AENSI Journals

Australian Journal of Basic and Applied Sciences

ISSN:1991-8178

Journal home page: www.ajbasweb.com

Enhancement of RSA Key Generation Using Identity

[1]Norhidayah Muhammad, [1]Jasni Mohamad Zain, [2]Md Yazid Mohd Saman, [3]Mohd Fadhil Ramle

[1]*University Malaysia Pahang, 26600, Pahang, Malaysia*
[2]*University Malaysia Terengganu, 21300, Terengganu, Malaysia*
[3]*Kolej Komuniti Kuala Terengganu,*

ARTICLE INFO

Article history:
Received 25 April 2014
Received in revised form
8 May 2014
Accepted 20 May 2014
Available online 17 June 2014

Keywords:

ABSTRACT

The purpose of this paper is to enhance previous algorithm called Tripathi algorithm. The Tripathi algorithm proposes an RSA based algorithm to generate cryptographic keys using user identity such as email address of a person. This algorithm used user's identity to replace the numbers that are used as a public key in the RSA algorithm. However, the Tripathi algorithm cannot use all of the users' email addresses as a public key. This is because, there are two reasons why it is unable to use all email addresses: i) this algorithm use the same modulo value for every email, if the email is not related prime to modulo value, the new email should be entered. ii) Entered email is composed of odd and even number. If the email is even number, then it cannot be the public key. Therefore the Tripathi algorithm needs to be improved. Proposed algorithm called CLB-RSA has been implemented. This algorithm can used all user emails as a public key, and this achievement is after two experiments are done on this study.

To Cite This Article: Norhidayah Muhammad, Jasni Mohamad Zain, Md Yazid Mohd Saman, Mohd Fadhil Ramle, Enhancement of RSA Key Generation Using Identity. ***Aust. J. Basic & Appl. Sci.,*** *8(21): 93-98, 2014*

INTRODUCTION

Information security or also known as computer security is an approach to protect information from unauthorized access, disruption, modification or manipulation of information (Diffie, 1976). Sensitive data need to protect from unauthorized by any unrelated person. As done by (Zain, 2006) and (Zain, 2007) to protect medical image. Cryptographic system or cryptosystem is a form of encryption, decryption algorithm, and the key generation (Wenbo, 2004). Key generation become the biggest problem for cryptography. This is because the key generation process to determine the public key and private key for use during encryption and decryption process. The safety of a cryptography algorithm depends on the complexity of a cryptography keys. RSA algorithm security is commonly known can be a tool for good security. The actual RSA cryptographic process is generally a complicated mathematical formulation, the more complex of keys, the more difficult to break the cipher text, and more secure, the disadvantage of RSA algorithm is the RSA algorithm using a key consisting of a row of numbers and also requires large storage space and is only suitable for use on large device memory.

Several techniques had already been proposed for distribution of public keys. Out of them one was public key certificates. The basic idea was to use trusted third party called Certificate Authority (CA) to provide trusted public key to the various participants on demand. To setup the hierarchical infrastructure for numerous CA's extra overhead was required. In 1984, Shamir (Shamir, 1985) proposed 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 IBE schemes (Desmedt, 1987; Maurer, 1991; Hühnlein, 2003; Boneh, 2003). Hence the Tripathi algorithm (Rivest, 1983) presents an RSA based algorithm to generate the cryptographic keys required by participants for secure communication using their identities which is similar to identity based encryption scheme (IBE).

However the main problem in Tripathi algorithm is cannot use all of the user email addresses as a public key. This is because, there are two reasons why it is unable to use all email addresses: i) this algorithm use the same modulo value for every email, if the email is not related prime to modulo value, the new email should be entered. ii) Entered email is composed of odd and even number. If email users cannot be used as a public key, then the user must enter a new email and the process is repeated until the user's email can be used as a public key. Therefore the Tripathi algorithm needs to be improved. In this study, two times of experiments that were

Corresponding Author: Norhidayah Muhammad, University Malaysia Pahang, 26600, Pahang, Malaysia,
E-mail: mrs.hidayah@yahoo.com.my

carried out to produce satisfactory results and make sure all emails entered by user can be a public key. In a first experiment, the algorithm that has been improved from the Tripathi algorithm and it's called LB-RSA.

Looping process has been added to this algorithm to produce a new modulo value and helps to produce more email addresses that can be used as a public key and LB-RSA have shown good results, that is, 50% of the total number of emails, can be used as a public key. This number was increased compared to the amount generated by Tripathi algorithm that is 25%. The resulting decision issued by LB-RSA algorithms in experiment 1 do not reach 100%, the second experiment carried out to provide a better algorithm and can achieve 100% of emails that can be used as a public key. The classification process added in LB-RSA algorithm is to determine whether the decimal is even or odd number, this process helps to make all email can be a public key. Before emails are tested in these algorithms, emails will be converting to decimal value.

