ENHANCE KEY GENERATION ALGORITHM BASED USER STRING IDENTITY AS PUBLIC KEY

NORHIDAYAH MUHAMMAD

Thesis submitted in fulfillment of the requirements

for the award of the degree of

Master of Computer Science

Faculty of Computer System &Software Engineering

UNIVERSITI MALAYSIA PAHANG

JUN 2015

ABSTRACT

This study aims to enhance a previous algorithm called the Tripathi algorithm. The Tripathi algorithm proposes an RSA based approach to generate cryptographic keys using a user's identity, such as an E-mail address. This algorithm uses a user's identity to replace the numbers that are used as a public key in the RSA algorithm. The main advantage of the Tripathi algorithm is that users can easily recall the public key, since it is their own identity. However, this algorithm is unable to use all of users' E-mail addresses as public keys, only certain E-mails can be used as public keys. This is due to two main reasons: i) this algorithm uses the same modulo value for every E-mail. Consequently, if the E-mail is not relative prime to modulo value, the new E-mail should be entered, and ii) once the E-mail converted to decimal value, it is composed of odd and even numbers. If the decimal value is in even numbers, then it can't be used as a public key. Only odd decimal values can be used in a public key using the Tripathi algorithm. Therefore, the Tripathi algorithm needs to be improved so that all E-mail addresses can be used as public keys. The proposed algorithm enables the algorithm to be applicable to all E-mail address domains, such as yahoo, gmail, edu, company, among others. In this study, two experiments were conducted. In the first experiment, an algorithm called the LB-RSA was developed. A looping process was added to this algorithm in order to produce a new modulo value to make the E-mail entered relative prime to the new modulo value, which helps to produce more E-mail addresses that can be used as public keys. This algorithm has shown promising results, and 50% of the total number of E-mails can be used as public keys. This number is greater than the amount generated by the Tripathi algorithm, which is only 10%. Since the result produced by this algorithm did not reach 100%, the second experiment was conducted to further enhance the LB-RSA algorithm. In this experiment, an algorithm called the CLB-RSA was developed. Classifications of decimal values to determine whether the E-mail entered contains odd or even numbers were added to this algorithm. This algorithm achieved 100%, hence, all E-mails considered can be used as public keys.

ABSTRAK

Tujuan kajian ini adalah untuk membuat penambahbaikan pada algorithm yang lepas yang dikenali sebagai algoritma Tripathi. Algoritma ini telah mencadangkan satu algoritma yang berasaskan algoritma RSA. Algoritma ini menjana kunci kriptografi menggunakan identiti pengguna seperti alamat e-mel. Algoritma ini menggunakan identiti pengguna untuk manggantikan nombor yang digunakan sebagai kunci awam dalam algoritma RSA. Kelebihan utama algoritma adalah memudahkan pengguna mengingat kunci awam kerana ia adalah identiti mereka sendiri. Walau bagaimanapun, algoritma ini tidak dapat menggunakan semua alamat E-mel pengguna sebagai kunci awam, hanya E-mel tertentu sahaja yang boleh dijadikan kunci awam. Ini kerana, terdapat dua masalah yang timbul didalam algoritma ini iaitu: i) algoritma ini menggunakan nilai modulo yang sama bagi setiap E-mel pengguna, jika E-mail yang dimasukkan oleh pengguna tidak dapat digunakan sebagai kunci awam, maka pengguna perlu memasukkan E-mel yang baru. ii) E-mail yang dimasukkan terdiri daripada dua kategori nombor, iaitu nombor ganjil dan juga nombor genap. Jika e-mail yang dimasukkan adalah nombor genap, maka ia tidak boleh menjadi kunci awam. Oleh itu algoritma Tripathi perlu ditambahbaik agar semua alamat e-mel yang dimasukkan oleh pengguna boleh digunakan sebagai kunci awam. Algoritma yang dicadangkan berupaya menjadikan semua E-mail sebagai kunci awam walaupun daripada domain yang pelbagai seperti yahoo,gmail,syarikat dan lain-lain. Dalam kajian ini, dua eksperimen telah dijalankan. Dalam ekperimen pertama, algoritma dipanggil LB-RSA telah dibangunkan dan diuji. Proses gelung ditambah pada algoritma ini untuk menghasilkan nilai modulo yang baru, supaya nilai modulo dan nilai decimal adalah relative prime. Apabila nilai modulo dan decimal adalah relative prime, maka alamat e-mel yang dimasukkan boleh dijadikan sebagai kunci awam. Algoritma ini telah menunjukkan hasil yang baik, iaitu 50 % daripada bilangan E-mail yang boleh digunakan sebagai kunci awam. Jumlah ini adalah lebih besar daripada jumlah yang dihasilkan oleh algoritma Tripathi iaitu 10 %. Disebabkan keputusan yang dihasilkan oleh algoritma ini tidak mencapai keputusan 100%, maka ekperimen kedua dilakukan untuk meningkatkan algoritma LB- RSA. Dalam eksperimen ini, satu algoritma dipanggil CLB-RSA telah dibangunkan. Proses klasifikasi pada nilai decimal ditambah dalam algoritma ini bagi menentukan samaada nilai decimal yang dimasukkan adalah nombor genap atau ganjil. Ini adalah masalah yang dihadapi oleh algoritma LB-RSA, iaitu tidak boleh menjadikan email yang berjenis genap sebagai kunci awam, hanya yang berjenis ganjil sahaja yang boleh dijadikan sebagai kunci awam. Oleh itu bagi setiap nilai decimal yang berjenis genap akan ditukarkan kedalam bentuk ganjil terlebih dahulu. Huruf "C" menandakan untuk proses klasifikasi dan "LB" menandakan untuk proses gelung. Algoritma ini mencapai keputusan penuh iaitu 100 % kerana semua E-mel yang dimasukkan boleh dijadikan kunci awam.

