Combination analysis of ElGamal algorithm and LUC algorithm in file security

H Mawengkang*, A F Siregar, and S Efendi

Departemen S2 Teknik Informatika, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara, Jl. Universitas No. 9-A, Kampus USU, Medan 20155, Indonesia

*annisa.fsir@gmail.com

Abstract. Cryptography is a science to keep information secure, including encryption process and description. There are four fundamental objectives of cryptography that are aspects of information security: confidentiality, data integrity, authentication and non-repudiation. The ElGamal algorithm is one of the most secure algorithms because it has a high security in algorithm complexity. The ElGamal algorithm gives different ciphertext each the plaintext is encrypted. The LUC algorithm is one of the variants of the asymmetry cryptography algorithm. The LUC algorithm uses two primes to generate public keys and secret keys. The LUC algorithm has much in similarity with the RSA algorithm but in the LUC algorithm the ranks function in RSA is replaced by the Lucas function. Super Encryption is a method of combining between the two algorithms that aims to get a stronger cipher making it very difficult to solve. The result of this study is the process time stated that the average time of plaintext encryption process with 10 characters is 7.33 milisecond while the mean time of plaintext encryption process with 100 characters is 134 milisecond. Then for the average time of encryption process ciphertext with 10 characters is 47.33milisecond mean time ciphertext encryption process with 100 characters is 186.33 milisecond.

1. Introduction

The rapid technological developments support need for access to data and information exchange can be done quickly using internet every day. Security is a big problem and secures very important data, so the data can’t be tapped or misused for illegal purposes to the detriment of others. The data submitted sometimes often contain important information data even very secret and must be kept safe. There are several ways and techniques are used to keep the security of a message confidential. One of them is cryptography where messages are disguised as encrypted messages. Cryptography is a science to keep information secure, including encryption process and description. There are four fundamental objectives of cryptography that are aspects of information security: confidentiality, data integrity, authentication and non-repudiation.

The power of cryptography is determined by the key used [1]. To resolve this problem by using ElGamal algorithm and LUC Algorithm. In an asymmetry algorithm or public key there is an ElGamal algorithm that has a high security algorithm complexity. The ElGamal algorithm is good to use because it gives ciphertext which is always different every time message encryption or plaintext [2]. ElGamal algorithm is a type of public key algorithm whose security is based on Discrete Logarithm Problem (DLP) [3].
The LUC algorithm is one of the variants of the asymmetry cryptography algorithm. The LUC algorithm was invented by Smith and Lennon in 1993 at the University of Auckland, Auckland, New Zealand. The LUC algorithm uses two prime numbers "p" and "q" to generate public key and private key. The LUC algorithm has similarities to the RSA algorithm but in the LUC algorithm the function of power in RSA is replaced by the function of Lucas whose function itself is introduced by F.E.A Lucas. Combined with the algorithm will lead to a combination of algorithms that can improve security so that the message more difficult to solve.

2. Method

The first encryption process will be done with LUC algorithm and then encrypted again using ElGamal algorithm. And then the first decryption process will be done with the Elgamal algorithm then LUC algorithm.

ElGamal algorithm consists of three processes, namely the process of forming the key, the process of encryption and the decryption process. This algorithm is a block cipher, which is doing the encryption process on the plaintext blocks and generate ciphertext blocks which then done the decryption process and the results are combined. In the framework of the operation to be followed, to simplify the statement, modulo q will be calculated without being accurately written, where q is taken at the key generation stage. For example $h = gx$ means $h = gx \mod q$ [4].

The quantities used in generating ElGamal's public key algorithm [5]:
1. Prime number, $p$ (not secret)
2. The random number $\alpha$ as the primitive root ($\alpha < p$) (not secret)
3. Random numbers $d$ ($d < p$) (secret)
4. Plaintext blocks $p$ (plaintext) (secret)
5. $C_1$ and $C_2$ (ciphertext) (not secret)  

Key Generate Process
Steps in key generate process:
1. Select a prime $p$ as the basis of the multiplication group $(Z*p,x)$
2. Select $\alpha$ as the primitive root of the group
3. Select $d$ which satisfies $1 \leq d \leq p - 2$
4. Calculate $\beta = \alpha^d \mod p$
Obtained public key $(p,\alpha, \beta)$, private key = $d$.

Encryption Process
Steps in encrypting the message:
1. Accept public key $(p,\alpha, \beta)$
2. Plaintext $m$ is arranged into blocks $m_1,m_2, \ldots, m_p - 1$ such that each block represents a value in the range 0 to $p - 1$
3. Change the message block value to ASCII value
4. Take an original number $r < p - 1$.
5. Calculate $C_1 = \alpha^r \mod p$
    Calculate $C_2 = P_1 \cdot \beta^r \mod p$
6. Send $C_1$ and $C_2$
Decryption Process
In decryption of the message used calculation $P = c_2 \cdot (c_1^d)^{-1} \mod p$. The calculation can be simplified by Fermat’s theorem:

$$P = c_2 \cdot c_1^{p-1-d} \mod p \quad (1)$$

To simplify calculation on each message block, it can be formulated by:

$$Z = c_1^{p-1-d} \mod p \quad (2)$$
$$P = c_2 \cdot Z \mod p \quad (3)$$

In 1993, Smith and Lennon introduced asymmetric cryptography algorithms based on Lucas function (Lucas Function). Lucas function can be described as follows [6]:

