The innovations made on the RSA cryptosystem using the Euler's totient function

Dongfu Han

Northern Secondary School, Toronto, Canada, M4P 2L5

dongfuhanisabelle@gmail.com

Abstract. Cryptography has been invented for thousands of years as an important tool to safely transfer sensitive messages. Nowadays, cryptography has already become an essential component of our society. The RSA cryptosystem, a widely used cryptosystem in our current online system, is a safe but time-consuming method. Euler's totient function, in the modular arithmetic that the RSA is based on, plays an important role and has strong potential to be applied with other cryptosystems to make innovations on the RSA. This paper will introduce the potential proposed risk of hacking the RSA cryptosystem using Euler's totient function, and how this function can be applied itself or with other ciphers to create more optimized algorithms and methods based on the RSA cryptosystem. The analyzed results show that the current innovations made to the RSA cryptosystems are rarely balanced on time complexity and security levels and need improvements under most circumstances.

Keywords: RSA cryptosystem, modular arithmetic, Euler's totient function.

1. Introduction

Cryptography plays an essential role in our modern society due to the need for secure information transfer. Encryption and decryption are the two fundamental processes of cryptography. To enhance the security level of information, the RSA cryptosystem was introduced by Ron Rivest, Adi Shamir, and Leonard Adleman in 1977. This system is widely used in online transactions and web browsing. However, the RSA is very time-consuming, and its time complexity needs to be reduced while maintaining security. The challenge lies in finding a balance between complexity and security, especially with the increasing risk of attacks due to advances in technology. Researchers have explored the potential risks and benefits of using Euler's totient function, which is part of the RSA's original modular arithmetic. This paper discusses the innovation of the RSA cryptosystem using this function and its combination with other ciphers to improve efficiency while maintaining security.

2. Related concepts

Euler's Totient Function: $\varphi(n)$ refers to the count of the integers k in the range $1 \leq k \leq n$ in which $\gcd(k,n)=1$;

Asymmetric Cryptography: A cryptosystem that involves two keys: public key for encryption, and private key for decryption.

© 2023 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

Symmetric Cryptography: A cryptosystem that has the same cryptographic keys for both encryption and decryption. The two keys are not necessarily identical, but the transformation between them is straightforward.

RSA Cryptosystem: It is also known as Rivest–Shamir–Adleman, is a public-key cryptosystem used widely in data transmission. It is one of the oldest cryptosystems, established in 1977 [1]. It is an asymmetric cryptosystem. Mathematically explaining, let p, q be two individual large primes, as well as the modulus of n; e and d are the encryption exponent and decryption exponent, respectively, which satisfy the relation ed≡1 mod φ(n). The pair (n,e) is the public key, and d is the private key.

Caesar Cipher: The basic type of substitution cipher used by Julius Caesar. It rotates the letters in the plaintext by a fixed number under a certain modulo of spaces [2].

Hill Cipher: A cryptosystem in which each letter from A to Z is represented by a number modulo m. To encrypt, the original plaintext is transformed into one or several n-component vectors. Then an invertible n x n matrix, also the cipher key, will multiply the n-component vector(s) and then modulo m. To decrypt, the obtained vector/matrix will be multiplied by the inverse of the cipher key [3].

Transposition Cipher: A method that mixes up the sequence of the characters in the plaintext without changing the characters themselves [4].

3. RSA

3.1. Potential risks of RSA

In the paper Properties of the Euler Totient Function Modulo 24 and Some of its Cryptographic Implications by R. Gorgui-Naguib and S. Dlay, published in 1988, it introduced some of the potentials of the integer 24 implied in the Euler's totient function in the RSA cryptosystem based on its unique properties [5].

They first proved that for any n=pq, with p, q being prime numbers larger than 3,

$$\varphi(n^2) \equiv \varphi(n)[24] \tag{1}$$

$$\varphi(n^2) \equiv n\varphi(n) \tag{2}$$

Then they wrote congruence (1) in its Diophantine equation form:

$$\varphi(n^2) \equiv 24x + \varphi(n); x = 1,2, \dots \tag{3}$$

Lastly, they equated (2) and (3) together, obtaining the result:

$$\varphi(n) = \frac{24x}{n-1}; x = 1,2 \dots [5] \tag{4}$$

They hence proved that there exists a definite structure of the Euler's Totient function $\varphi(n)$, therefore, in terms of the RSA, this structure increases the vulnerability of the cryptosystem from attacks. However, since the specific evaluation of factor x is still complicated, the specific impact remained unproven.