TABLE OF CONTENTS

CHAPTER 5 RESULTS AND DISCUSSION

CHAPTER 6 CONCLUSION

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

RSA	Rivest, Shamir, Adleman
IBE	Identity Based Encryption
PKC	Public Key Cryptosystem
LB-RSA	Loop Based RSA algorithm
CLB-RSA	Classification and Loop Based RSA algorithm
NIST	National Institute for Standards and Technology
GCD	Greatest Command Division
CA	Certificate Authority
IEEE	Institute of Electrical and Electronics Engineer

CHAPTER 1

INTRODUCTION

1.1 Research Background

Information security, also known as computer security, is an approach to protect information from unauthorized access, disruption, inspection, modification or manipulation of information (Diffie & Hellman, 1976). Computer security or cryptography ensures a secure communication from the manipulation of third parties (Rivest, 1992b). Security and confidentiality of the cryptography process are also known as encryption. In encryption technology, only legitimate recipients can read a message sent across the network. In a secure communication, a message should be read by the intended recipients, hence, shouldn't be intercepted by any third parties. Therefore, an encryption algorithm is used to transmit data over public networks.

A cryptography, or encryption, consists of three important elements, namely, encryption, decryption, and key generation (Wenbo, 2004). Encryption is the process used to convert or transform plain text to unreadable code known as cipher text. Decryption is the reverse transformation (encryption transformation). Key generation is the main problem faced in cryptography, because hackers attempt to crack keys used. The key generation process involves determining the public and private keys used during the encryption and decryption processes. The level a cryptography algorithm provides security depends on the complexity of the cryptography keys. Cryptography can be divided into symmetric and asymmetric techniques. In symmetric cryptosystems, the same key is used for both encryption and decryption, so a secure medium is needed to ensure the key is transmitted securely, because the keys are shared between sender and receiver (Tan *et al*., 2009). These cryptosystems are considered fast and easy to use. However, problems occur if the key used is acquired by third parties. When this

problem occurs, the third party will be able to decrypt the data, since the same key is used in both the encryption and decryption processes.

In asymmetric or public-key cryptosystems, two different keys are used: public key and private keys. Using a receiver's public key, the sender encrypts the message and sends it to the receiver, and the receiver decrypts the message with his private key. The well-designed implementation of the public-key cryptosystem was carried out by Rivest, Shamir and Adleman, called the RSA Public-Key Cryptosystem (Rivest *et al*., 1983). RSA is the most popular and widely used cryptosystem, because RSA algorithm security depends on the difficulty of discovering the private key (Anane *et al*., 2010). The RSA algorithm is commonly known as a mechanism for providing a relatively good security. Although the RSA algorithm is more secure than symmetric key algorithms, there are several disadvantages in implementing the RSA algorithm. The RSA algorithm uses a key consisting of a row of numbers. Moreover, it requires a large storage space, and is only suitable to be used on devices with a large amount of memory.

In 1984, Shamir (Shamir, 1985) proposed a public key encryption scheme in which the public key can be an arbitrary string. Shamir's original motivation for identity based encryption was to simplify the certificate management. Several proposals for identity based encryption (IBE) schemes have been proposed (Desmedt & Quisquater, 1987; Maurer & Yacobi, 1991; Boneh & Franklin, 2003; Hühnlein *et al*., 2003). In this algorithm, a user's identity is chosen as the public key to replace the numbers that are used in the RSA public key algorithm. However, IBE requires a Certification Authority (CA) as a public key authentication agent. Identity based encryption needs a CA to provide a trusted public key to the various participants on demand, and to set up the hierarchical infrastructure for numerous CA's extra overhead that is required.

Therefore, the Tripathi algorithm has produced a similar algorithm to IBE that uses a string as the public key. The Tripathi algorithm proposes an RSA based algorithm to generate the cryptographic keys using an identity, such as a person's E-mail address. This algorithm helps the sender to recall the public key for each receiver before encrypting the messages to be sent to a specific receiver. In this algorithm, a user's identity will be chosen as the public key to replace the numbers that are used in

the RSA public key algorithm. The Tripathi uses a hash function to create a public key. A string as a public key is also used in IBE, but the Tripathi algorithm does not use CA to authenticate the public key, but uses a hash function instead. The advantage of using this algorithm is that users can easily recall the public, key since it is their own. The E-mail address is public information that is well-known by its owner, hence, making it suitable to be used as a public key in an encryption algorithm.

