Hybrid cryptosystem for image file using elgamal and double playfair cipher algorithm

S M Hardi, J T Tarigan and N Safrina
Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Jl. Doktor Mansyur No.9, Padang Bulan, Medan Baru, Sumatera Utara
20155

Email: nanda.safrina@usu.ac.id

Abstract. In this paper, we present an implementation of an image file encryption using hybrid cryptography. We chose ElGamal algorithm to perform asymmetric encryption and Double Playfair for the symmetric encryption. Our objective is to show that these algorithms are capable to encrypt an image file with an acceptable running time and encrypted file size while maintaining the level of security. The application was built using C# programming language and ran as a stand alone desktop application under Windows Operating System. Our test shows that the system is capable to encrypt an image with a resolution of 500x500 to a size of 976 kilobytes with an acceptable running time.

1. Introduction

There are many algorithms that can be used to perform an encryption of a file. These algorithms can be categorized into two types: symmetric and asymmetric. In symmetric encryption, both sender and receiver hold identical key which can be used to encrypt and decrypt the data. This fact may cause a problem since both party have to communicate to share the key, and this process may be eavesdropped by the middleman. Therefore, sender and receiver must have a secured channel to share the key, which is not possible most of the time.

Asymmetric encryption solves this problem by using two different keys. First introduced by Diffie et al. [1], asymmetric encryption uses public keys which distributed widely and private keys which is known only by the owner. Hence, unlike symmetric encryption, both parties do not require a secure channel to distribute the key. These keys often rely on mathematical problems with no efficient way to solve such as integer factorization, discrete algorithm, and elliptic curve. However, due to it needs to perform mathematical operations to perform encryption, the time to perform encryption heavily depends on the file size. Hence, asymmetric encryption is often not suitable to encrypt large files.

Since both encryptions have their own weakness, it is not possible to a single type of encryption to secure a data. Hybrid cryptosystem tries to solve this problem by mixing both algorithm to increase the security while maintaining the performance. The system uses symmetric encryption to encrypt the plain text and asymmetric to encrypt the symmetric key. Since the size symmetric key is usually small, there wouldn't be a significant time added to encrypt the key.

In this paper, we propose a hybrid cryptosystem that combines asymmetric encryption Elgamal and symmetric encryption Double Playfair Cipher. We chose image file as our plain text to see the performance of the algorithm in both time and space. Our objective is to measure the usability of the system and to see whether the combination of these two algorithms are applicable. The paper is...
structured as follow: the next section will discuss related previous works that can be used as reference. Later, we will go shortly describe the algorithms used in this paper. Section 4 will describe the design and implementation of the system, followed by test and results in section 5. In the last section, we will go through the conclusions and future works.

2. Related Works
Hybrid cryptosystem has been used in most security. It capability to increase security while maintaining security level is one of the reason of it's use in modern cryptography. One of the most commonly used cryptosystem that built based on hybrid method is Pretty-Good-Privacy. Developed and introduced by Phillip Zimmerman [2], this method has become a backbone in internet security [3]. The encryption process of PGP contains symmetric encryption, usually AES [4] or IDEA [5], to encrypt the data and asymmetric encryption, usually RSA [6], to encrypt the key for symmetric encryption. Additionally, PGP also contains a process to validate message authentication by using certificates and digital signatures. Currently, PGP is used in most computer security system such as Transport Layer Security (TLS) [7] and its predecessor, Secure Socket Layer (SSL) [8]. First defined in 2007 by Internet Engineering Task Force, OpenPGP has become the standard part of TSL [9].

Other than PGP, there are many research focused on hybrid cryptosystem. Some of the work focus on using other encryption algorithm and test its usability. Hazra et al. developed a cryptosystem by using Blowfish and Diffie-Hellman algorithm [10]. Dahake et al. developed a fingerprint based cryptosystem where the biometric fingerprint data is used for key generation [11].

