A study of public key ‘e’ in RSA algorithm

C Intila1,2, B Gerardo1,3, R Medina1
1 Technological Institute of the Philippines Quezon City, Quezon City, Philippines 1000
2 Poblacion Southeast Zone, Lemery, Iloilo Philippines 5043
3 West Visayas State University Iloilo City, Philippines 5000
1carloski_0822@yahoo.com, 2bobby.gerardo@gmail.com, 3ruji.medina@tip.edu.ph

Abstract. RSA Algorithm is public-key cryptography which exposed from factorization attack based on the public key ‘e’ and modulus ‘n’. The study aimed to modify an RSA Algorithm based on generation of different value of public key ‘e’ and show different result based on original equation of RSA Algorithm. A modified RSA Algorithm was developed to generate different value of public key ‘e’ and compared the result based on the output value and speed of execution. The results show that the modification of RSA based on public key ‘e’ provides additional security.

1. Introduction

The development of internet application has grown so fast but needs secured cryptosystem for data communication. To obtain the user’s satisfaction, high speed algorithms with enhanced security have to be developed [1].

Public key cryptosystem is suitable and a better answer for more secured data communication and it comprises of encryption, decryption, authentication that is design for better data security and privacy [2] [3]. There two categories to select for encryption, the symmetric and asymmetric. Symmetric encryption algorithms are AES [2], DES [4], RC5 [5] , RC6 [6], Blowfish [7]. Asymmetric encryption algorithms are RSA [8], ECC [9], Diffie Helman [10].

The RSA algorithm is best solution in asymmetric cryptography not only for its mostly used but for its large scale of applications[11] and provides better security providing large prime numbers for public key, private key ‘d’ and modulus ‘n’ generation [12]. The Rivest Shamir Adleman (RSA) is an algorithm based on public-key cryptography and is considered one of the great developments in IT security due to its resistance to multiple attacks [13].

RSA Algorithm is the mostly used public key cryptography and the demand for a better security is one of the challenges faced to continue the reliance of the users [14]. Nevertheless, RSA suffers from slow speed, a demand for key deposit, and is generally inappropriate for use in several systems [3]. With the speed and security issues, the researchers are encourage to propose new algorithms with multiple parameters towards better performance of secured network communication [1].

In the RSA Algorithm shows that the public key ‘e’ can be used to factor n respectively [15]. Different studies and researches[16][17][18][19] in modification of RSA Algorithm have been conducted with the modification of modulus ‘n’ but disregard the modification of public key ‘e’. With this, the study aims to
modify the RSA Algorithm based on public key ‘e’ and show the importance of its security in the encryption and decryption process.

2. Related Literature

Several studies conducted to RSA Algorithm with different factors to modify. The following are the studies: A better version of RSA algorithm with new element ‘f’ as replacement for the exclusion of modulus ‘n’. It increases the security against factorization but slightly increase the time consumed [17]; another paper presented a one sender to many receiver asymmetric cryptosystem that uses RSA as a base. The proposed approach used Chinese Remainder Theorem (CRT) to address the limitation of multi-asymmetric RSA Scheme [8]; a modification comparing to Original RSA Algorithm in terms of speed and security. Their Modified RSA Algorithm compare to Original RSA Algorithm is better in terms of speed and security [20]; a modification of RSA Algorithm that utilized four prime numbers in generating modulo ‘n’ to improve the security against factorization attack[21][18]; a study that proposed a RSA Algorithm that uses two public key which increase the security in decrypting message but decrease its speed [22]; a research conducted where the proposed Modified RSA algorithm improves the security against the brute force attack [3]; a study developed a Modified RSA Encryption Algorithm (MREA) which improved security against factorization attack and brute force attack[16]; proposed a Binary RSA Encryption Algorithm. They developed a modified RSA Algorithm that transform original message to binary code that gives better security in decryption process but gives moderate time and space difficulty [23].

3. Methods

3.1 Scope and Limitations

The study limits its content in the modification of RSA Algorithm based on public key ‘e’. The comparison will limit only between the Original RSA Algorithm and the Modified RSA Algorithm. The objective of the study is to give importance of the public key ‘e’ in modification of RSA Algorithm.