The Tripathi algorithm enhances the RSA algorithm by using a string as the public key instead of numbers. However, there are also some problems with using the Tripathi algorithm. The main problem is that the Tripathi algorithm is unable to use all users' E-mail addresses as public keys; only a few E-mails can be used as public keys. Therefore, the Tripathi algorithm needs to be improved in order so that all E-mail addresses can be used as public keys. The algorithm should be applicable to all E-mail domains, such as yahoo, gmail, edu, companies, etc. Therefore, this research aims to enhance the Tripathi to ensure all E-mails can be used as public keys.

1.2 Problem Description

There are three main problems associated with the Tripathi algorithm. The first problem is that it is cannot produce satisfactory results because it is unable to use all users' E-mail addresses as public key keys. Therefore, only certain E-mails can be used as a public key. If a particular E-mail cannot be used as a public key, the user must provide another E-mail.

The second problem is that the Tripathi algorithm uses the same modulo value to test every E-mail, and when the same modulo value is used for every E-mail, the E-mail probably cannot be used as a public key. This is because, if the E-mail is not a relative prime to modulo value, the E-mail cannot be used as a public key, and consequently the user must use another E-mail.

The third problem is that the Tripathi algorithm cannot use an even decimal value as a public key. This is because a decimal value can be an odd number and an even number. Therefore, the proposed algorithm ensures all E-mails can used as a public key.

1.3 Objectives of the Study

The objective of this study is to solve the problems mentioned in the problem description. There are three research objectives to achieve in this study. The first objective is to enhance previous algorithm, this is because previous cannot make all E-mails as public keys.

The second objective is to implement a loop based classification process as the proposed algorithm. The implementation of this process is required because the Tripathi algorithm cannot use all E-mails addresses as a public key; only a few E-mail addresses can be used as a public key. Therefore, this creates problems for the user, who must reenter a new E-mail if the primary E-mail address cannot be used as a public key.

The third objective is to evaluate the new algorithm using the Maple software application. Each phase of the improved algorithm is tested in Maple to obtain results for each algorithm.

1.4 Research Question

Some questions arise regarding this topic and areas relevant to the research. The research questions are as follows:

1. Why do previous algorithms need to be improved?

Previous algorithms cannot use all E-mails input by a user as a public key. So the proposed algorithm enhances the Tripathi algorithm to ensure all E-mails can be used as public keys. Therefore, the user doesn't need to input another E-mail address.

2. What is the problem if the Tripathi uses the same modulo for every E-mail address?

When the same modulo is used for different E-mails, a few E-mails cannot be used as public keys because there are E-mails that are not relative prime to the modulo value. So a new modulo value will be created and probability to E-mail and modulo is relative prime is high.

3. Why is the Tripathi algorithm unable to use an even number of the decimal value a public key?

The proposed algorithm will convert an even number of the decimal value to an odd number before testing whether it can be used as a public key or not. When the decimal value is an odd number, it can be used as a public key.

1.5 Scope of the Study

This study focuses on three main points and limitations.

1. Enhance key generation process

   This study focused on the key generation process in the RSA algorithm. Modifications made are in the key generation process. This is because the key generation process is the phase that produces cryptography keys, which are the public and private keys in the RSA algorithm. This study aims to produce algorithms that are able to use all users' E-mail addresses from various domains as a public key. Therefore, only this phase is improved, and the phase is otherwise similar to the Tripathi algorithm.

2. Using user E-mail addressess as a data set

   The E-mail address is used to test the proposed algorithm. The E-mail address is one of the user's identities, and the test does not involve a user's other identities. There are many user identities that can be used to test this algorithm, such as username, nickname and so on. However, for this study, only the E-mail address is used as a public key from various domains.

3. Using CRC32 hash function

   CRC32 hash function will be used to convert an E-mail address into a public key. Another hash function can also be used, but the Tripathi has done the comparison between the CRC32 and the Adler32 hash functions. The Tripathi chooses the CRC32 because the results for CRC32 are better than for the Adler32 hash function (Tripathi *et al*., 2011).

1.6 Contribution of the Study

The main contribution of this study is to enhance the previous algorithm (Tripathi algorithm) and make all user E-mails applicable as public keys. The previous algorithm can only use a few E-mails as a public key. More specifically, this enhancement ensures that E-mails of different domains (as shown in A*ppendix A*) can be used as public keys. A user does not need to provide another E-mail if the first Email cannot be used as a public key. This is because the proposed algorithm ensures that all types of domains can be public keys. The Tripathi algorithm uses a hash function to create public keys, replacing certificate authority or key generation center (KGC) as public key authentication agents which use IBE to create public keys. This is because distributing the CA requires great costs, along with other problems, which are discussed in the literature review in Chapter 2.

1.7 Research Interests

This study contributes indirectly to the RSA algorithm and cryptography. The algorithm that is developed does not only contribute to the algorithm itself, it also helps the user to select the desired E-mail as a public key without having to worry about whether the email is valid to be used as a public key or not. This algorithm is also of benefit in terms of algorithm performance, because the number of E-mails that are used as a public key are greater compared to the Tripathi algorithm, this result and performance will be discuss in chapter 5.

