Jurnal Sains dan Teknologi
Volume 6 No. 1 | Mei 2024 |pp: 52-57
E-ISSN : 2714-8661 | DOI : https://doi.org/10.55338/saintek.v6i1.3190

Enhancing Text Messages with a Combination of Vigenère Cipher and One Time Pad Using Random Key LFSR

Anzas Ibezato Zalukhu[1], Zulham Sitorus[2*], Suhardiansyah[3], Nadya Septiani[4]

[1,2,3,4] Master of Information Technology, University Pembangunan Pancabudi, Medan, Indonesia
Author's correspondence email: [2]zulhamsitorus@dosen.pancabudi.ac.id

Abstract− In today's global era, the development of information systems is rapidly advancing and becoming increasingly sophisticated, with nearly all elements utilizing information and communication technology in their daily activities. One frequent activity is the transmission or exchange of data over the internet. However, this advancement also raises concerns about the security of messages or data being sent. To mitigate the risk of message misuse, cryptographic techniques can be used to maintain data or message confidentiality by encrypting the information before transmission. This research aims to combine the Vigenère cipher algorithm with a one-time pad using a random key generated by a linear feedback shift register (LFSR) method to enhance the security of text messages. The research methodology involves generating a public key for the One-Time Pad algorithm using LFSR. The encryption process is initially performed with the Vigenère cipher, and the resulting encrypted message is further encrypted using the one-time pad with a key generated by the LFSR method. This algorithm is implemented using the Visual Basic programming language. The research findings indicate that the combination of the two algorithms, with the random key generated by the LFSR for the One-Time Pad, is capable of enhancing text message security by producing random and unique ciphertexts. By using Modulo 256 and ASCII conversion, random ciphertexts can be generated, thereby reducing the likelihood of message breaches. Additionally, this research provides further insights into the process of text message encryption and decryption.

Keywords: Cryptography, Algorithm, Linear Feedback Shift Register, One Time Pad, Vigenère Cipher.

I. INTRODUCTION

In the current global era, the development of information systems is advancing rapidly and sophisticatedly, with nearly every sector utilizing information and communication technology in daily activities. The transmission or exchange of data has become a common activity in the world of information technology. However, this also raises concerns about the security of messages or data being transmitted, especially with the increasing risk of information theft on the internet. Information or data theft on the internet often occurs during message transmission, where third parties have already obtained the information sent by the sender to the receiver. These third parties will attempt to read the sent information. In safeguarding our data, we need to use information security, aimed at reducing highly detrimental risks. This threat arises from the misuse of such messages.

Efforts to reduce the misuse of these messages can be carried out by altering the content of the message or information sent to the receiver. So that the information received by the receiver may differ from the message sent by the sender, but essentially the message remains the same, albeit disguised by the sender. Data exchange or data transmission can be secured using cryptographic techniques. Cryptography is the art and science of protecting data transmission by converting it into a specific code and having only one key to convert that code back, which functions to maintain the confidentiality of data or messages [1].

The previous authors in their research explained the generation of a key linear feedback shift register in the modified Hill cipher algorithm using convert between bases [2]. Sulaiman's research further continued to show that the linear feedback shift register method can be used for random key generation in the one-time pad algorithm. Both of these previous researchers tend to explain the process of encryption using one algorithm with random key linear feedback shift register (LFSR) [3]. Method from the previous research titled "Combination of Vigenere Cipher and One Time Pad Algorithms to Secure Text Data," a study on message encryption with two algorithms in general was conducted, where the keys for both algorithms were not randomized [4].

However, here the author will attempt to conduct research by combining the Vigenere cipher algorithm and the one-time pad algorithm with random key generation using the linear feedback shift register (LFSR) method in the one-time pad algorithm. The use of random key generation in the one-time pad algorithm can strengthen the security of subsequent keys because this LFSR is used to automatically create a public key, and with the presence of this LFSR, the key used in the one-time pad will be automatically generated from the plaintext. The encryption process of this combination works by generating plaintext using the linear feedback shift register (LFSR) method, the resulting plaintext generation will be used as the key for the one-time pad algorithm. After obtaining both algorithm keys, the predetermined plaintext is encrypted. The first encryption process is carried out using the Vigenere cipher

Jurnal Sains dan Teknologi
Volume 6 No. 1 | Mei 2024 |pp: 52-57
E-ISSN : 2714-8661 | DOI : https://doi.org/10.55338/saintek.v6i1.3190

algorithm and then encrypted again using the one-time pad algorithm.