3. Encryption Algorithm

3.1 Elgamal Algorithm
Elgamal is an asymmetric algorithm developed by Taher Elgamal and introduced firstly in 1985 [12]. The security of the algorithm relies on the difficulty of computing discrete logarithms over finite fields. It is based on key exchange algorithm proposed by Diffie-Hellman [1]. The implementation of Elgamal Algorithm is as follow: Let say Bob tries to send a message to Alice. Alice has to provide a public key to be used as encryption key. To generate this public key, Alice have to choose these values: a large prime number $P_A$ a primitive element $a_A$ modulo $P_A$, and a random integer $d_A$ where $2 \leq d_A \leq P_A - 2$. Using these values, Alice calculates $\beta_A$ where $\beta_A \equiv a_A^{d_A}(mod P_A)$. Alice then publish the public key $(P_A, a_A, \beta_A)$ and keep the private key $d_A$.

Bob as the sender of message $M$ where $M < P_A$ chooses as random integer $k$. Bob calculates then computes the encrypted message $(r, t)$ where $r \equiv a_A^k(mod P_A)$ and $t \equiv \beta_A^k M(mod P_A)$. The encrypted message $(r, t)$ is sent to Alice. Alice then decrypt the message by using the private key $d_A$ in the equation $tr^{-d_A} \equiv M \ mod \ P_A$. Should any eavesdropper intercept the message $(r, t)$, it will require him to know the value $d_A$. While it is possible to calculate the value by using the public key $(P_A, a_A, \beta_A)$, it would be too difficult to be practical since the algorithm suggest the use of a very large prime number (200-300 digits integer).

3.2 Double Playfair Chiper
Playfair Cipher is one of the earliest encryption algorithm and categorized as classical cipher. It was first introduced by Charles Wheatstone in 1854. The idea is to use a 5x5 table to substitute each character in the plain text. Each cell from the table contains one character from A to Z and is unique to each other. The characters are placed sequentially. However, the first set of the cell contains the character of the key used in the encryption. The table below shows the table for playfair cipher with the key "EXAMPLE". It is also important to notice that since there are 25 cell in the table, there will be one missing character on the table. In the table below, we assume the character 'i' and 'j' is interchangeable.
Table 1. Example of Playfair table with Key "EXAMPLE"

|   |   |   |   |   |
|---|---|---|---|---|
| E | X | A | M | P |
| L | B | C | D | F |
| G | H | I | K | N |
| O | Q | R | S | T |
| U | V | W | Y | Z |

The message is then split into digraph and replaced by the character on the table by following these rules:
1. If both letters are the same, add a predefined character, eg. "X" after the first letter and encrypt the new pair.
2. If the letters appear on the same row of the table, replace them with the letters to their immediate right respectively.
3. If the letters appear on the same column, replace them with the letter immediately below respectively.
4. If the letters are not on the same row or column, replace them with the letters on the same roe respectively but at the other pair of corners of the rectangle defined by the original pair. Notice that in this step, the order is important.

However, the behavior of playfair cipher that does not allow a double letter diagram eg. EE, makes it easy to detect by using cryptanalysis. Moreover, playfair digraph and its reverse will decrypt the same letter pattern in the plain text. This is a problem especially in English where reversed digraph (eg. RE and ER, DE and ED) is common (eg. RECEivER, DEpartED). The use of double playfair cipher, or usually called two-square cipher, solves this problem. The idea is simple; to use two tables and two keys instead of one. In addition, the substitution rules are modified as follow:
1. For a vertical two-square, the first character of both plaintext and cipher text uses the top matrix, while the second use the bottom matrix.
2. For a horizontal two-square, the first character of both digraphs uses the left matrix, while the second character uses the right.
3. Find the first letter in the digraph in the upper/left text matrix.