1. Related work:

1.1 Tripathi Algorithm:

RSA algorithm was gone through several phases of change towards, a variety of improvements have been made in the original RSA algorithm, including algorithms that have been developed by Sachin Tripathi (Tripathi, 2011). This algorithm has several advantages compared with the original RSA algorithm. This algorithm also has several advantages compared with the original RSA algorithm. 1) Tripathi algorithm used user string identity as a public key such as an email address. 2) Need small device memory to store the public key. The algorithm presents the RSA based algorithm to generate the cryptographic keys required by participants for secure communication using their identities, which is similar to identity based encryption (IBE) scheme. Hence the authentication of the public key in IBE is a big challenge and for which public key certificates, provided by a Certification Authority (CA), are used. At large scale communication to set up numerous CA's is a major overhead of public key cryptography.

Hence this algorithm attempts to avoid the use of public key certificates and proposes an RSA based algorithm to generate the cryptographic keys using identity such as an email identity of a person. Fig 1 shows the key generation algorithm of the Tripathi flow algorithm. The email address entered will be converted into a fixed - size string using CRC32 hash function and then converted to a decimal value before test in this algorithm. After that the decimal value h(id) will be tested whether it is related prime to modulo phi (n) or not, h(id) is related prime to the modulo phi (n), then email address can be used as a public key, but if the email is not related prime to the modulo, then the user must enter a new email as a public key. This is an example of email address convert to form of decimal using CRC32 hash function. However the problem in Tripathi algorithm is unable to use all user email as a public key. There are two reasons why it is unable to use all email addresses: i) this algorithm use the same modulo value for every email, if the email is not related prime to modulo value, the new email should be entered. ii) Entered email is composed of odd and even number.

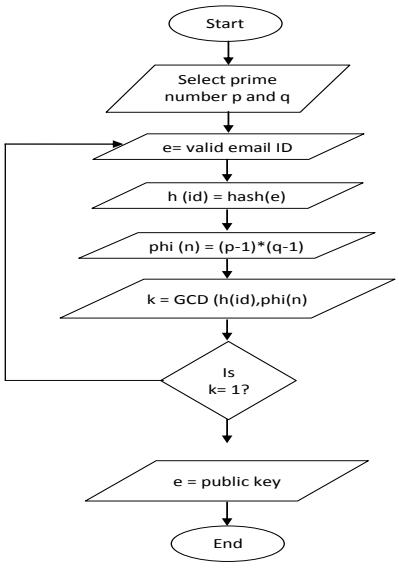

Fig. 1: Tripathi Algorithm Flowchart.

2. Proposed Algorithm (CLB-RSA):

The methodology proposed will be named CLB-RSA algorithm. The character of "CLB" is standing for classification and loop based process. A representation of "C" is standing for classification and "LB" is standing for loop based. The advantage of this algorithm is able to use all user emails address as a public key. There are

two enhancement process added in this proposes algorithms are: looping and classification process. Advantages of using user identity abridged in order to facilitate the user to remember public key. Example of user identity is including user name, user email, nickname and so on. Improvements done on this algorithm has improved the performance of CLB-RSA, the performance on the number of emails that can be used as a public key is increase and reach of hundred percent results.

The improvement was made to ensure that all email address can be used as a public key. As discussed in related work, Tripathi algorithm cannot use all the emails as a public key, and only certain email that can be used as a public key. Comparison of the performance can be seen between the Tripathi algorithms and CLB-RSA algorithms. Comparison between these algorithms is in terms of the number of emails that can be made as a public key email based on twenty samples listed in Table 1. CLB-RSA is second algorithm evaluate from first experiment by Muhammad [12].

2.1 LB-RSA Algorithm:

In this study, two times of experiments that was carried out to produce an algorithm that can produce a satisfied result. In a first experiment, the algorithm that has been improved from the Tripathi algorithm and it's called LB-RSA [12]. Looping process is a process that is added in LB-RSA to generate the new parameters of modulo phi (n). In Tripathi algorithm, looping process will be run on email address, it's mean, if email cannot be a public key, so users need to enter new email address, so the loop process is happen on email address. However in LB-RSA algorithm, the looping process is running modulo value phi (n), if the email address cannot be a public key, so new modulo value will be created. When the new parameters of phi (n) are produced, then the probability of the value of h (id) and phi (n) is related prime is higher, and it is usually referred to equation k =GCD (h(id), phi (n)). If k is equal to 1, so this it meant h (id) is related prime to phi (n), and the email's address can be used as public key. In Tripathi algorithm, modulo phi (n) generated only once and if the value of k is not equal to 1, then the email entered is declared as a non-public key, and the user should enter a new email address. These processes write:

