$, with the knowledge of the group public key, verifier performs the following equality:

$$8^{1307^{19}} = 9187^{9187} \cdot 4130^{805} = 2455 \bmod 14447$$

Since the equality holds, then the verifier declares that the group signature is valid for the hash-functioned message $h(m)$.

CHAPTER 4

SECURITY ANALYSIS AND PERFORMANCE EVALUATION

In previous chapter, we already show our new threshold scheme and how it works. In this chapter, we will evaluate the scheme in terms of the security and efficiency.

4.1 SECURITY ANALYSIS

It is important to make sure that a digital signature scheme has being developed is secure against some security attacks. To do this, we will test our threshold signature scheme with some security attack and show that the scheme is heuristically secure. The attacks in general we considered are:

1) **The key-only attack**: Adversary (Adv) searches for the secret key by using all information from the system.
2) **The feed attack**: Adv fixes one of the components of signature and tries to find the rest of the component.

3) **The chosen-message attack**: Adv has an access to the signature oracle and get a number of valid pair of signatures-messages.
4) **The FAC attack**: It is assumed that FAC is solvable. Then Adv tries to find any secret information of the system.
5) **The DLP attack**: It is assumed that DLP is solvable. Then Adv tries to find any secret information of the system.
6) **The impersonate-member attack**: Adv tries to impersonate a member of the group and generates a valid partial signature to satisfy the partial signature verification equation.

Next, we will show one-by-one how these attacks cannot successfully break our scheme.

Attack 1 (The key-only attack)

i. Adv wishes to obtain group secret keys d and $P(0)$ by using all information from the system. In this case, Adv needs to solve $ed \equiv 1 \left(\mathrm{mod}\, \phi(n)\right)$ and $V \equiv g^{a_0} \pmod{p}$, which are clearly infeasible due to the difficulty of solving FAC and DLP.
ii. Adv also cannot derive the individual secret key, $P(x_i)$ from the equation $y_i \equiv g^{P(x_i)} \pmod{p}$ due to the difficulty of solving DLP.

Attack 2 (The feed attack)

Adv might try to derive their own group signature (K, S) from the verifying equation $g^{S^e} \equiv K^K \cdot V^{h(m)} \pmod{p}$ for a given message m by letting one integer fixed and finding the other one. We can divide this attack into two cases:

i. Adv selects K and tries to figure out the value of S. In this case, Adv calculates $\lambda \equiv K^K \cdot V^{h(m)} \pmod{p}$. Then, he has to solve $\lambda \equiv g^{S^e} \pmod{p}$. Unfortunately, ho cannot find S from this equation due to the difficulty of solving FAC and DLP simultaneously.

ii. Adv also might try to fix S and find K. In this case, he calculates $\gamma \equiv g^{S^e} \cdot V^{-h(m)} \pmod{p}$ and tries to solve $\gamma \equiv K^K \pmod{p}$. This is worse scenario because even FAC and DLP are solvable, the value of K still hard to find except by try and error, but it is time consuming.

Attack 3 (The chosen-message attack)

Adv may also try collecting t pairs of message-signature (k_{ij}, s_{ij}) and m_j , where $j = 1, 2, \dots, t$ and attempts to find the individual secret key $P(x_i)$. In this case, Adv has t equations as follows:

$$s_{i1} \equiv K_1 \cdot r_{i1} + h(m_1) \cdot P(x_i) \cdot v_i \pmod{n}$$
$$s_{i2} \equiv K_2 \cdot r_{i2} + h(m_2) \cdot P(x_i) \cdot v_i \pmod{n}$$
$$\vdots$$
$$s_{it} \equiv K_1 \cdot r_{it} + h(m_t) \cdot P(x_i) \cdot v_i \pmod{n}$$

In the above equations, there are $(t + 1)$ unknowns, i.e., $P(x_i)$ and r_{ij}. Hence, $P(x_i)$ stays hard to detect because Adv can generate infinite number of solutions of the above system of linear equations but cannot figure out which one is correct.

Attack 4 (The FAC attack)

It is assumed that Adv is able to solve FAC. In this case, he knows the prime factorization, a and b, and find d. Then he tries to solve the equation $\lambda \equiv g^{S^e} \pmod{p}$. However, he still cannot find S from this equation because he does not know $S^e \pmod{n}$ since DLP is not solvable.

Attack 5 (The DLP attack)

Supposed that DLP is solvable, then from the equation $\lambda \equiv g^{S^e} \pmod{p}$ Adv can find $S^e \pmod{n}$. However, he still cannot find S due to the difficulty of solving FAC.

Attack 6 (The impersonate-member attack)

Adv might try to impersonate member u_i by randomly selects an integer r_i and broadcasting $k_i \equiv g^{r_i} \pmod{p}$. Since the group signature is determined by all t members, without knowing the individual secret key $P(x_i)$, Adv cannot generate a valid partial signature (k_i, s_i) to satisfy the verification equation.

4.2 PERFORMANCE EVALUATION

From the security analysis, we already show that how our scheme is secure against some security attacks. Another evaluation for a signature scheme is efficiency analysis. In efficiency analysis, we investigate the performance of our scheme in terms of number of

keys, computational complexity, and communication cost. Then, we compare our scheme with the threshold signature scheme based on DLP, which is proposed by Harn (1994). The following notations are used to analyze the performance of the scheme:

- SK and PK are the number of secret and public keys respectively.
- T_{exp} is the time complexity for executing the modular exponentiation computation.
- T_{mul} is the time complexity for executing the modular multiplication computation.
- T_{inv} is the time complexity for executing the modular inverse computation.
- T_{sqrt} is the complexity for executing the modular square root computation.
- T_h is the time complexity for performing hash function.
- $|\eta|$ denotes the bit length of η.

