Advances in Science, Technology and Engineering Systems Journal Vol. 2, No. 5, 31-34 (2017)

www.astesj.com

2nd International Conference of Computer, Environment, Social Science, Health Science, Agriculture & Technology (ICEST) 2017 5-7 May 2017, Medan, Indonesia

ASTESJ

ISSN: 2415-6698

A Theoretical and Experimental Comparison of One Time Pad Cryptography using Key and Plaintext Insertion and Transposition (KPIT) and Key Coloumnar Transposition (KCT) Method

Pryo Utomo*[,1], Nadia Widari Nasution[1], Arisman[1], Rahmat Widia Sembiring[2]

[1]*School of Informatics Engineering, Univesritas Sumatera Utara, Medan, 20155, Indonesia*

[2]*Department of Informatics Engineering, Univesritas Sumatera Utara, Medan, 20155, Indonesia*

Email; utomopryo@gmail.com, nadianwidari@gmail.com, arisman.pili@gmail.com, rahmatwsphd@gmail.com

ARTICLE INFO

Article history:
Received: 17 March, 2017
Accepted: 20 April, 2017
Online: 13 June, 2017

Keywords:
cryptography
Key insertion
Exclusive-OR

ABSTRACT

One Time Pad (OTP) is a cryptographic algorithm that is quite easy to be implemented. This algorithm works by converting plaintext and key into decimal then converting into binary number and calculating Exclusive-OR logic. In this paper, the authors try to make the comparison of OTP cryptography using KPI and KCT so that the ciphertext will be generated more difficult to be known. In the Key and Plaintext Insertion (KPI) Method, we modify the OTP algorithm by adding the key insertion in the plaintext that has been splitted. Meanwhile in the Key Coloumnar Transposition (KCT) Method, we modify the OTP algorithm by dividing the key into some parts in matrix of rows and coloumns. Implementation of the algorithms using PHP programming language.

1. Introduction

In terms about security especially Information Technology (IT), security is very important to be applied. There are many methods that used for this security. But there is no security method that guaranteed reliability. All of them must have weakness. So, for decreasing every weakness we need to make experiment about security method. One of this method is enhancement of cryptography algorithm.

In cryptography are known some algorithms, one of them is One Time Pad (OTP) Algorithm. OTP includes flow ciphers. Discovered by Major J Maugborne and G Vernam in 1971. Each key is only used for a single message.

The encryption process with One Time Pad (OTP) Algorithm or One Time Pad Cryptography is essentially an Exclusive-OR algorithm that is consistent with a less complicated implementation [4].

In transposition cipher, plaintext is similar, but its order was changed. On the other words, the algorithm do transposition of the character set in the text. The other name for this method is permutation because of doing the transposition each character in the text is similar with make permutation the characters.

In this paper, modified the One Time Pad (OTP) algorithm by splitting each binary plaintext into 4 sections and then inserting a key between separate plaintexts.

In transposition process, there is a key in which is mostly used in order to make the cryptanalist be difficult to guess the ciphertext. Transposition would be solution before doing insertion process. Transposition method in this process using matrix [4x2].

After that, Exclusive-OR process will be done so that the ciphertext will be more difficult to be known.

2. Materials and Methods

The underlying mathematical basis of the process of encryption and decryption is the relation between two sets, plaintext and ciphertext. Encryption and decryption are functions of transformation between the sets.