1.8 Research Methodology

There are three main phases in the research methodology, which are associated with the three research objectives, respectively, as shown in Figure 1.1. Each phase consists of a research phase, methodology phase and output phase for every stated research objective. Objective 1 involves determining the algorithm analysis and understanding the algorithm framework (Tripathi) in the research phase. The methodology phase involves conducting the literature review and exploring the existing research works related with the algorithm. The output phase fof Objective 1 provides

results that can be used for experimentation with new data on the existing algorithm (Tripathi algorithm).

The next objective is to implement two processes, which are the looping process and the classification process. The research phase involves studying the looping process and the classification process. The methodology phase involves implementing the looping process and the classification process in the proposed algorithm. The output process attempts to analyze the results based on the two process implementation of looing and classification.

The third objective involves evaluating the result. In the research phase, after the implementation of the processes into the proposed algorithm, the results for every enhancement process are evaluated and analyzed. Complexity analysis for both enhancement processes was conducted. In the methodology phase, performance testing was conducted for both enhancement processes. The output phase involves the final enhancement algorithm, which was analyzed with satisfying results.

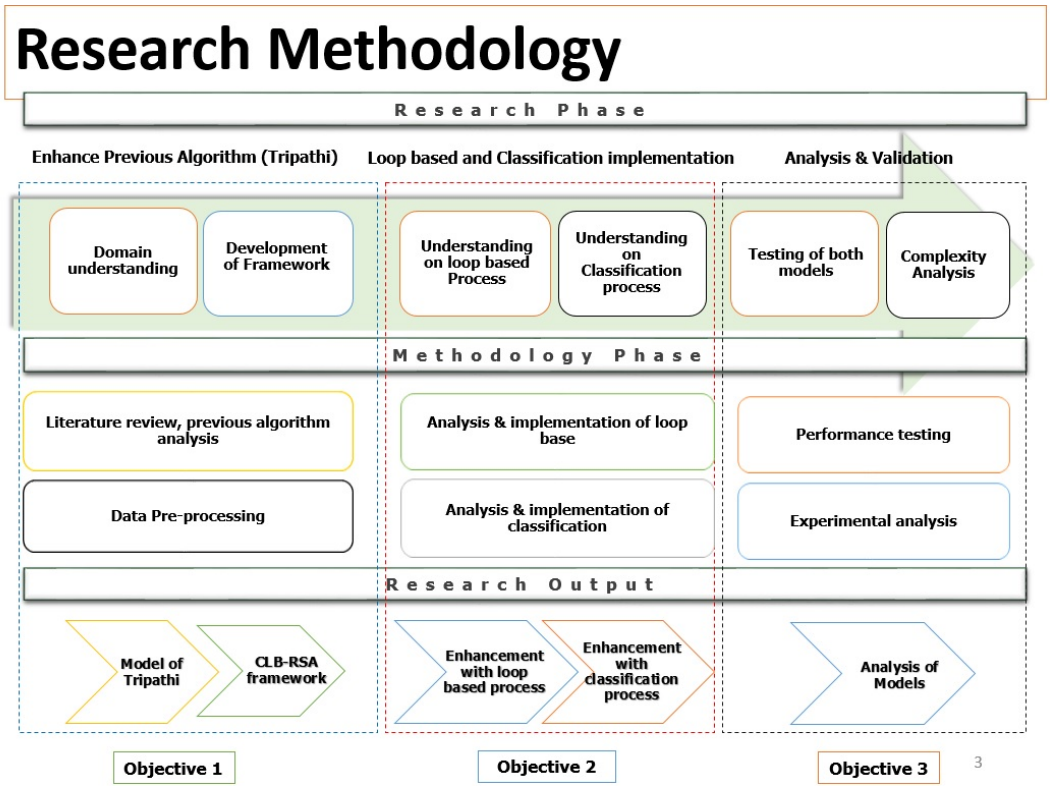

Figure 1.1: Research Methodology

1.9 Thesis Organization

This thesis consists of six chapters. This thesis begins with the introduction. The introduction chapter explains in detail the introduction of the research, objectives, scope, and all the related topics. Chapter 2 presents the literature review, also known as the related work. This chapter explains previous research findings related to this research, with the aim of obtaining knowledge from the previous studies. Chapter 3 explains the design phase, and how the proposed algorithm was developed to satisfy all of the research objectives. Included is an explanation on two additional techniques in the proposed algorithm, as well as on the data used to test the algorithm and how E-mails are converted to decimal values.

The next phase is implementation. In this phase, the algorithm that was developed is implemented in the Maple software application to test the content, and whether it is able to produce the results stated in the research objectives. Chapter 4 discusses the experiments carried out. This chapter describes in details the first and second experiments. Chapter 5 explains the results produced by the first experiment and the second experiment, and provides a comparison of the results of the proposed algorithm and the previous algorithm. Finally, the conclusion is provided in Chapter 6, as well as scenarios in which the proposed algorithm can be used.

1.10 Summary

This chapter provides an introduction to this study, as well as an explanation of the outline of this research. The next chapter explains in detail the processes involved to satisfy the research objectives. The proposed algorithm produces better results than the Tripathi algorithm because all E-mails can be used as public keys. The problems inherent in the Tripathi algorithm are identified, and these problems are solved by adding two processes to make sure all E-mails can be used as public keys, which is discussed in Chapter 3. In conclusion, this chapter provides a basic guide to the direction of the overall study. This research is driven towards achieving the objectives that have been stated and discussed in Section 1.8 (Research Methodology).