The efficiency analysis of the scheme and the comparison are shown in the Table 4.1.

TABLE 4.1. The efficiency analysis and comparison

		New threshold signature scheme	**Single-problem threshold signature scheme (Harn, 1994)**
No of keys	SK PK	$t+2$ $t+2$	$t+1$ $t+1$
Computational complexity	Sign	$(4t+1)T_{exp} + (2t^2+3t)T_{mul} + (t^2-t)T_{inv} + T_{sqrt} + T_h$	$(4t)T_{exp} + (5t^2-2t)T_{mul} + 2(t^2-t)T_{inv} + T_h$
	Verify	$4T_{exp} + 2T_{mul}$	$3T_{exp} + T_{mul}$
Size of parameters / Communication cost		$(2t+1)\|n\| + (3t+1)\|p\|$	$(t+1)\|n\| + (3t+1)\|p\|$

From Table 4.1, it is shown that our threshold signature scheme is less efficient compare with the threshold signature scheme with single problem. This disadvantage of our scheme due to the element of both factoring and discrete logarithm in the signing and verifying steps. Even though our new scheme runs no faster than the scheme with single problem, it will be a choice for everyone who prefers a more secure system rather than a more efficient system.

CHAPTER 5

SUMMARY AND CONCLUSION

5.1 SUMMARY

For integrity reason, nowadays most of the online documents, transactions, and messages from an organization or society need to be signed by more than one person. That's why threshold signature schemes based on various problems in number theory have being developed. From the past decades until now, many threshold signature schemes were being designed based on single number theory problem, such as factoring, discrete logarithm, residuosity, elliptic curve, etc. Since it is understood that one day these problems could be solved, all signatures that depend on the problem will be no longer secure. One of the strategy to overcome this situation is by designing a signature scheme based on multiple hard number theory problems.

In this research, we developed a new threshold signature scheme based on two number theoretical problems; namely factoring and discrete logarithm. In this report, we start with research introduction, where we reviewed some literatures and discussed the properties of digital signatures. Then we defined our problem statement, research objectives, and significance of the research.

In Chapter 2, we discussed some important topics in number theory that related to our scheme. We also reviewed some cryptographic functions that we used throughout the scheme.

In Chapter 3, we presented our new threshold signature scheme. Generally, a digital signature scheme consists of three steps: generating keys, signing message, and verifying signature. We explained all of the steps and we also showed a numerical example to show how our scheme works.

For any digital signature scheme, it is important to evaluate the scheme in terms of the security and performance. That's why in Chapter 4, we evaluated our scheme by showing it is heuristically secure against some cryptography attacks. We also showed that our scheme is significantly efficient, compare with a threshold signature scheme based on single problem.

5.2 FUTURE WORKS

In the past, there were many threshold signature schemes have being developed based on various single problem in number theory. To the best of our knowledge, this is the first threshold signature scheme designed based on factoring and discrete logarithm. As we already stated in this report, the scheme based on multiple hard number theory problems are more secure than the scheme based on single problem. Here, we list some future research that can be done related to this area:

i. Developing threshold ordinary signature schemes based on others number theoretical problems such that residuosity and elliptic curve.
ii. Developing threshold function-based signature schemes based on two hard problems.
iii. Developing threshold cryptosystems and authenticated encryptions based on hybrid problems.

REFERENCES

Desmedt, Y. (1988). Society and group oriented cryptography: a new concept. *Advances in Cryptology, Proceedings of Crypto '87*, 120-127.

Desmedt, Y. and Frankel, Y. (1989). Threshold cryptosystem. *Advances in Cryptology, Proceedings of Crypto '89*, 307-315.

Desmedt, Y. and Frankel, Y. (1991). Shared generation of authenticators. *Advances in Cryptology, Proceedings of Crypto '91*, 457-469.

ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transaction on Information Theory*, IT-31(4), 469-472.

Harn, L. (1994). Group oriented (*t*,*n*) threshold digital signature scheme and digital multisignature. *IEE Proceedings-Computers and Digital Techniques*, 141(5), 307-313.

He, W. (2001). Electronic signature scheme based on residuosity and discrete logarithms. *Electronic Letters*, 37(4), 220-222.

Ismail, E., S. 2004. Digital signature schemes and their respective applications. Ph.D. Thesis. Universiti Sains Malaysia.

Ismail, E., S., Tahat, N., M., F., and Ahmad, R., R. (2009). A new signature scheme based on factoring and discrete logarithms. *Journal of Discrete Mathematical Sciences & Cryptography*, 12(3), 313-318.

Laih, C. S. and Kuo, W. C. (1997). New signature scheme based on residuosity and discrete logarithms. *IEICE Transactions on Fundamentals on Cryptography and Information Security E80-A1*, 46-53.

Lee, N. Y. and Hwang, T. (1996). Modified Harn signature scheme based on residuosity and discrete logarithms. *IEE Proceedings-Computers and Electronic Techniques*, 143(3), 196-198.

Mollin, R., A. 2001. An introduction to cryptography. Chapman & Hall. USA.

Rivest, R., Shamir, A. and Adleman, L. (1978). A method for obtaining digital signature and public-key cryptosystem. *Communication of the ACM*, 21(2), 120-126.

Shamir, A. (1979). How to share a secret. *Communications of the ACM*, 22(11), 612-613.

Tahat, N., M., F. 2010. The development of digital signature schemes using multiple cryptographic assumptions in algebra and number theory. Ph.D. Thesis. Universiti Kebangsaan Malaysia.

Trappe, W., and Washington, L., C. Introduction to cryptography with coding theory. Prentice-Hall. USA.

Wang, C. T. and Chang, C. C. (2003). Signature scheme based on two hard problems simultaneously. *Proceeding of the 17th International Conference on Advanced Information Networking and Application*, 557-560.