3.2 The Modification

Figure 1 shows the flow involved in the Original RSA Algorithm processes. Figure 2 shows the Modified RSA Algorithm of the study. Initial modification starts with the public key and followed by encryption formula.
Figure 3 shows the modification of public key ‘e’ from original public key ‘e’ of RSA Algorithm. RSA includes three processes[24].

A. Key Generation
1. Select large prime numbers p and q.
2. Compute n = p * q.
3. Compute \( \phi(n) = (p-1) \times (q-1) \).
4. Select public key e.
\( \text{GCD}(e, \phi(n)) = 1 \), \( 1 < e < \phi(n) \) and coprime with n, \( \phi(n) \).
5. Compute where condition “\( de \mod(n) = 1 \)”

B. Encryption
\[ C = M^e \mod n \]

C. Decryption
\[ D = C^d \mod (n) \]

The steps below will show the modification process.

**Modified RSA Algorithm Based on Public Key ‘e’**
The proposed study was to modify the Public Key ‘e’ value for more secure RSA Algorithm.

A. Key Generation
1. Select large prime numbers p and q.
2. Compute \( n = p \times q \).
3. Compute \( \phi(n) = (p-1) \times (q-1) \).
4. Collect e with the following condition \( \{ p > e > \phi(n), \text{coprime } \phi(n) \text{ and } n \} \) (maximum of ten values)
5. Select random ‘e’ from list
6. Calculate \( f = (e^2 + 1) \).
7. Select d with the following condition \( \{ de \mod(n) = 1 \} \)
8. Send Public key (f, n)
The ‘f’ serve as a new public key which will hide the original e value.
9. Send Private key (d, n)

B. Encryption
\[ C = M^{(f-1)/2} \mod (n) \]

C. Decryption
\[ D = C^d \mod (n) \]

Sample
A. Key Generation
1. Select large prime numbers p and q, \( p=7 \), \( q=11 \)
2. Compute \( n = p \times q, n \)
   \( = 7 \times 11 \)

![Image](image.png)
3. Compute $\phi n = (p-1)(q-1)$
   
   $= (7-1)(11-1)$
   
   $= 6 \times 10$
   
   $= 60$

4. Collect list of public key ‘e’ with the following condition { $p > e > \phi n$, coprime $\phi n$ and n}

   $e = \{13, 17, 23, 27\}$ (maximum of ten values)

5. Select Random ‘e’ from the list.

   $e = 23$

6. Calculate $f = (e^2)+1$.

   $= (23 \times 2) - 1$
   
   $= 46 + 1$
   
   $= 47$

7. Select $d$ with the following condition {de mod n = 1}

   $d = 47$

8. Send Public key (f, g) Public Key = (47, 77)

9. Send Private key (d, g) Private Key = (47, 77)

B. Encryption

M = 4

$C = M^{(e)1/2} \mod (n)$

$= 4^{(47)^{1/2}} \mod 77$

$= 4^{23} \mod 77$

$= 70,368,744,177,664 \mod 77$

$= 9$

D. Decryption

$D = C^d \mod (n)$

$= 9^{47} \mod 77$

$= 7,069,650,490,1e+44 \mod 77$

$= 4$

The Original RSA Algorithm and Modified RSA Algorithm Based on Public Key Generation were developed using Java Programming Language. A modified key generation of RSA Algorithm will be implemented to improve its security. The method to modify the equation of public key will be utilized in concern with the process of encryption and decryption involved. The method to test the two algorithms is to differentiate the result of RSA in generating the public key ‘e’ between Original RSA Algorithm and Modified RSA Algorithm. Also, the two algorithms will be tested in the same hardware and software specifications. Figure 4 shows the method of the study.
4. Result

The Original and Modified RSA Algorithm were implemented using Java Programming Language. The two algorithms were assessed by different file sizes of messages and recorded the efficiency in terms of encryption and decryption process. Both algorithms conducted a test to output cipher text based on formula of different related studies.

4.1 Result for Public Key ‘e’

The first test conducted was generating different value of private key ‘d’ and cipher based on the different value of public key ‘e’ and message ‘m’. The result of cipher text of the Original RSA and Modified RSA Algorithm on public key ‘e’ based on the formula “C=Me mod n” [16] [3][19].