II. METHODOLOGY

In this section, the research procedure will be explained. The method in this research is carried out through several stages. Broadly speaking, it can be outlined as follows.

A. Cryptography

Cryptography is the art and science of protecting data transmission by converting original characters into specific codes and only intended for individuals who possess a predetermined key to revert the code back into different characters, serving the purpose of maintaining data or message confidentiality [5][6]. There are four (4) main components of cryptography:

a) Plaintext, which is a message readable by the public.
b) Ciphertext, which is a random message or random code that cannot be read.
c) Key, which is a key to perform cryptographic algorithm techniques.
d) Algorithm, which is a method for performing encryption and decryption processes.

The Objectives of Cryptography There are four fundamental objectives of cryptography, which are aspects of information security[7][8]:

a) Confidentiality
Confidentiality is a service used to keep the content of information hidden from anyone except those who have the authority or the secret key to decrypt the information.
b) Data Integrity
Integrity is related to the protection of data from unauthorized changes. To maintain data integrity, the system must have the ability to detect manipulation by unauthorized parties, including insertion, deletion, and substitution of data within the actual data.
c) Authentication
Authentication is concerned with identification and recognition, both of the system as a whole and the information itself.
d) Non-repudiation
Non-repudiation is the effort to prevent the denial of sending or creating information by the sender or creator.

B. Symetric Key

Symetric key is an algorithm that can be referred to as conventional algorithm. In theory, a symmetric key is an algorithm that uses the same key for both encryption and decryption processes. This key is also often referred to as a classical algorithm [9].. Examples of symmetric key algorithms include the Vigenere cipher algorithm and the one-time pad algorithm[10].

C. Vigenère Cipher

The Vigenère Cipher algorithm is a part of polyalphabetic cryptography, first discovered in 1586 by a French diplomat named Blaise de Vigenère (1523-1596). The Vigenère cipher utilizes a standard Vigenère table in encrypting messages. The table used consists of a simple 26-letter alphabetical table starting from the letter A to Z [11].

Table 1. Recta Vigenère Cipher Table

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

Encryption formula for Vigenère Cipher:
Ci = (Pi + Ki) mod 26 (1)
or
Ci = (Pi + Ki) - 26, if the sum of plaintext and key exceeds 26 (2)
Decryption formula for Vigenère Cipher:
Pi = (Ci - Ki) mod 26 (3)
or
Pi = (Ci - Ki) + 26, if the subtraction of ciphertext with key (-) (4)
Explanation:
Ci = Decimal value of the i-th ciphertext character
Pi = Decimal value of the i-th plaintext character
Ki = Decimal value of the i-th key character
Decimal value of characters: A=0 B=1 C=2…Z=25

D. XOR

XOR (Exclusive-OR) is a logic gate operation symbolized by the symbol "⊕". The XOR operation will result in "0" (zero) if both values to be XORed are the same and will result in "1" (one) if XORing two bits with different values.

Table 2. XOR rule

A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

E. Linear Feedback Shift Register (LFSR)

Jurnal Sains dan Teknologi
Volume 6 No. 1 | Mei 2024 |pp: 52-57
E-ISSN : 2714-8661 | DOI : https://doi.org/10.55338/saintek.v6i1.3190

Linear Feedback Shift Register (LFSR) is an algorithm that can be applied to generate a sequence of binary numbers randomly for the key generation process in cryptography [12]. Linear Feedback Shift Register is a shifting register with a certain number of shifts, the output result is selected and added modulo 2 (two) then fed back to the first input register at every clock cycle [13]. Linear Feedback Shift Register uses the operation of a feedback shift register to obtain random numbers. The above feed shift register consists of two parts, namely:

a) Shift register, which is a sequence of bits (bn, bn-i, ..., b3, b2, b1) with a length of n (also called an n-bit shift register).
b) Feedback function, which receives input from the shift register and returns the function value back to the sequence or array of registers.