Enter the values of $a$ and $b$ into the polynomial root equations: $x^2 - P \cdot x + Q = 0$. Then $P = a + b$ and $Q = ab$. In general, the two linear repeats are:

$$U_n = (a^n - b^n) / (a - b) \quad (3)$$
$$V_n = a^n + b^n \quad (4)$$

There are two functions that can be derived from equations (3) and (4). That is:

$$U_{n+1} = P \cdot U_n - Q \cdot U_{n-1} \quad (5)$$
$$V_{n+1} = P \cdot V_n - Q \cdot V_{n-1} \quad (6)$$

Then in equation (6) can be derived,

$$V_n = P \cdot V_{n-1} - V_{n-2} \quad (7)$$

Where $n \geq 2$, $V_0 = 2$ and $V_1 = P$

The LUC algorithm is a public key algorithm in cryptography. There are three main parts in the LUC algorithm, [7]:

Key generating process with LUC algorithm:
1. Select two prime numbers, e.g. $p$ and $q$ where $p \neq q$
2. Calculate $n = p \times q$
3. Calculate $t = (p-1)*(q-1)*(p+1)*(q+1)$. 
4. Take a random number where the number $(1 < e < t)$ then the random number is an integer. Random numbers are symbolized by $e$.
5. Then calculate the value of $gcd(e, t) = 1$ or $e$ relative to $t$.
6. Calculate the value of $R(n)$ by the formula: $R(n) = LCM(p-1, q-1, p + 1, q + 1)$.
7. Calculate $d$ to get the result $e \cdot d \mod R(n) = 1$.
8. After being able to $d$ as the private key of the LUC algorithm, encryption and decryption are performed.

Encryption Process
1. Create plaintext you want to secure
2. Encrypt the plaintext by using the formula:
   - $V[0] = 2$
   - $V[1] = P$
   - If $i \geq 2$ then $V[i] = (P \cdot V[i-1] - V[i-2]) \mod n$

The encryption process generates ciphertext.
Decryption Process
1. Obtaining the ciphertext result from the encryption process, C = ciphertext
2. Then decrypt by using the formula:
   \[ V[0] = 2 \]
   \[ V[1] = C \]
   If \( i \geq 2 \) then
   \[ V[i] = (C \times V[i-1] - V[i-2]) \mod n \]
   The decryption process generates plaintext.

3. Result and Discussion
   The system was built using SharpDevelop 4.4 with the programming language is C#. This system is tested with Personal Computer with 1.0 GHz processor specification AMD C-70 APU, 2 GB Memory.
   The result of average time of process test in LUC algorithm with the number of characters 10, 50 and 100 is 7.33 ms, 42.33 ms and 134 ms while the result of average time of process test in ElGamal algorithm with number of characters 10, 50 and 100 is 47.33 ms, 576 ms and 1876.33 ms. The results can be illustrated in a graph of figure 1.

![Character Length Chart of Process Time](image)

**Figure 1.** Graph of plaintext length against the process time of LUC algorithm and Elgamal algorithm

Figure 1 shows a graph of plaintext length for the process time of the LUC algorithm, and ElGamal algorithm, where the length of the plaintext is directly proportional to the processing time. The longer character will be processed then the required process time will also be longer.

4. Conclusion
   - The test results at the time of the process obtained that the algorithm process time is directly proportional to the length of the plaintext character. This means that the
longer the plaintext used, the longer it will take for encryption and plaintext decryption.

- The results of the processing time LUC algorithm and the ElGamal algorithm show that the LUC algorithm processing time is faster than the ElGamal algorithm.
- ElGamal algorithm encryption result is longer than encryption of LUC algorithm even though the input is the same plaintext.

5. References
[1] Sadikin, R. 2012. *Kriptografi untuk Keamanan Jaringan dan Implementasinya dalam Bahasa Java*. Andi Offset: Yogyakarta.
[2] Singh, Rashmi dan Shiv Kumar. 2012. *Elgamal’s Algorithm in Cryptography*. International Journal of Scientific & Engineering Research. Vol. 3. Issue 12. December 2012:1-4.
[3] Rivest, Ronald L., Adi Shamir, and Leonard Adleman. 1978. "A method for obtaining digital signatures and public-key cryptosystems." *Communications of the ACM* 21.2: 120-126.
[4] Flonta, Stelian, and Liviu Miclea. 2008. "An Extension Of The El Gamal Encryption Algorithm." *Automation, Quality and Testing, Robotics, 2008. AQTR 2008. IEEE International Conference on*. Vol. 3. IEEE.
[5] Taufiq, Tamam, M., Dwiono, Wakhyu, & Hartanto, Tri., 2010. *Penerapan Algoritma Kriptografi Elgamal untuk Pengamanan File Citra*. Jurnal EECCIS: vol. 4, No. 2.
[6] Ali, Z., MD. 2012. *Implementation Of Parallel Algorithm For LUC Cryptosystems Based On Addition Chain By A Message Passing Interface*. Software Technology and Management Center, Faculty of Information Science and Technology, National University of Malaysia, 43600 Bangi, Selangor Darul Ehsan, Malaysia. Department of Computer Science, Faculty of Defence Science and Technology, National Defence University of Malaysia, Kem Sungai Besi, 57000 Kuala Lumpur, Malaysia
[7] Saputra, R., Yismianto, B. & Suhartono. 2006. *Kriptografi Teks Dengan Menggunakan Algoritma LUC*. Seminar Nasional SP MIPA, PP. 1-5.