The next three tables show the different result of cipher text with different value of public key ‘e’ in the Original RSA Algorithm.

| Table 1. | p=7 q=11 |
|----------|-----------|
| e | d | m| m| m|
| 13 | 37 | 53 | 26 | 62 |
| 17 | 33 | 16 | 3 | 41 |
| 23 | 47 | 9 | 59 | 62 |

| Table 2. | p=5 q=11 |
|----------|-----------|
| e | d | m| m| m|
| 13 | 37 | 9 | 15 | 51 |
| 17 | 33 | 49 | 25 | 41 |
| 23 | 7 | 9 | 15 | 51 |
| 27 | 3 | 49 | 25 | 41 |
| 29 | 29 | 14 | 25 | 29 |

| Table 3. | p=11 q=17 |
|----------|-----------|
| e | d | m| m| m|
| 3 | 107 | 64 | 125 | 29 |
| 7 | 33 | 115 | 146 | 184 |
| 13 | 37 | 174 | 37 | 95 |
| 19 | 59 | 47 | 108 | 46 |
| 29 | 149 | 157 | 146 | 184 |

The Table 4 and Table 5 show the different result of ciphertext with different value of public key ‘e’ between Original RSA and Modified RSA Algorithm.

| Table 4. Result for p=11 q=17 |
|-------------------------------|
| e(Original RSA) | e(Modified RSA) | d | Original RSA | Modified RSA |
| 3 | 7 | 107 | 64 | 115 |
| 7 | 15 | 23 | 115 | 166 |
| 13 | 27 | 37 | 174 | 115 |
| 19 | 59 | 47 | 0 | 0 |
| 29 | 149 | 157 | 157 | 0 |

| Table 5. Result for p=5 q=11 |
|-------------------------------|
| e(Original RSA) | e(Modified RSA) | d | Original RSA | Modified RSA |
| 13 | 27 | 37 | 9 | 49 |
| 17 | 35 | 33 | 49 | 0 |
| 23 | 47 | 7 | 9 | 0 |
| 27 | 55 | 3 | 40 | 0 |
| 29 | 59 | 29 | 14 | 0 |

4.2 Result for Performance

The following tables and graphs show the result of performance of the original and the modified RSA Algorithm. The two algorithms are developed using Java Programming Language. Table 6, Table 7, Graph 1, and Graph 2 show the encryption and decryption time using a Laptop with specification of Intel Core i7 2.7GHz, 8GB RAM, and with a 64 bit Windows 10 Operating System.

| Table 6. Encryption Time |
|---------------------------|
| Size(KB) | Original | Modified |
| 10KB | 0 | 0 |
| 1000KB | 2 | 2 |
| 5000KB | 7 | 7 |
| 10000KB | 16 | 16 |
| 20000KB | 25 | 26 |
| 50,000KB | 62 | 65 |
| 100,000KB | 152 | 159 |

| Table 7. Decryption Time |
|---------------------------|
| Size(KB) | Original | Modified |
| 10KB | 0 | 1 |
| 1000KB | 2 | 3 |
| 5000KB | 8 | 10 |
| 10,000KB | 16 | 18 |
| 20,000KB | 28 | 31 |
| 50,000KB | 71 | 75 |
| 100,000KB | 146 | 156 |
The table 6 and 7 show the execution time of different file sizes of messages of Original RSA Algorithm and Modified RSA Algorithm. The Graph 1 and Graph 2 display the comparison of execution time between Original RSA Algorithm and Modified Algorithm.

The table below shows the difference in time execution of encryption and decryption process between Original RSA and Modified RSA Algorithm.