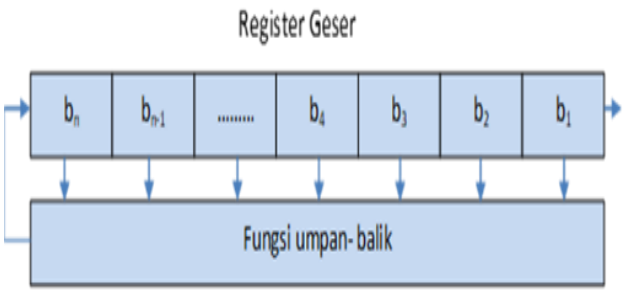

Figure 1. Parts of LFSR

For each required bit, all bits inside the register are shifted 1 bit to the right. The leftmost bit (bn) is a function of the other bits inside the shift register. The rightmost bit (b1) that will be shifted 1 bit becomes the output of the register. The shift register period is the length of the output sequence before it repeats.

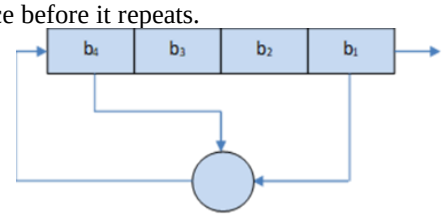

Figure 2. LFSR 4-Bit

Figure 2. is an example of a 4-bit LFSR, where the feedback function XORs b_1 and b_4 and stores the result in b4. The linear feedback shift register process is as follows:

1. B1 to B4 are filled with predetermined bits.
2. In the first step, b1 and b4 are XORed.
3. B1-B4 are shifted to the right by one bit.
4. The first bit becomes the output.
5. The XOR result between b1 and b4 (before shifting) is input into b4.

To obtain an LFSR with a maximum period, the feedback function used must be a primitive polynomial modulo 2. One example of a polynomial is $x^4 + x + 1$. The degree of the polynomial is the length of the shift register. A primitive polynomial with a degree is an irreducible polynomial that divides **$x_m + 1$ with m = 2n-1**, but does not divide **$X_d + 1$** for any d that divides 2n-1.

Generating pseudo-random numbers can be done using a Linear Feedback Shift Register (LFSR) by determining two random numbers first, $\mathbf{e^1}$ and $\mathbf{e^2}$ [3]. The LFSR can be formulated according to Sulaiman's research (2020, p. 172) as follows: "LFSR can be formulated as:

$$\mathbf{e^{i+n+1} = a^1e^{i+1} + a^2e^{1+2}+,\ldots\ldots\ldots+a^ne^{i+n} \bmod 26} \quad (5)$$

$$\mathbf{e^{1+3} = a^1e^{i+1} + a^2e^{1+2} \bmod 26} \quad (6)$$

$$\mathbf{a^1 = 1 \qquad a^2 = 2} \quad (7)$$

$$\mathbf{e^{1+3} = e^{i+1} + 2e^{i+2} \bmod 26} \quad (8)$$

An example of generating numbers with a linear feedback shift register is to determine two random numbers for e1 and $\mathbf{e^2}$. For example, $\mathbf{e^1}$ = 5 and e^2 = 10, then to find $\mathbf{e^3} = \mathbf{e^1 + 2 * e^2 \bmod 26}$ and so on. An example of generating 4 numbers is as follows:

$$\mathbf{e1 = 5} \quad (9)$$

$$\mathbf{e2 = 10} \quad (10)$$

$$\mathbf{e3 = 5 + (2*10) \bmod 26 = 25 \bmod 26} \quad (11)$$

$$\mathbf{e4 = 10 + (2*25) \bmod 26 = 8 \bmod 26} \quad (12)$$

The result obtained from the LFSR calculation: **5, 10, 25, 8**

F. One Time Pad (OTP)

One-time pad is an example of a cryptographic method with symmetric algorithm type, as the key used for encryption process is the same as the key used for decryption process. One-time pad is also claimed to be a perfect algorithm. OTP (pad = notebook paper) contains a series of randomly generated key characters [14]. In the one-time pad algorithm, the length of the key for the encryption process is equal to the length of the plaintext [15].