3.2. Determination of the true value

When attacking the RSA cryptosystem, the value of the Euler's totient function can be guessed, so based on the previous research, later in 1993, C.-K. Wu and X.-M. Wang, in the paper Determination of the true value of the Euler totient function in the RSA cryptosystem from a set of possibilities, based on the previous research, further considered that in order to discover some possibilities to attack the RSA cryptosystem, it is essential to determine the true value of the Euler's totient function [6]. They proposed a method that narrows down the range of this value: among all the possible values of $\varphi(n)$, only that satisfying $2^x \equiv 1[n]$ is the true value [6]. This suggests that there exists greater risk to the RSA cryptosystem, hence pushing the requirement for the complexity of the RSA to another level.

4. Innovations

4.1. Decreasing the computing complexity with multi-pass algorithm

In 2007, Rohit Pandharkar, Madhuri Joshi, and Nitin Narappanawar, developed a multiple pass algorithm upon the RSA cryptosystem using Euler's totient function [7]. The two parties first select a large prime p together (public key), then individually pick two large primes, m and M, with no common factor (p-1). Then they each solve the Diophantine Equations-finding an integer n so that m+n=(p-1)z+1, where z is any integer and an integer N so that M+N=(p-1) k, where k is any integer. Then, m and n are the private keys of the first party, and M and N are the private keys of the second. The first party selects its message x and the second party decides its secret number y, such that x<<p and y<<p. The process of transferring a message is:

First party: Compute $A \equiv x^{m}[p]$ and transmit A;

Second party: Compute $B \equiv A * x^{m}$ [p] and transmit B;

First party: Compute $C \equiv B * x^{m}$ [p] and transmit C;

Second party: Compute $D \equiv C * x^{m}$ [p] and this will be the original text.

In order to prove that D is the original text, they proved that: $D = y^{M+N}x^{m+n}$[p]=x; by Euler's Totient Theorem ($a^{\varphi(n)} \equiv 1[n]$).

The only two methods to attack this system are either knowing all four private keys, m, n, M, N, or in knowing all the two private keys, x and y. Since the core modulo function [a(mod p)] is a one-way function, this system is likely to be impossible to attack [7]. The biggest advantage of this method is that it can solve the key distribution problem. It can bring forth computational simplicity without compromising security levels. It is an efficient and applicable method derived from the RSA cryptosystem and Euler's totient function's intrinsic properties.

4.2. Involvement of an additional prime for key generation

In A Secure Cryptosystem by using Euler Totient Function and Modified RSA, by Sanjeev Kumar Mandal and A. R. Deepti in 2018, the complexity of the traditional RSA has been increased to better prevent possible crypto attacks [8]. Instead of imposing two large primes, p and q, their innovative algorithm involves three primes p, q, and r in their key transmission. To increase the key size, their modified RSA is generated by the following steps:

1. Choose 3 prime numbers, named p, q, and r.
2. Multiply p, q, and r to obtain the modulus N, denoted as n.
3. Select an exponent value E, where $1 < E < \phi(n)$.
4. Calculate ϕ(n) by multiplying (p-1), (q-1), and (r-1).
5. Use e, p, q, and r to compute the private exponent D.
6. The public key is (n, e), and the private key is (n, d).

The encryption is done by $C = m^{e}$ mod n, where m is the message that needs to be transmitted to the receiver and the output C is the resulting ciphertext.

The decryption is done by $M = c^{d}$ mod n [8].

They combined this modified RSA with the ASCII and operated in the binary system, giving relatively short time complexity, also high security as analyzed by various tools. This is a very successful approach.

4.3. Enhancing key generation with Hill cipher

Another way to increase the security of the RSA cryptosystem is to combine it with Hill Cipher.