If GCD (h(id),phi(n))=1;
h(id)=public key;
else
Looping process until k=1;

Fig 3 shows the flowchart of LB-RSA and Tripathi key generation algorithm. This comparison between two algorithms: Tripathi algorithm and LB-RSA algorithm. As shown in Fig 3, the main differences between these algorithms are, in Tripathi algorithm, looping process is running on email address. When the email entered by the users, can't be a public key, so users need to enter the new email address until email address can be used as a public key. So it's mean reenter the new emails is the looping process in Tripathi algorithm. However in LB-RSA algorithm, the looping process occurred on modulo value, if the email address can't be a public key, thus a new modulo value phi(n) will be continuously created until the email can be a public key. So it's mean the looping process is done in modulo value and not on email address. This process will be fully effective because the email addresses that can be a public key are increasing in LB-RSA algorithm.

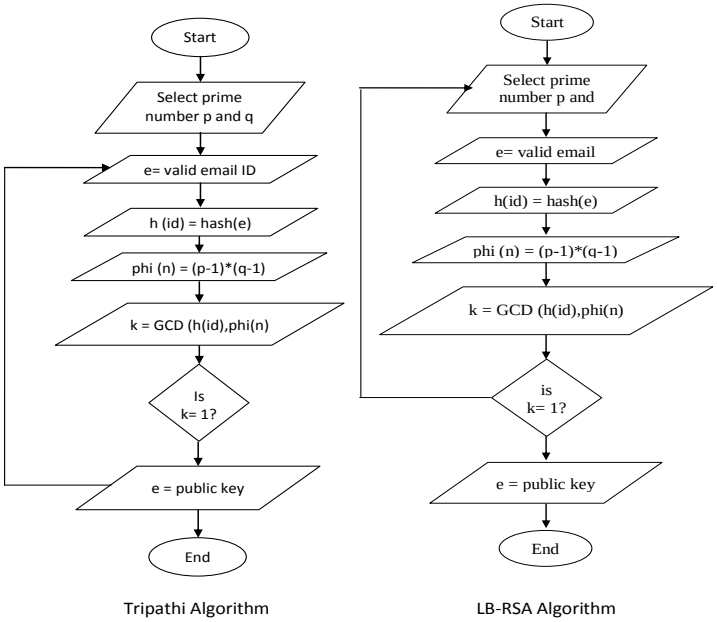

Fig. 2: LB-RSA key generation algorithm.

2.2 CLB-RSA Algorithm:

The mathematical calculations that are used in CLB-RSA key generation algorithm describe to show detail how it's calculated. Steps 6 and 7 are the additional steps of this algorithm different with Tripathi algorithm. Small number used in example below to give simple calculation, and also used even decimal value as a public key to show how this algorithm make a classifying process to determine either the decimal entered is even or odd number. The even number of decimal value is converted to odd number by reducing the value as shown in step 6. After emails tested, and the result for entered email is an odd number, then the next process can be carried out and an email address can be used as a public key, even maybe the looping process requires more than a one-time loop. However, if the decimal value is even, then the value should be converted to the odd before proceed to the nest process. Therefore, when the value is odd umber, then the email can be used as a public key. This classifying process will be solving the second problem (even decimal value can't be a public key) in Tripathi algorithm that causes this algorithm cannot used all email as a public key. Therefore LB-RSA can solve the first problem that causes the Tripathi algorithm cannot used all email as a public key using looping process hence create the new modulo value until email an modulo value is related prime.

To convert an even decimal value to odd decimal value, decimal value should be minus 1 or plus 1. But for this algorithm, the decimal value will be minus 1 to produce odd decimal value. Once the email is turned to the odd number, and then the next process can be implemented. This is the difference process between LB-RSA and CLB-RSA, and this improvements step can complement this algorithm because CLB-RSA can make all email entered can be used as a public key. The addition process of this algorithm is to make tests on the decimal to classify the decimal value, whether it is an odd number, or even number. This equation is used for this process. If h(id) = odd, then continue step 7, else h(id)-1, continue step 7. Figure 3 shows the flowchart of the steps in CLB-RSA key generation algorithm, the process begins with the declaration of two random numbers as a parameter p and q and lastly public key and private key are determined.