Formula for the OTP algorithm:

Encryption: Ci = (Pi + Ki) mod 26 (13)

Decryption: Pi = (Ci-Ki) mod 26 (14)

Explanation:
Ci = Decimal value of the i-th ciphertext character
Pi = Decimal value of the i-th plaintext character
Ki = Decimal value of the i-th key character
Decimal value of characters: A=0 B=1 C=2….Z=25

G. Research Flow

This research begins with a series of general stages in the research process, consisting of a literature review stage to be used as references or sources in this research, aimed at strengthening the problem and theoretical basis in conducting the research. Literature reviews are obtained from reading books, journals, articles, and relevant websites on the internet related to the research issue. After conducting the literature review, the researcher identifies the problems to be studied as the research object. The results and outputs obtained are used to draw conclusions from the research findings. The research flow and design can be seen in Figure 3 below.

Jurnal Sains dan Teknologi
Volume 6 No. 1 | Mei 2024 |pp: 52-57
E-ISSN : 2714-8661 | DOI : https://doi.org/10.55338/saintek.v6i1.3190

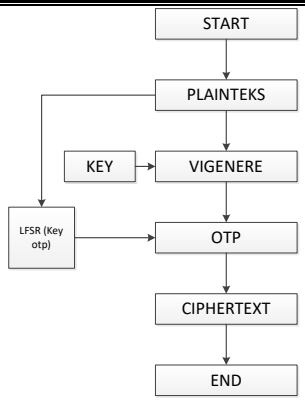

Figure 3. Encryption Process

In Figure 3 above, we can see 2 (two) algorithms, namely the Vigenere cipher algorithm and the one-time pad algorithm. In the one-time pad algorithm, the key is generated from the plaintext using the LFSR method, which functions to encrypt the plaintext into ciphertext.

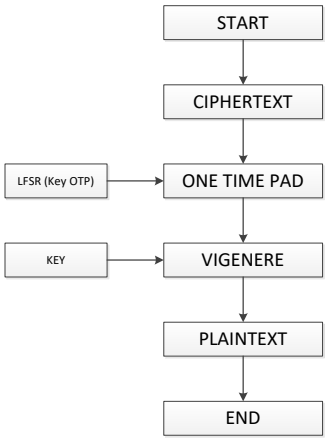

Figure 4. Desryption Process

In Figure 4, we can see the decryption process of the combination of the two algorithms, where the ciphertext is decrypted using the one-time pad algorithm with the key generated from LFSR, then continued with the Vigenere cipher algorithm with the initial key and will return to the original plaintext.

H. Implementation of Methodology

The methodology used in this research involves utilizing the Vinegere Cipher, LFSR, and One Time Pad algorithms.The first step involves generating keys using LFSR. The plaintext chosen by the author is "MTIUNPAB" with the Vigenere Cipher key "ANZAS," which will be randomized 8 times to produce the OTP key. The plaintext "MTIUNPAB" is randomized 8 times (period).

a) Convert the plaintext "MTIUNPAB" into binary:
01001101010101000100100101010101010011100101000001000000101000010
b) Randomize the Plaintext with LFSR 8 times (period):

- **Period 1 =**
00100110101010100010010010101010101001110010100000100000101000001
- **Period 2 =**
10010011010101010001001001010101010100111001010000010000010100000
- **Period 3 =**
11001001101010101000100100101010101010011100101000001000000101000
- **Period 4 =**
11100100110101010100010010010101010101001110010100000100000101000
- **Period 5 =**
11110010011010101010001001001010101010100111001010000010000010100
- **Period 6 =**
11111001001101010101000100100101010101010011100101000001000000101
- **Period 7 =**
01111100100110101010100010010010101010101001110010100000100000010
- **Period 8 =**
00111110010011010101010001001001010101010100111001010000010000001

Separation of Binary in period 8:
Key 1 = 00111110 = 62
Key 2 = 01001101 = 77
Key 3 = 01010100 = 84
Key 4 = 01001001 = 73
Key 5 = 01010101 = 85
Key 6 = 01001110 = 78
Key 7 = 01010000 = 80
Key 8 = 01000001 = 65