In their 2020 paper, Properties of the Euler Totient Function Modulo 24 and Some of its Cryptographic Implications, Mohiuddin Ahmed and Md. Ashik Iqbal, combined the RSA cryptosystem with the Hill cipher to generate more keys [9]. The plaintext is first encrypted using normal Hill cipher, then the list of numbers generated will be encrypted again using the RSA cryptosystem.

Encryption:

A. The plaintext will be transformed in a i*j matrix P, then a random i*i matrix A will be chosen and multiply P to obtain the new i*j matrix AP.

B. Next, using the congruence number chosen to operate in AP, we obtain the new matrix E.

C. Chose two relatively large primes p, q, and nominate a possible integer value k and find the corresponding j such that kj≡1 mod $\varphi(n)$.

D. Each number in E will be encrypted by $m^k = C$ [n], and C is the finished code.

Decryption:

A. Using the value j, $c^j = m[n]$, we obtain the matrix/vector E

B. Multiply E by the inverse A and find the corresponding letter to each obtained number. This gives us the completed decoded text [9].

The advantage of this method is that since, the number of keys has increased—the first key matrix A will only be recognizable to the first and second parties, and the value j cannot be easily obtained from n and k(because p and q are two very large primes, it is very time-consuming to enumerate all the possible j), it is very difficult to decipher without knowing all the keys. Additionally, this unique combination is very hard for the third party to think of, thus further ensuring the safety of the transferred information.

4.4. Symmetric and transposition cipher

In 2019, A. P. Madushani and P. Ranasinghe, took a new approach in their A symmetric and a transposition cipher using Euler's totient function. They created a transposition cipher and symmetric cipher based on the RSA cryptosystem and the Euler's totient function [10].

The transposition cipher used the characteristic of Euler's totient function to assign each letter in the plaintext a new position in the string with the following steps:

Encryption:

Step 1: Ignore the spaces in the plaintext.

Step 2: Assign a number to position each letter as it appears in the plaintext.

Step 3: Determine the number of distinct letters, n and find Euler's totient function of n, φ(n).

Step 4: Choose m to be the total number of letters in the plaintext or the smallest odd number greater than that to satisfy the condition $\gcd(\phi(n), m) = 1$.

Step 5: Obtain the new positions q under modulo m such that, $q \equiv p \cdot \phi(n)[m]$.

Decryption: Reverse the encryption process with an inverse modulo m. Since we obtain $\gcd(\phi(n), m) = 1$, the existence of an inverse modulo m is guaranteed by the corollary. If x is the plaintext position that we need to determine it should satisfy, $\phi(n) \cdot x \equiv q[m]$.

The symmetric cipher they developed is an enhanced version of the previous transposition cipher. They also combined it with ASCII. The process includes the following steps:

Encrypting:

1. Group plaintext characters as odd/even based on their positional values.

2. Convert characters to decimal values.

3. Calculate φ(n) based on the number of characters in each group.

4. Identify the maximum decimal value for each group. If necessary, find the nearest prime greater than the maximum value.

5. Generate a new value q for each p using $q \equiv p \cdot \phi(n)[m]$.

6. Create random key streams for odd/even cases using ASCII symbols.

7. Add q and key streams to obtain the decimal value of each ciphertext character.

8. Convert ciphertext characters to ASCII symbols and combine odd/even groups to get the final ciphertext.

Decryption:

Reverse the encryption process with an inverse modulo m. Since we choose m to satisfy the condition.

$\gcd(\phi(n), m) = 1$. The existence of an inverse modulo m is guaranteed by the corollary. If the decimal values corresponding to plaintext characters that we need to determine are x, we shall solve, $\phi(n) \cdot x \equiv q[m]$ [10].

Yet these ciphers create a relatively simple and secure algorithm, which is quite difficult to figure out in transposition ciphers. Nevertheless, the use of ASCII does not seem to increase the security of their cipher since it is expressed in this cipher straightforwardly and ASCII is an open source, although ASCII provides convenience to their cipher to some extent. The biggest shortcoming of this approach is that it seems to be secure only when applied to large paragraphs, but in such instances, it would be relatively time-complex.