In principle, cryptography has 4 main components:

1.Plaintext : Readable messages
2.Ciphertext : Random messages that can not be read
3. Key : The key that's doing cryptographic
4. Technique Algorithm: Methods for encryption and decryption

*Corresponding Author: Pryo Utomo, School of Informatics Engineering, Univesritas Sumatera Utara, Medan, 20155, Indonesia
Email: utomopryo@gmail.com

https://dx.doi.org/10.25046/aj020506

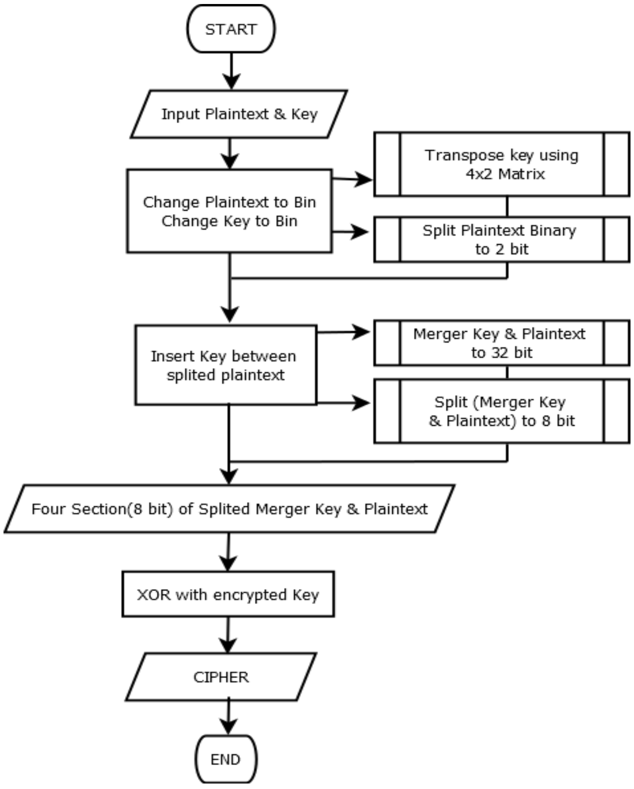


Figure 1. Flowchart of Encrypt Process

In cryptography, the main process used is encryption and decryption. Encryption is formed based on an algorithm that will randomize an information into a form that cannot be read or cannot be seen. Decryption is a process with the same algorithm to return random information to its original form. The algorithm used should consist of carefully planned arrangement of procedures that must effectively produce an encrypted form that cannot be returned by someone, even if they have the same algorithm.

Encryption is the process used to encode plaintext by converting plaintext into ciphertext. While decryption is the reverse process, which is to change the ciphertext to plaintext [2].

Formula of the encrypt and decrypt process as follows:

$$Encrypt:$$

$$pla\operatorname{int}ext \oplus key = ciphertext$$

$$Decrypt:$$

$$ciphertext \oplus key = pla\operatorname{int}ext$$

Here the flowcharts of encrypt and decrypt process:

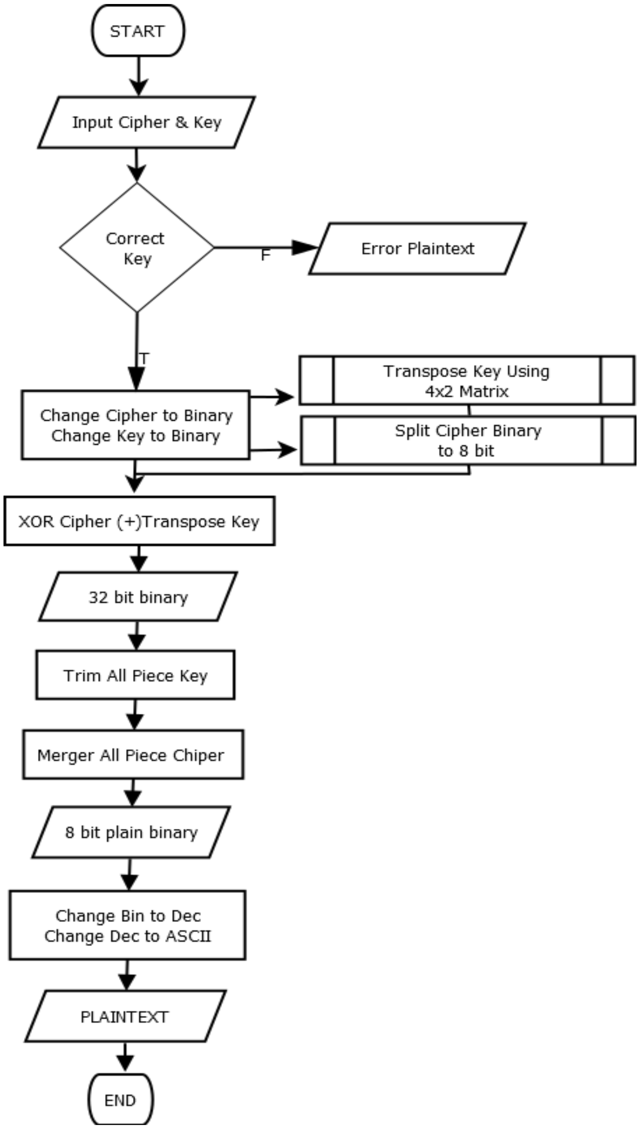


Figure 2. Flowchart of Decrypt Process

Table 1. Encryption Proces

a	*PLAINTEXT*=K	01001011				**FLOW** 8 bit binary
b	=(a1/4)	01	00	10	11	Split 8 bit binary@2bit
c	*KEY*=T=	01010100				8 bit of key
d	*TRANSPOSE Key*	00001110				Transpose Key using 4x2 Matrix
e	*MIX*(**d**) *to* (**b**)	0100001110	0000001110	1000001110	11	Insert key to splitedplaintextupto 32 bit
f	*MERGER*(**e**)	01000011100000001110100000111011				unallocated merger result
g	=(f/4)	01000011	10000000	11101000	00111011	Splitedunallocated merger result @ 8 bitbinary
h	XOR KEY(**d**) with(**g**)	01001101	10001110	11100110	00110101	XOR result from d1 with f1
i	*DEC*(**h**)	77	142	230	53	Change to dec
j	*ASCII*(**i**)	M	Ä	µ	5	Cipher

Tabel 2. Decryption Process

a	*CIPHER* K=	**MÄµ5**				**FLOW** Enkripsi of "K"
b	*DEC(***a***)*	77	142	230	53	Change to dec
c	*BINARY CIPHER(***a***)*	01001101	10001110	11100110	00110101	XOR key and plain before it
d	*KEY T*	00001110	00001110	00001110	00001110	Key Transpose
e	XOR KEY(**d**) to CIPHER(**c**)	01000011	10000000	11101000	00111011	Xor result
f	*MERGER(***e***)*	01000011100000001110100000111011				Unallocated 32 bit binary
g	=TRIM(**e**)	01	00	10	11	Trim key result
h	*=CONC(***g***)*	01001011				Mix Trim key result
i	*=DEC(***h***)*	75				Change to dec
e	*ASCII (***i***)*	K				Plaintext

3. Results and Discussions

One Time Pad (OTP) in the encryption and decryption process will perform XOR logic on plaintext-key and chipertex-key. The ciphertextthat is generated by OTP has the exact character length of plaintext and key so it opens up opportunities for cryptanalysis to guess key and plaintext.

But in this case, the authors perform the XOR process after the insertion of key into individual plaintext characters. So the ciphertext will be generated more difficult to be guessed. For more details here is a description of how the KPI algorithm works:

3.1. Encryption Process

We must determine the plain text to be encrypted.

According both of the tables, ciphertext was obtained from plaintext **"K"** with *key* **"T"** is " MÄµ5" and plaintext from *cipher*" MÄµ5**" with** *key* **"T" is "K".**

4. Implementation of The Algorithm

To find out the success of this algorithm, we will try to implementation this algorithm at PHP Language. This application is a program that do encryption and decryption process from the text. On this application User will be requested for entering a plain text, algorithm will processing it and make a result of chiper text up to four times from plain text.