Key Generation:

1. Select two parameters, namely p and q, these two parameters must be a prime number.

p= 3, q= 7

2. Compute n by using the following formula.

n = p * q

n= 21

3. Compute phi (n).

phi (n) = (p – 1) * (q – 1)

phi(n) = 12

4. Input the user valid email Id.

e = valid email id

e = odedny@012.net.il

5. Convert email id to hash function.

h(id) = hash(e)

h(id) = 1812161612

6. Classified decimal value.

if h(id) = odd number?

Continue step 7

else

h(id) – 1

Continue step 7

1812161612== odd?= NO

else

1812161612-1=1812161611

h(id)==1812161611

Continue step 7

7. Choose the exponent k if :

If GCD (h(id), phi (n)) = 1

Continue step 8

Else, loop step 1 the process until k=1

GCD(1812161611, 12) = 1

8. Compute private key exponent d through following formula

d = e ^ -1 mod (phi (n))

d = 7

9. Thus the public key consists of public key exponent e and n. And the private key consists of private key exponent d and n.

Public key: (n, e)
Public key: (n, d)
Public key: (21, 1812161611)
Public key: (21, 7)

Fig 3 shows the comparison between two key generation algorithms, LB-RSA and CLB-RSA. As shown in Fig 3, the main difference is the classification process. This process is added in this algorithm to make classifying on decimal value, this process is very important because if the decimal value is even number, so the email address entered cannot be a public key. Classification process added in this algorithm is to make a classifying of decimal value to odd or even number. If the decimal value is an odd number, so there is nothing happened to the decimal value and can continue to the next step and make is as a public key. However, if the decimal value is even, so this decimal value must be converted to odd number first before proceeds as a public key. However the looping process is still available in this algorithm because it will be useful when the decimal value is an even number. If a decimal value is an odd number, but decimal value and modulo value is not a prime number, so new modulo value will be created and this process is fully working when the email that can be a public key is increasing.

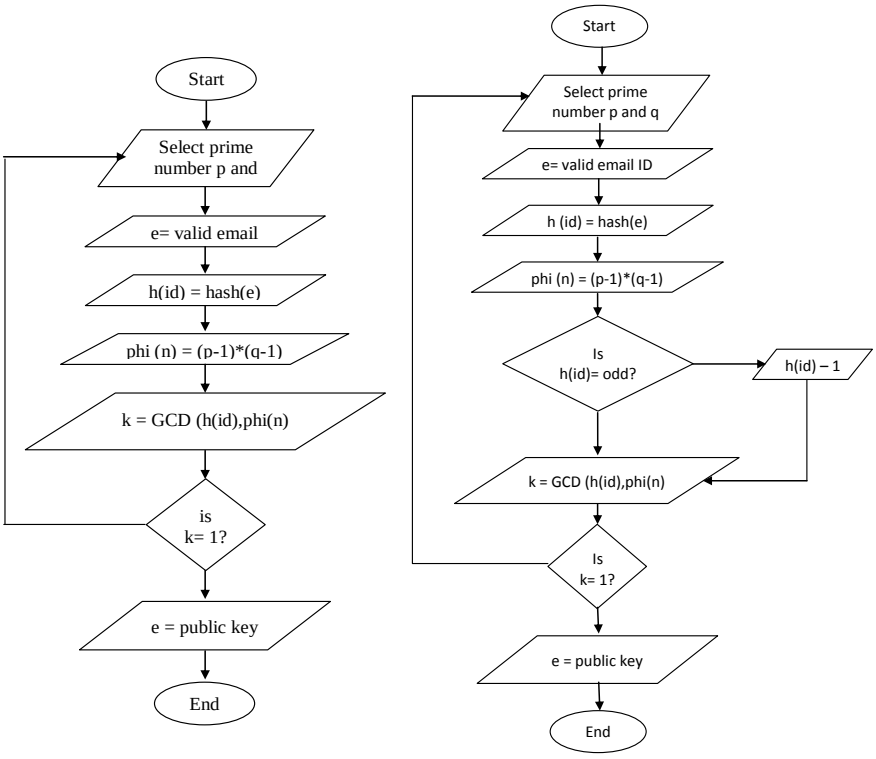

Fig. 3: CLB-RSA Algorithm Flowchart.

Table 1 shows the result produced by LB-RSA and CLB-RSA. LB-RSA algorithm is tested in experiment 1 and CLB-RSA algorithm is tested in algorithm 2. Both of these tests using the same twenty sample email listed in table 1. In LB-RSA, half of data can be used as a public key, and a half of the data cannot be a public key, but in CLB-RSA all data can be used as a public key. This proves the effectiveness of CLB-RSA algorithm and the addition step is performed on this algorithm have a significant impact on this algorithm, successfully making all the emails as a public key.