**Table 8. Result of Time Execution based on ‘p’ and ‘q’**

| No | p   | q   | n    | Time (second) |
|----|-----|-----|------|---------------|
|    |     |     |      | Encryption | Decryption |
|    |     |     |      | Original | Modified | Original | Modified |
| 1  | 97  | 367 | 35,599 | 2       | 3        | 2        | 2        |
| 2  | 127 | 383 | 48,641 | 2       | 3        | 2        | 2        |
| 3  | 233 | 409 | 95,297 | 2       | 3        | 2        | 2        |
| 4  | 353 | 431 | 152,143| 2       | 4        | 2        | 2        |
| 5  | 373 | 487 | 181,651| 2       | 4        | 2        | 2        |
| 6  | 401 | 523 | 209,723| 2       | 5        | 2        | 2        |
| 7  | 431 | 587 | 252,997| 2       | 6        | 2        | 2        |
| 8  | 457 | 619 | 282,883| 2       | 6        | 2        | 2        |
| 9  | 547 | 677 | 370,319| 3       | 6        | 2        | 2        |
| 10 | 601 | 733 | 440,533| 3       | 6        | 2        | 2        |

5. Findings and Analysis

5.1 Findings

The following are the results and findings of this study:

1. In the Table 1, Table 2 and Table 3, the result shows that with different value of public key ‘e’ designate a different value of private key and different value of public key ‘e’ will also produce different value of cipher text.

2. In the Table 4 and Table 5, the result shows that the Modified RSA Algorithm produce different value of cipher text from the original RSA Algorithm as the result for encryption process.

3. In the result of performance, Table 6, Table 7, Graph 1 and Graph 2, the Modified RSA Algorithm slightly increased its time complexity in encryption. In the decryption process, the Modified RSA Algorithm has almost the same time execution with the Original RSA Algorithm.

4. In the Table 8, the result shows that the Modified RSA Algorithm has slightly increased in time consumed in encryption but not in decryption process.
5.2 Analysis
Based on the findings of the study, in terms with the different valid value of public key ‘e’ designates a corresponding different value of private key ‘d’ and also produce a different value of cipher. In this regard, the level of security in encryption and decryption process increases. In terms of transforming the selected value of public key ‘e’ produced another different value of cipher which increases another level of security in the encryption and decryption process.

In terms of performance, the modification of public key ‘e’ by collecting the list of public key and transforming to new value will slightly increase the time complexity in the key generation and encryption process of RSA Algorithm but not in the decryption process.

6. Summary and Conclusion

6.1 Summary

| Performance   | Original RSA Algorithm | Modified RSA Algorithm |
|---------------|------------------------|------------------------|
| Encryption    | √                      |                        |
| Decryption    |                        | √                      |

The Table 9 shows the summary result between the original RSA Algorithm and the modified RSA Algorithm. The table shows that the modified RSA Algorithm have increase the level of security in terms of public key ‘e’ and encrypted message while increase slight complexity in terms of performance.

6.2 Conclusion

This study presented different values of public key ‘e’ which were evaluated in the encryption and decryption process. Conclusion is established based on the two categories mainly on the security and performance and compared the Modified RSA Algorithm to Original RSA Algorithm.

In the result based on the performance, the modified RSA algorithm has slightly increased in time complexity in the encryption process. In the decryption process, the Modified RSA Algorithm is almost the same of time consumed compare to the original RSA Algorithm.

In the result based on the security, the Modified RSA Algorithm based on Public Key ‘e’ produced more complex result in terms of encryption process than the Modified RSA Algorithm. Different value of public key ‘e’ produced different value of private key and cipher and converting it to new values produced more complex values of cipher which makes the Modified RSA Algorithm based on Public Key ‘e’ more secured encrypted message which also increased security in the decryption process.

Finally, this study suggested that in the modification of RSA Algorithm, the Public Key ‘e’ is one of the factors that enhanced the security in the encryption and decryption process. Modification of public key ‘e’ will increase the security level of the RSA Algorithm.

7. Future Enhancement

The implementation of this study uses the small list of public key ‘e’. The next step of this work is to employ this to the bigger list and implement the modification of RSA algorithm based on public key ‘e’ with modification of modulus ‘n’ which are both exposed to public to increase the level of security of RSA Algorithm against factorization attack, and with the encryption and decryption process. Also, the Modified RSA Algorithm based on public key ‘e’ and modulus ‘n’ will be compared to other Modified RSA Algorithm in terms of security and performance.
References

[1] D. I. G. Amalarethinam and J. S. Geetha, “A Survey on Secured Communication with High Speed using Public Key Cryptography,” vol. 7, no. 4, 2016.