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

With the widespread popularity of electronic communication and commerce, data security issues have become an increasing concern. Cryptography is one of methodologies in computer security applied to increase data security. In the 90's, the field of cryptography has faced a new problem beyond privacy, which was the main goal at that time. Diffie and Hellman mentioned that they were at the brink of a revolution in cryptography (Diffie & Hellman, 1976) as an introduction to their paper on the concept of Public-Key Cryptosystems (PKC). The PKC was born after two years in 1978. An elegant implementation of the public-key cryptosystem came from Rivest, Shamir and Adleman, called the RSA Public-Key Cryptosystem (Rivest *et al*., 1983).

Today, because of the high-security provided, RSA is still known as the most widely used public-key cryptosystem in the world. Although it provides high security, currently available RSA hardware needs to be improved on the speed and area issues. The security of RSA increases as the number of bits in the algorithm increases. However, using a greater number of bits ends up with slower architectures and increased area. The challenge is to provide fast architectures and efficiently used resource as the number of bits increases.

This chapter aims to comprehensively investigate issues associated with the RSA algorithm discussed. Among the issues to be discussed is the component involved in RSA. This chapter also discusses several methods in public key cryptography, such as Identity Based Encryption (IBE). IBE is competitive algorithm with RSA because this algorithm is able to use strings as public keys. Therefore the Tripathi algorithm was

developed with a new theory based on RSA that is able to use strings as public keys, as in the IBE algorithm. This chapter also provides an example of the RSA step calculation which was given, including key generation, encryption and decryption, as well as an RSA simulation in the Maple software application. Maple is a commercial computer algebraic system which helps to facilitate the simulation process of an RSA algorithm. Users can enter mathematics in traditional mathematical notation. In Maple, user interfaces can also be created. There is support for numeric computations, to an arbitrary precision, as well as symbolic computation and visualization (Felder, 1998).

2.2 Information security

The terms information security, computer security and information assurance are frequently used interchangeably. These fields are interrelated, and share the common goals of protecting the confidentiality, integrity and availability of information. Today, researchers focus more on technical security, and less on the soft issues of security awareness and harm caused by end users (Katz, 2005).

This awareness must be inculcated at all levels to ensure data security is assured. Information Security (IS) is a member of security awareness aimed at providing real awareness to the user (Siponen, 2001). Security awareness is very important for the organization of information security (Straub & Welke, 1998). Recent studies show that human error is a threat to their information assets, and is rated among the main threats in the organization (Whitman & Mattord, 2011). IT staff should be made aware of the urgent need to ensure effective information security within the organization (Pfleeger & Pfleeger, 2003). Thus, a single misuse of the organization's information systems is more expensive than designing a security system (Czernowalow, 2005). IS currently being used in critical business operations (Conner & Coviello, 2004).

2.3 Cryptography/Encryption

The word of cryptography is derived from two Greek words. ‘Crypto’ means hidden, and ‘graph’ means write. Cryptography is also known as an encryption. Any encryption algorithm depends on the length of the key and the computational effort required breaking that key (Alaa H Al-Hamami & Bilal S O Al-Kubaysee, 2011).

Cryptography is used to hide information from invaders. The term refers to the science and art of transforming messages to make them secure and immune to attacks (Forouzan, 2006). Cryptosystem is the process of changing a message (plain text) to non-readable code known as cipher text so that only the intended recipient can read it. The recipient of the encrypted text uses a "private key" to decrypt the message, returning it to its original plain text form.

A security system with a wireless communication should be an enhanced so that the process of data transmission and data storage will be more secure. Encryption is a technique in the field of security, which is the most efficient for the time being. It has long been introduced, and is often used in the system of data security. Generally, a good cryptosystem scheme must satisfy a combination of four different goals (Felder, 1998; Malhotra *et al*., 2007; Furht & Ilyas, 2013):

i. Authentication

The sender of the message should be able to sign it in such a way that an intruder cannot forge the signature.

ii. Non repudiation

The owner of a signed message should not be able to gainsay his/her signature.

iii. Data integrity

The intended recipient of the encrypted message should make sure that an intruder has not modified the message.

iv. Confidentiality

Keeping the data involved in an electronic transaction private. Typically this is provided by encryption.

Among the important components that should be common knowledge associated with the encryption are plain text, cipher text and cryptography keys. Cryptography keys are a very important component in determining the level of security of an

encryption algorithm. This is because, if the encryption key the more complex, the algorithm becomes more difficult to hack.

Plain text is the original message before being transformed. After the message is transformed, it is called cipher text. An encryption algorithm transforms the plain text message into cipher text. A decryption algorithm transforms the cipher text back into plain text. The sender uses an encryption algorithm, and the receiver uses a decryption algorithm (Diffie & Hellman, 1976). The term *cipher* is also used to refer to different categories of algorithms in cryptography. This is not to say that every sender-receiver pair needs their very own unique cipher for a secure communication. On the contrary, one cipher can serve millions of communicating pairs (Gupta, 2012).