4. Implementation
As previously stated, we will simulate the encryption process by building a stand alone desktop application. To simplify the process, this process is done in a single computer; data communication will be simulated by creating files. The application is built using C# and run under Windows Operating System. To simplify the process, we also use a small digit prime number as key for asymmetric encryption. While this may vague the running time, our concern is only to see the usability of the algorithm. Figure 1 shows the process of encryption that simulated in our program. There are two role of user in the program: sender and receiver. The process begins when the sender sends a request to send an encrypted message. The receiver will generate a public/private key using ElGamal algorithm. The receiver sends the public key to the sender. The sender will generate a 2 key for the symmetric algorithm (as required by the double playfair algorithm). The sender encrypts the message using these 2 keys, and encrypt the symmetric keys using receiver's public key.
Figure 1. Hybrid Cryptosystem Simulated in the Program

Figure 2 below shows the first step of the encryption process. User is acting as a sender and required to enter 3 values as the parameter of the public and private key. The public key \((P_A, \alpha_A, \beta_A)\) will be kept saved as a file and loaded later upon decryption.

![Figure 2. Input Form for Asymmetric Generator](image)

In the next form, user will be asked to choose the file that will be encrypted. User will also need to enter two keys for the double play fair algorithm. When all the required fields are done, the program will encrypt the image and save it as an encrypted file format. Figure 3 shows the encryption form described previously.
Figure 3. Input Form for Encryption Process

Acting as a receiver, the user will be presented with a different form, as shown as figure 4. Unlike the previous one, in this form the user does not have to enter the symmetric key. However, the user have to choose the file that contains the encrypted key and enter the receiver's private key in the input form. If the encryption is successfully performed, the same image will be rendered on the left part of the form as shown as the figure below.

Figure 4. Input Form for Decryption Process

It is also important to know that the encrypted file size is significantly larger than the original file. Our tests show the size is increased averagely 6 to 8 times. As an example, an image file with a resolution 400x400 is 105 KB before the encryption and 625 KB after the decryption. This is however acceptable considering the image is rendered per pixel.

5. Conclusion and Future Work

We have successfully build a cryptosystem application that allows the use ElGamal and Double Playfair as the encryption algorithms. The implementation shows that the system has successfully perform an encryption and decryption process when the correct input has been entered. However, due to the nature of playfair as a classic encryption, it is better to perform the symmetric encryption using a more advanced symmetric algorithm such as AES.
Acknowledgments
The authors gratefully thank to all who have supported in the completion of this research. And to the Universitas Sumatera Utara that has given me the opportunity to publish this paper.

References
[1] Diffie W and Hellman M 1976 New Directions in Cryptography IEEE Transactions on Information Theory 22 644
[2] Zimmermann P R 1996 The Official PGP Users’s Guide (Massachusetts Ave: The MIT Press)
[3] Callas J, Donnerhacke L, Finney H, Shaw D and Thayer R 2007 OpenPGP Message Format https://tools.ietf.org/html/rfc4880
[4] NIST 2001 Advanced Encryption Standard Federal Information Processing Standard Publications 197 http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf
[5] Rivest R L, Shamir A and Adleman L 1977 A Method for Obtaining Digital Signatures and Public-Key Cryptosystems (Massachusetts Ave: USA) http://people.csail.mit.edu/rivest/Rsaper.pdf
[6] Lai X 1992 On the Design and Security of Block Ciphers (Zurich: Hartung-Gorre)
[7] Dierks T and Rescorla E 2008 The Transport Layer Security (TLS) Protocol Version 1.2 https://tools.ietf.org/html/rfc5246
[8] Freier A, Karlton P and Kocher P 2011 The Secure Sockets Layer (SSL) Protocol Version 3.0 https://tools.ietf.org/html/rfc6101
[9] Mavrogiannopoulus N 2007 Using OpenPGP Keys for Transport Layer Security (TLS) Authentication https://tools.ietf.org/html/rfc5081
[10] Hazra T K, Mahato A, Mandal A and Chakraborty A K 2017 A Hybrid Cryptosystem of Image and Text Files Using Blowfish and Diffie-Hellman Techniques Proc. Industrial Automation and Electromechanical Engineering Conf. (Bangkok: Thailand) p 137
[11] Dahake P and Nimbhorkar S 2015 Hybrid Cryptosystem for Maintaining Image Integrity Using Biometric Fingerprint Proc. Int. Conf. on Pervasive Computing (Pune: India) p 1031