LFSR Result: **62 77 84 73 85 78 80 65**

c) Encryption Process
Vigenere encryption formula:
Ci = (Pi + Ki) mod 256
OTP encryption formula:
Ci = (Pi + Ki) mod 256

Plain: MTIUNPAB
Vigenere Key: ANZAS
OTP Key: 62 77 84 73 85 78 80 65

M = 77 + A = 65 + 62 = 204 mod 256
T = 84 + N = 78 + 77 = 239 mod 256
I = 73 + Z = 90 + 84 = 247 mod 256
U = 85 + A = 65 + 73 = 223 mod 256
N = 78 + S = 83 + 85 = 246 mod 256
P = 80 + A = 65 + 78 = 223 mod 256
A = 65 + N = 78 + 80 = 223 mod 256
B = 66 + Z = 90 + 65 = 221 mod 256

Cipher Text Result: 204 239 247 223 246 223 223 221 = **Ìï÷ßößßÝ**

Jurnal Sains dan Teknologi
Volume 6 No. 1 | Mei 2024 |pp: 52-57
E-ISSN : 2714-8661 | DOI : https://doi.org/10.55338/saintek.v6i1.3190

d) Decryption Process:
OTP decryption formula:
Pi = (Ci - Ki) mod 256
Vigenere decryption formula:
Pi = (Ci - Ki) mod 256

Cipher Text : Ìï÷ßößßÝ
Vigenere Key : ANZAS
OTP Key : 62 77 84 73 85 78 80 65

Ì = 77 - 62 - A = 65 = 77 mod 256
ï = 84 - 77 - N = 78 = 84 mod 256
÷ = 73 - 84 - Z = 90 = 73 mod 256
ß = 85 - 73 - A = 65 = 85 mod 256
ö = 78 - 85 - S = 83 = 78 mod 256
ß = 80 - 78 - A = 65 = 80 mod 256
ß = 65 - 80 - N = 78 = 65 mod 256
Ý = 66 - 65 - Z = 90 = 66 mod 256

Plaintext Result: 77 84 73 85 78 80 65 66 = **MTIUNPAB**

III. RESULT AND DISCUSSION

A. Encryption Process

The result of application development using the Visual Basic programming language and cryptographic methods utilizing the Vigenere, OTP, and LFSR algorithms. An example of encryption application is as follows:
Plaintext = **MAGISTERTEKNOLOGIIINFORMASI**
Vigenere Key = **UNPAB**
OTP Key = **199 77 65 71 73 83 84 69 82 84 69 75 78 79 76 79 71 73 73 78 70 79 82 77 65 83** (generated LFSR random key for 8 periods)
The resulting ciphertext =
iÜØÑÞüçççÛåçíÜÝëÞâØÖêïïÏÖñ

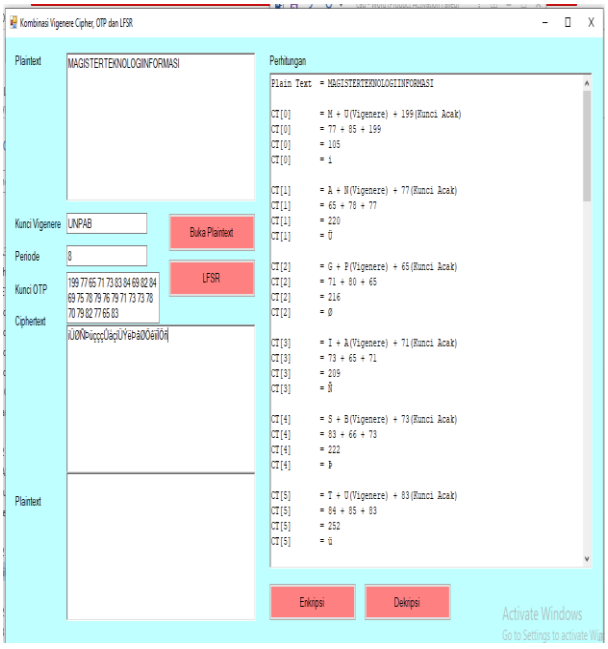

Figure 5. Encryption Process

B. Description Process

To perform the decryption of the combined algorithms, where the ciphertext is decrypted using the one-time pad algorithm with the key generated by LFSR, followed by the Vigenere cipher algorithm with the initial key, resulting in the original plaintext.
Here's an example application:
Ciphertext = **iÜØÑÞüçççÛåçíÜÝëÞâØÖêïïÏÖñ**
OTP Key = **199 77 65 71 73 83 84 69 82 84 69 75 78 79 76 79 71 73 73 78 70 79 82 77 65 83**
Vigenere Key = **UNPAB**
Plaintext = **MAGISTERTEKNOLOGIIINFORMASI**