2.3.3 Cryptography Key

A key is a number (or a set of numbers) that the cipher, as an algorithm, operates on. To encrypt a message, senders need an encryption algorithm, an encryption key, public key and the plaintext. These create the cipher text. To decrypt a message, receiver needs a decryption algorithm, a decryption key, and the cipher text. These reveal the original plaintext (Forouzan, 2007). As shown in Figure 2.1, cryptography keys are very important to make an encryption process and decryption process. The strength of the cryptography algorithm depends on the complexity of cryptography keys, because unauthorized parties can potentially break the keys.

Figure 2.1 shows the scenario of the encryption process and the components interlinked with each other. The key is the trigger mechanism to the algorithm. The actual cryptographic process is generally a complicated mathematical formulation, the more complex, the more difficult to break the cipher text, and more secure. A key is supplied to the recipients so that they can then decipher the message. Keys for encryption algorithms are described in terms of the number of bits. The higher the number of bits, the more difficult that cryptosystem would be to break. Decryption is the process of converting encrypted data back into its original form, so it can be understood.

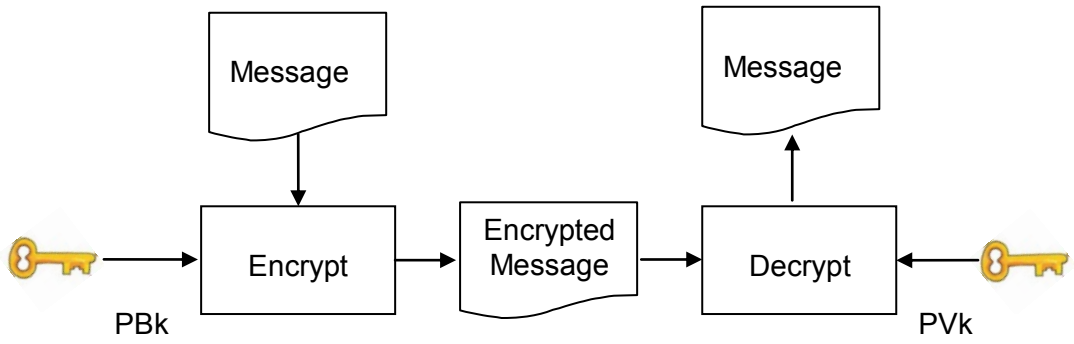

Figure 2.1: Cryptography Application Process

Encryption and decryption are especially important in wireless communications. This is because wireless circuits are easier to tap than their hard-wired counterparts. Nevertheless, encryption or decryption is a good idea when carrying out any kind of sensitive transaction, such as a credit-card purchase online, or the discussion of a company secret between different departments in the organization. However, as the strength of the encryption or decryption increases, so does the cost. Cryptography can be divided into two groups: symmetric keys, also called secret-key cryptography algorithms, and asymmetric keys, also called public key cryptography algorithms (William & Stallings, 2006), as shown in Figure 2.2.

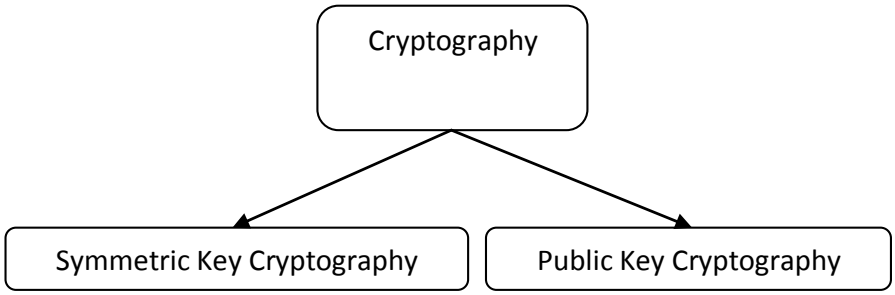

Figure 2.2: Type of Cryptography

2.3.1 Symmetric Key Cryptography

Secret key encryption is commonly known as secret key, or symmetric encryption. Symmetric encryption is the oldest and best known technique. A secret key, which can be a number, a word, or just a string of random letters, is applied to the text

of a message to change the content in a particular way. This might be as simple as shifting each letter by a number of places in the alphabet. It is called private key encryption because each party must know it before the message is sent. The same private key is used to encrypt and decrypt the information in symmetric algorithms (Babu *et al*., 2010), as shown in Figure 2.3. As the same (secret) key is used for encryption and decryption in symmetric cryptosystems, it is necessary to transfer the secret key among all communicating parties before secure communication begins. The theory behind symmetric ciphers is a little fuzzier. Most ciphers are based on pieces of information theory and experience and theory from cryptanalysis (Diffie & Hellman, 1976).

One of the major issues with symmetric-key systems are to find an efficient method to agree upon and exchange keys securely. This problem is referred to as the key distribution problem (Menezes *et al*., 1996). The second problem is that a digital signature is not available in secret key cryptosystems (Stinson, 2005). To solve the problem faced by private key cryptography, Diffie and Hellman proposed in 1976 the concept of public key cryptography. This concept has been the biggest revolution in cryptography (Guo *et al*., 1999).