Table 1: Result LB-RSA and CLB-RSA

No	Email	Decimal	LB-RSA	CLB-RSA
1	yaron4329@coolmail.co.il	786745081	Yes	Yes
2	odedny@012.net.il	2730798235	Yes	Yes
3	sismal@t2.technion.ac.il	1812161612	No	Yes
4	knarik2000@mail.ru	3238888356	No	Yes
5	simon.goldman@intel.com	795509258	No	Yes
6	kristal@netvision.net.il	3700340025	Yes	Yes
7	daniel227@bezequnt.net	4067366447	Yes	Yes

8	nitzan@tzel.net	1199321065	Yes	Yes
9	tall_g@telepark.co.il	3074048744	No	Yes
10	zhan_t@hotmail.com	2205045334	No	Yes
11	Jacklml@consultant.com	3009834837	Yes	Yes
12	alayan1@012.net.il	2654073728	No	Yes
13	romangr@matrix.co.il	3765641505	Yes	Yes
14	rbarash@elta.co.il	296517678	No	Yes
15	motyy@isa.gov.il	2386521674	No	Yes
16	mrs.hidayah@yahoo.com.my	2962995952	No	Yes
17	fatem_alyahya@yahoo.com	3843922542	No	Yes
18	igor@bizportal.co.il	2489219047	Yes	Yes
19	max@usermail.com	3211800885	Yes	Yes
20	sharon_m@ifat.com	2388137343	Yes	Yes

Conclusion:

The algorithm has been improved and called CLB-RSA has achieved the objectives of improving the RSA algorithm using the email address as the public key, and make sure all the email entered by the user can be used as a public key. The email addresses used in this study consisted of a variety of different domains. The results produced by CLB-RSA are very gratified that 100% of the email can be public key. These results are better than the previous algorithm (Tripathi algorithm). Problems faced by Tripathi algorithm can be solve by CLB-RSA algorithm using loop based and classification process. However in future, this algorithm can be improved better than CLB-RSA such as private key can be replaced with private user identity.

ACKNOWLEDGMENTS

This study is funded by Skim Latihan Akademik IPTA (SLAI) under the Malaysia Ministry of Higher Education (MOHE) and Universiti Sultan Zainal Abidin (UniSZA) Malaysia. The authors would like to acknowledge all contributors, and others who have helped and greatly assisted in the completion of the study.

REFERENCES

Boneh, D., M. Franklin, 2003. Identity-based encryption from the Weil pairing. Vol. 32.Journal on Computing SIAM, pp: 586-615.

Desmedt, Y., J.J. Quisquater, 1987. Public-key systems based on the difficulty of tampering (Is there a difference between DES and RSA?). Advances in Cryptology—CRYPTO'86.

Diffie, W., M.E. Hellman, 1976. New directions in cryptography. Vol. 22.IEEE Transactions on, Information Theory, pp: 644-654.

Hühnlein, D., J. Michael Jr, D. Weber, 2003. Towards practical non-interactive public-key cryptosystems using non-maximal imaginary quadratic orders. Vol. 30.Designs, Codes and Cryptography, pp: 281-299.

Maurer, U.M., Y. Yacobi, 1991. Non-interactive public-key cryptography. Advances in Cryptology—EUROCRYPT'91.

Muhammad, N., J.M. Zain, M.Y. Mohd Saman, 2013. Loop-based RSA key generation algorithm using string identity. 13th International Conference on Control, Automation and Systems (ICCAS).

Rivest, R.L., A. Shamir, L. Adleman, 1983. A method for obtaining digital signatures and public-key cryptosystems. Vol. 26.Communications of the ACM, pp: 96-99.

Shamir, A., 1985. Identity-based cryptosystems and signature schemes. Advances in cryptology.

Tripathi, S., G.P. Biswas, S. Kisan, 2011. Cryptographic keys generation using identity. 3rd International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom 2011).

Wenbo, M., 2004. Modern cryptography: theory and practice. Vol.Publisher: Prentice Hall PTR, Copyright: Hewlett Packard.

Zain, J.M., A.R. Fauzi, 2006. Medical image watermarking with tamper detection and recovery. 28th Annual International Conference of the IEEE on Engineering in Medicine and Biology Society, EMBS'06.

Zain, J.M., A.R. Fauzi, 2007. Evaluation of medical image watermarking with tamper detection and recovery (AW-TDR). 29th Annual International Conference of the IEEE on Engineering in Medicine and Biology Society, EMBS 2007.