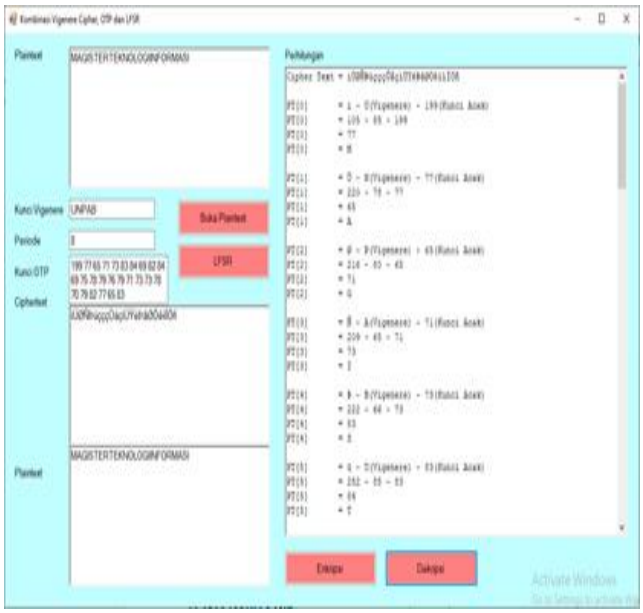

Figure 6. Descryption Proses

C. Testing

Plain	Key Vigenere	LFSR (OTP) 8 periode	LFSR (OTP) 14 periode	Ciphertext 8 periode	Ciphertext 14 periode
MTI	ANZA	62 77	252	Ìï÷ßö	Š›
UNP	S	84 73	249	ßßÝ	Øç
AB		85 78	53 81		Æ
		80 65	37 85		æÈ
			57 65		Ý
MAG	UNPA	199 77	59 29	iÜØÑ	Ý¬Ì
ITER	B	65 71	53 5	ßîåö	³¿ñ
TEK		73 84	29 37	ÚÒîë	¹ÏÞ¸
NOL		69 82	81 21	ëÜØ	ÊÕÍ
OGII		84 69	73 81	åàçÕ	ºÛ´
NFO		75 78	21 45	×öíÞ	Ã¬
RMA		79 76	57 61	ÕÞ	ÊÀ
SI		79 71	49 61		ØÚ
		73 73	29 37		É
		78 70	37 57		

Jurnal Sains dan Teknologi
Volume 6 No. 1 | Mei 2024 |pp: 52-57
E-ISSN : 2714-8661 | DOI : https://doi.org/10.55338/saintek.v6i1.3190

79 82	25 61
77 65	73 53
83	5

IV. CONCLUSION

The combination of the Vigenère cipher and one-time pad algorithms with random key LFSR serves as a promising method to enhance data security. By employing the LFSR method to generate OTP keys, the vulnerability to attacks and data breaches is significantly reduced. Moreover, increasing the scrambling period enhances the randomness of the OTP key. Utilizing a modulus of 256 yields a wider range of character results, adding a layer of complexity to character interpretation. It is suggested to further bolster security by generating the Vigenère key using XOR or Matrix methods.

ACKNOWLEDGEMENTS

The author would like to express deep gratitude to the leadership of Pembangunan Pancabudi University and the Head of the Master of Information Technology Program for granting permission and providing intellectual support for this research. The author also wishes to thank all fellow MTI students who have supported this research. In closing, I extend my thanks.