When compared to public-key algorithms, symmetric algorithms are very fast and require less CPU-intensive calculations. That is why they are preferred when encrypting large amounts of data. Some of the most common symmetric algorithms are RC4 and Data Encryption Standard (DES). Many modern cryptographic protocols use a combination of public-key cryptography and symmetric cryptography to obtain the benefits of both: public-key algorithms to exchange a symmetric key, and then use symmetric algorithms to quickly encrypt or decrypt data. Such encryption protocols include TLS, SSL, IPSec, EFS, S/MIME, among others.

CHAPTER 1

INTRODUCTION

1.1 Research Background

Information security, also known as computer security, is an approach to protect information from unauthorized access, disruption, inspection, modification or manipulation of information (Diffie & Hellman, 1976). Computer security or cryptography ensures a secure communication from the manipulation of third parties (Rivest, 1992b). Security and confidentiality of the cryptography process are also known as encryption. In encryption technology, only legitimate recipients can read a message sent across the network. In a secure communication, a message should be read by the intended recipients, hence, shouldn't be intercepted by any third parties. Therefore, an encryption algorithm is used to transmit data over public networks.

A cryptography, or encryption, consists of three important elements, namely, encryption, decryption, and key generation (Wenbo, 2004). Encryption is the process used to convert or transform plain text to unreadable code known as cipher text. Decryption is the reverse transformation (encryption transformation). Key generation is the main problem faced in cryptography, because hackers attempt to crack keys used. The key generation process involves determining the public and private keys used during the encryption and decryption processes. The level a cryptography algorithm provides security depends on the complexity of the cryptography keys. Cryptography can be divided into symmetric and asymmetric techniques. In symmetric cryptosystems, the same key is used for both encryption and decryption, so a secure medium is needed to ensure the key is transmitted securely, because the keys are shared between sender and receiver (Tan *et al*., 2009). These cryptosystems are considered fast and easy to use. However, problems occur if the key used is acquired by third parties. When this

problem occurs, the third party will be able to decrypt the data, since the same key is used in both the encryption and decryption processes.

In asymmetric or public-key cryptosystems, two different keys are used: public key and private keys. Using a receiver's public key, the sender encrypts the message and sends it to the receiver, and the receiver decrypts the message with his private key. The well-designed implementation of the public-key cryptosystem was carried out by Rivest, Shamir and Adleman, called the RSA Public-Key Cryptosystem (Rivest *et al*., 1983). RSA is the most popular and widely used cryptosystem, because RSA algorithm security depends on the difficulty of discovering the private key (Anane *et al*., 2010). The RSA algorithm is commonly known as a mechanism for providing a relatively good security. Although the RSA algorithm is more secure than symmetric key algorithms, there are several disadvantages in implementing the RSA algorithm. The RSA algorithm uses a key consisting of a row of numbers. Moreover, it requires a large storage space, and is only suitable to be used on devices with a large amount of memory.

In 1984, Shamir (Shamir, 1985) proposed a public key encryption scheme in which the public key can be an arbitrary string. Shamir's original motivation for identity based encryption was to simplify the certificate management. Several proposals for identity based encryption (IBE) schemes have been proposed (Desmedt & Quisquater, 1987; Maurer & Yacobi, 1991; Boneh & Franklin, 2003; Hühnlein *et al*., 2003). In this algorithm, a user's identity is chosen as the public key to replace the numbers that are used in the RSA public key algorithm. However, IBE requires a Certification Authority (CA) as a public key authentication agent. Identity based encryption needs a CA to provide a trusted public key to the various participants on demand, and to set up the hierarchical infrastructure for numerous CA's extra overhead that is required.

Therefore, the Tripathi algorithm has produced a similar algorithm to IBE that uses a string as the public key. The Tripathi algorithm proposes an RSA based algorithm to generate the cryptographic keys using an identity, such as a person's E-mail address. This algorithm helps the sender to recall the public key for each receiver before encrypting the messages to be sent to a specific receiver. In this algorithm, a user's identity will be chosen as the public key to replace the numbers that are used in

the RSA public key algorithm. The Tripathi uses a hash function to create a public key. A string as a public key is also used in IBE, but the Tripathi algorithm does not use CA to authenticate the public key, but uses a hash function instead. The advantage of using this algorithm is that users can easily recall the public, key since it is their own. The E-mail address is public information that is well-known by its owner, hence, making it suitable to be used as a public key in an encryption algorithm.

The Tripathi algorithm enhances the RSA algorithm by using a string as the public key instead of numbers. However, there are also some problems with using the Tripathi algorithm. The main problem is that the Tripathi algorithm is unable to use all users' E-mail addresses as public keys; only a few E-mails can be used as public keys. Therefore, the Tripathi algorithm needs to be improved in order so that all E-mail addresses can be used as public keys. The algorithm should be applicable to all E-mail domains, such as yahoo, gmail, edu, companies, etc. Therefore, this research aims to enhance the Tripathi to ensure all E-mails can be used as public keys.

1.2 Problem Description

There are three main problems associated with the Tripathi algorithm. The first problem is that it is cannot produce satisfactory results because it is unable to use all users' E-mail addresses as public key keys. Therefore, only certain E-mails can be used as a public key. If a particular E-mail cannot be used as a public key, the user must provide another E-mail.

The second problem is that the Tripathi algorithm uses the same modulo value to test every E-mail, and when the same modulo value is used for every E-mail, the E-mail probably cannot be used as a public key. This is because, if the E-mail is not a relative prime to modulo value, the E-mail cannot be used as a public key, and consequently the user must use another E-mail.