4.5. Involvement of Caesar Cipher

Inspired by the above research, Rajalaxmi Mishra and Jibendu Kumar Mantri, made a new algorithm in An Enhancement to Caesar Cipher using Euler Totient Function in 2022 [11]. Not only did they involve the position of the character in the plaintext, but also the line count and word length. The key K is generated by $\varphi(\text{line count})$. To encrypt, each character in the plaintext is added its value to its position in the line, the word length of the corresponding word, and the line number, then modulo 256. Then this ciphered character is added again to the key, then modulo 256. To decrypt, each character of the ciphered text would subtract the key from its value, then mod the 256. Then, from the computed value from the previous step, its position in the line, the word length of the corresponding word, and the line number are subtracted to get the original plain text [11]. They showed that this method balances the frequency of characters in the text file, therefore increasing the level of security. This method is more secure than the 2019 one because the characters do not appear as their original form in the text, and their frequencies are closed to be balanced, thus enhancing the difficulty for hackers to brutally attack. Furthermore, the position and word length are distinct for each word in the text, so that the overall pattern is hard to be observed. Nonetheless, the line count, position in line, and the word length are not encrypted. Hence, once the use of these three values is detected for even just one single character, the whole file can be decrypted in seconds.

5. Conclusion

The RSA cryptosystem is widely recognized as a secure method for encryption. However, its inefficiency presents a significant challenge. The optimization of RSA has the potential to lower its security level. To address this issue, innovative algorithms that leverage RSA's core function, Euler's totient function, can be developed. By combining this function with other ciphers, a balance can be struck between complexity and convenience. This paper presents several methods that vary in these two characteristics, allowing users to select the most appropriate method for their specific needs. The ultimate goal is to optimize the time complexity and key size of RSA while maintaining its security.

References

[1] Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. https://doi.org/10.21236/ada606588

[2] Computer Sciences. encyclopedia.com. (2018). Encyclopedia.com. Online: https://www.encyclopedia.com/social-sciences-and-law/law/crime-and-law-enforcement/cryptology-and-number-theory.

[3] Hill, L. S. (1929). Cryptography in an algebraic alphabet. The American Mathematical Monthly, 36(6), 306–312. DOI: https://doi.org/10.1080/00029890.1929.11986963

[4] Encyclopedia.com. (2023). Transposition Cipher. Encyclopedia.com. https://www.encyclopedia.com/computing/dictionaries-thesauruses-pictures-and-press-releases/transposition-cipher

[5] Gorgui-Naguib, R. N., & Dlay, S. S. (1988). Properties of the Euler totient function modulo 24 and some of its cryptographic implications. Lecture Notes in Computer Science, 267–274. DOI: https://doi.org/10.1007/3-540-45961-8_24

[6] Wu, C.-K., & Wang, X.-M. (1993). Determination of the true value of the Euler totient function in the RSA cryptosystem from a set of possibilities. Electronics Letters, 29(1), 84–85. DOI: https://doi.org/10.1049/el:19930055

[7] Pandharkar, R., Joshi, M., & Narappanawar, N. (2007). A Provably Secure Message Transfer System Using Euler's Totient Function . IADIS International Telecommunications, Networks and Systems, 110–114.

[8] Mandal, Dr & Deepti, A R. (2018). A Secure Cryptosystem by using Euler Totient Function and Modified RSA. International Journal of Scientific Research. 08, 1-07.

[9] Ahmed, M., & Iqbal, M. A. (2020). An execution of a mathematical example using Euler’s phi-function in Hill Chiper cryptosystem. International Journal of Material and Mathematical Sciences, 99–103.DOI: https://doi.org/10.34104/ijmms.020.0990103

[10] Madushani, A. P., & Ranasinghe, P. G. (2019). A symmetric and a transposition cipher using the Euler’s totient function. Ceylon Journal of Science, 48(4), 327. DOI: https://doi.org/10.4038/cjs.v48i4.7672

[11] Mishra, R., & Mantri, D. J. (2022). An enhancement to caesar cipher using Euler totient function. International Journal of Engineering and Advanced Technology, 11(3), 46–50. DOI: https://doi.org/10.35940/ijeat.c3363.0211322