[2] N. Khanezaei and Z. M. Hanapi, “A framework based on RSA and AES encryption algorithms for cloud computing services,” Proc. - 2014 IEEE Conf. Syst. Process Control. ICSPC 2014, no. December, pp. 58–62, 2014.

[3] I. Jahan, M. Asif, and L. Jude Rozario, “Improved RSA cryptosystem based on the study of number theory and public key cryptosystems,” Am. J. Eng. Res., 2015.

[4] A. S. Alkalbani, T. Mantoro, and A. O. M. Tap, “Comparison between RSA hardware and software implementation for WSNs security schemes,” Proceeding 3rd Int. Conf. Inf. Commun. Technol. Moslem World ICT Connect. Cult. ICT4M 2010, 2010.

[5] A. Biryukov and E. Kushilevitz, “Improved cryptanalysis of rc5,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 1998.

[6] R. Rivest, M. J. B. Robshaw, R. Sidney, and Y. L. Yin, “The RC6 Block Cipher,” First Adv. Encryption ..., 1998.

[7] B. Schneier, “The {Blowfish} encryption algorithm,” Dr. Dobb’s J. Softw. Tools, 1994.

[8] A. Mansour, A. Davis, M. Wagner, and R. Bassous, “Multi - Asymmetric Cryptographic RSA Scheme,” pp. 1–8, 2017.

[9] P. Patel, R. Patel, and N. Patel, “Integrated ECC and Blowfish for Smartphone Security,” in Physics Procedia, 2016.

[10] W. Diffie, W. Diffie, and M. E. Hellman, “New Directions in Cryptography,” IEEE Trans. Inf. Theory, 1976.

[11] F. Lombardi and R. Di Pietro, “Transparent security for cloud,” in Proceedings of the 2010 ACM Symposium on Applied Computing - SAC ’10, 2010.

[12] R. Tripathi and S. Agrawal, “Comparative Study of Symmetric and Asymmetric Cryptography,” Int. J. Adv. Found. Res. Comput., vol. 1, no. 6, pp. 2348–4853, 2014.

[13] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” Commun. ACM, 1978.

[14] P. K. Arya, M. S. Aswal, and V. Kumar, “Comparative Study of Asymmetric Key Cryptographic Algorithms,” Int. J. Comput. Sci. Commun. Networks, vol. 5, no. 1, pp. 17–21, 2012.

[15] D. Aggarwal and U. Maurer, “Breaking RSA Generically Is Equivalent to Factoring,” vol. 62, no. 11, pp. 6251–6259, 2016.

[16] R. S. Dhakar, “Modified RSA Encryption Algorithm ( MREA ),” pp. 2–5, 2012.

[17] J. Sahu, V. Singh, V. Sahu, and A. Chopra, “An Enhanced Version of RSA to Increase the Security,” vol. 7, no. 4, pp. 1–4, 2017.

[18] B. P. U. Ivy, P. Mandiwa, and M. Kumar, “A modified RSA cryptosystem based on ‘ n ’ prime numbers,” Int. J. Eng. Comput. Sci., 2012.

[19] N. Kumar, “Implementation of Modified RSA Cryptosystem for Data Encryption and Decryption based on n Prime number and Bit Stuffing,” 2016.

[20] S. A. Jaju and S. S. Chowhan, “A Modified RSA algorithm to enhance security for digital signature,” 2015 Int. Conf. Work. Comput. Commun. IEMCON 2015, 2015.

[21] A. Shamir and L. Adleman, “Introduction,” www.ijecs.in Int. J. Eng. Comput. Sci., vol. ISSN, no. 2, pp. 2319–7242, 2012.

[22] A. A. Ayele, “A Modified RSA Encryption Technique Based on Multiple public keys,” vol. 1, no. 4, pp. 859–864, 2013.

[23] S. A. Nagar and S. Alshamma, “High Speed Implementation of RSA Algorithm with Modified Keys Exchange,” pp. 639–642, 2012.
[24] “Design and Implementation of RSA Algorithm using FPGA Council for Innovative Research,” no. September, 2015.