The third problem is that the Tripathi algorithm cannot use an even decimal value as a public key. This is because a decimal value can be an odd number and an even number. Therefore, the proposed algorithm ensures all E-mails can used as a public key.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The proposed algorithm is called the CLB-RSA algorithm. “CLB” stands for classification and loop based process. “C” stands for classification, while “LB” stands for loop based. Two enhancement processes are used in the proposed algorithm. The results were good after implementations of these two processes. This algorithm focuses on the replacement of the Certificate Authority (CA) that used in Identity Based Encryption (IBE). A CRC32 hash function is used to create a public key, as used in the Tripathi algorithm. The Tripathi algorithm was able to use a user identity as a public key. The proposed algorithm can also use a user identity as a public key. Examples of a user’s identity are name, E-mail, and nickname. In algorithm testing, E-mail addresses are used as a public key to determine whether the developed algorithm is able to use all types of E-mails from various domains. E-mail addresses chosen were for testing from domains issued by yahoo, Gmail, Hotmail, etc. The aim of this research is to ensure that the resulting algorithm is able to use all of the different types of E-mail domains as public keys. The proposed algorithm is an enhancement of the Tripathi algorithm. The improvement of the Tripathi algorithm is to produce a more complete algorithm that gives better results.

The improvement was made to ensure that all E-mail addresses can be used as a public key. This is because the Tripathi algorithm cannot use all the E-mails as a public key; only certain E-mails can be used. A comparison of the performance can be seen between the Tripathi algorithm and the proposed algorithm. A comparison was performed for both algorithms in terms of the number of E-mails that can be used as public keys. The comparison was done on 20 E-mail samples. The proposed algorithm

(CLB-RSA) consists of five components, which is similar to the original RSA algorithm. The five components are input, key generation, encryption, decryption and output. However, this research focuses on the key generation process only, because public keys and private keys will be distributed in the key generation process prior to being used in encryption and decryption.

This chapter also discusses the key generation process, which is the most important component in the production of cryptographic keys. These key generations are modified and enhanced to make sure this algorithm is able use all types of E-mails as a public key. This enhancement process is used to complement the Tripathi algorithm, which is unable to use all E-mails as a public key. Only some of the E-mails are effectively used as a public key. For a simple example, user A has an E-mail address *asaiman@E-mail.com.my*. This E-mail is tested in the Tripathi algorithm, and whether it can be used as a public key or not. If this E-mail cannot be used as a public key, the user must provide another E-mail address to be used as a public key. Hence, this CLB-RSA algorithm was developed to make sure users only need to have one E-mail. The CLB-RSA will make sure that E-mails from any domain can be used as a public key. Therefore, all E-mails can be a public key, and every user only needs to provide one E-mail.

3.2 CLB-RSA Components

Figure 3.1 shows the five components in the CLB-RSA algorithm. These components are also included in previous algorithms, such as RSA and Tripathi. Five components are interlinked with each other. It consists of three very important process, which are the key generation process, process of encryption, decryption process, as well as input and output, as shown in Figure 3.1. The key generation process is the most important part of the RSA algorithm, because it provides the public and the private keys. An encryption algorithm transforms the plaintext into cipher-text, and a decryption algorithm transforms the cipher-text back into plaintext. The sender uses an encryption algorithm, while the receiver uses a decryption algorithm.

An original intelligible message which is fed as input before being transformed is called plain text (William & Stallings, 2006). Input is data to keep confidential from

the public. It consists of a variety of forms, including plain text data or image data. However, to test this algorithm, plain text data will be used. The sender enters the receiver's public key to send messages to the receiver. The input is converted into a code form that is known as cipher-text. If hackers acquire this cipher-text, they will not be able to read or understanding this code, because they do not have the private key. The cipher-text is a form of data that cannot be understood. Cipher-text can only be converted to plain text if the correct private key is used.

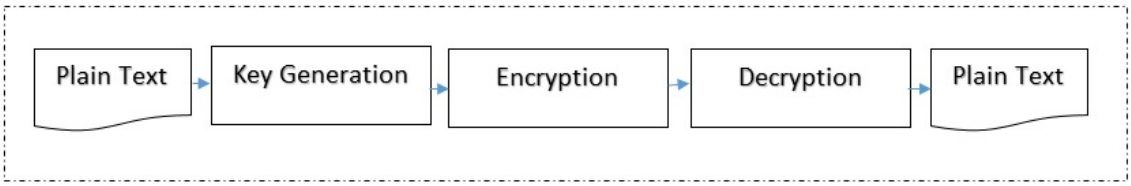

Figure 3.1: RSA Components

The enhancement for the proposed algorithm (CLB-RSA) is only on the key generation process, in order to make sure all E-mails can be used as public keys. So the others processes remain similar to the Tripathi algorithm. However, the idea to develop this proposed algorithm comes from the issue inherent in the Tripathi algorithm. The enhancement process is only on the key generation process, as show in Figure 3.2. This figure shows the processes used in the CLB-RSA algorithm to make sure all users' E-mails can be used as public keys. The difference between the CLB-RSA and the Tripathi algorithm is in the key generation process.