V. REFERENCES

[1] F. N. Pabokory, I. F. Astuti, and A. H. Kridalaksana, "Implementasi Kriptografi Pengamanan Data Pada Pesan Teks, Isi File Dokumen, Dan File Dokumen Menggunakan Algoritma Advanced Encryption Standard," *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 10, no. 1, p. 20, 2016, doi: 10.30872/jim.v10i1.23.

[2] S. T. Bonita, R. Marwati, and S. M. Gozali, "Pembangkit Kunci Linear Feedback Shift Register Pada Algoritma Hill Cipher Yang Dimodifikasi Menggunakan Convert Between Base," *J. EurekaMatika*, vol. 5, no. 2, pp. 20–28, 2017.

[3] O. K. Sulaiman, "Generate Pseudo-Random Numbers Linear-Feedback Shift Register (LSFR) Pada Kunci Algoritma One Time Pad (OTP)," *Semin. Nas. Teknol. Komput. Sains*, pp. 171–175, 2020.

[4] V. Hulu and B. Nadeak, "Kombinasi Algoritma Vigenere Cipher dan One Time Pad untuk Mengamankan Data Teks," vol. 02, no. 01, pp. 49–57, 2020.

[5] A. Amrulloh and E. I. H. Ujianto, "Kriptografi Simetris Menggunakan Algoritma Vigenere Cipher," vol. 5, no. 2, pp. 71–77, 2019.

[6] J. K. Panford, P. K. Yeng, J. B. H. Acquah, and F. Twum, "An Efficient Symmetric Cipher Algorithm for Data Encryption," *Int. Res. J. Eng. Technol.*, vol. 3, no. 5, pp. 1713–1732, 2016, [Online]. Available: https://www.academia.edu/download/54582100/IRJET-V3I5345.pdf.

[7] M. D. Irawan, "Implementasi Kriptografi Vigenere Cipher Dengan Php," *J. Teknol. Inf.*, vol. 1, no. 1, p. 11, 2017, doi: 10.36294/jurti.v1i1.21.

[8] A. Priyam, "Extended Vigenère using double Transposition Cipher with One Time Pad Cipher," *Intl J Engg Sci Adv Res.*, vol. 1, no. 2, pp. 62–65, 2015.

[9] G. G. Putri, W. Styorini, and R. D. Rahayani, "ANALISIS KRIPTOGRAFI SIMETRIS AES DAN KRIPTOGRAFI ASIMETRIS RSA PADA ENKRIPSI CITRA DIGITAL," *J. Penelit. Dan Pengabdi. Masy.*, vol. 6, no. 2, pp. 197–207, 2018.

[10] A. I. Permana, "Kombinasi Algoritma Kriptografi One Time Pad Dengan Generate Random Keys Dan Vigenere Cipher Dengan Kunci EM2B," *Tesis*, pp. 1–63, 2019.

[11] M. K. Harahap, "ANALISIS PERBANDINGAN ALGORITMA KRIPTOGRAFI KLASIK VIGENERE CIPHER DAN ONE TIME PAD," *J. Nas. Inform. dan Teknol. Jar.*, vol. 1, no. 190, pp. 61–64, 2016.

[12] R. T. Rahayu, A. Riski, and A. Kamsyakawuni, "Penyandian Citra Menggunakan Algoritma 4D Playfair Cipher Dengan Pembangkitan Kunci Modifikasi Linear Feedback Shift Register," *Maj. Ilm. Mat. dan Stat.*, vol. 19, no. 1, p. 17, 2019, doi: 10.19184/mims.v19i1.17261.

[13] Fareedha and K. Seetharam, "LFSR based Generation of Multicycle Test," *Int. J. Prof. Eng. Stuies*, vol. 10, no. 2, pp. 29–33, 2018.

[14] F. Diani and Y. Widhiyasana, "Enkripsi SMS dengan Menggunakan One Time Pad (OTP) dan Kompresi Lempel-Ziv-Welch (LZW)," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 7, no. 3, pp. 3–8, 2018, doi: 10.22146/jnteti.v7i3.436.

[15] A. Fauzi and Y. Maulita, "PERANCANGAN APLIKASI KEAMANAN PESAN MENGGUNAKAN ALGORITMA ELGAMAL DENGAN MEMANFAATKAN ALGORITMA ONE TIME," vol. 1, no. 1, 2017.