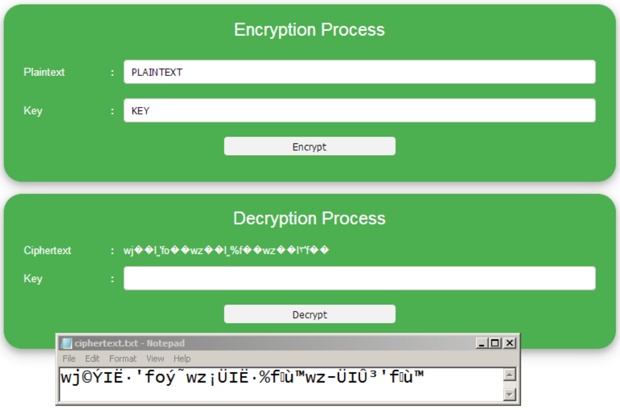
For the fruitfulness test of this algorithm, we will encryption a text as follows:

"PLAINTEXT"

The Following is view of encryption process result using the key "KEY".

5. Conclusions

According the Implementation, we have concluded that the modification of *One Time Pad* (OTP). Algorithm using insertion key On the splitting plain that the authors develop In this paper can working properly as an alternative cryptographic algorithm, so this algorithm able to improve the capability One Time Pad (OTP) algorithm for today and future. Based using this algorithm we can see the Compare result from the encryption process with a text as follows: "K" and the key "T".



Encryption Process

Plaintext : PLAINTEXT

Key : KEY

Decryption Process

Ciphertext : wj��l_'fo��wz��l_%f��wz��l٣'f��

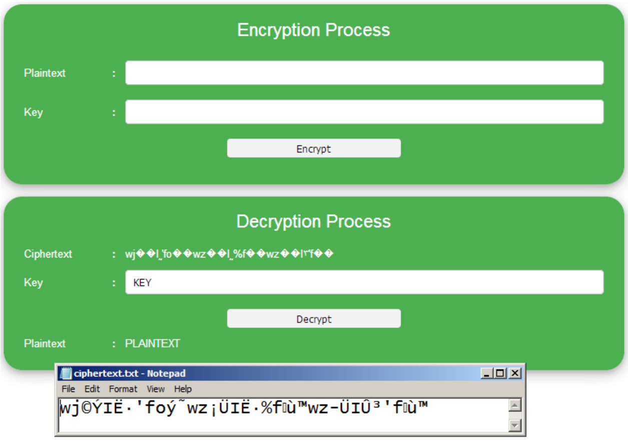
Key :

ciphertext.txt - Notepad

wj©ÝIË·'foý˜wz¡ÜIË·%f▯ù™wz-ÜIÛ³'f▯ù™

Figure 2. Encryption Interface

The view result of decryption process using the key "KEY" as follows:



Encryption Process

Plaintext :

Key :

Decryption Process

Ciphertext : wj��l_'fo��wz��l_%f��wz��l٣'f��

Key : KEY

Plaintext : PLAINTEXT

ciphertext.txt - Notepad

wj©ÝIË·'foý˜wz¡ÜIË·%f▯ù™wz-ÜIÛ³'f▯ù™

Figure 3. Decryption Interface

Table 3.0 Cipher Comparation, OTP with OTP Enhancementt

CHAR	OTP	OTP with KPI	OTP with KCT
Plaintext	K	K	K
Key	T	T	T
Ciphertext	▼	MÄµ5	☺**Q."**

References

[1] Chen, Z and Xu, J, One-Time-Pads Encryption in the Tile Assembly Model, IEEE Explorer and Conference Proceeding, 2008.

[2] Kurniawan, Y, Criptography: Internet Security and Comunnication System, Informatika Bandung, 2004.

[3] Mezaal, Y. S, OTP Encryption Enhancement Based on Logical Operations, IEEE Explorer and Conference Proceeding, 2016.

[4] Saragih, F. R, Using Chrtography One Time Pad (AlgoritmaVernam) in Information Security,2008.

[5] Pryo Utomo, Enhancement of OTP Cryptography Using Key and Plaintext Insertion, KPI Journal, 2017.