IJIE (Indonesian Journal of Informatics Education)

Research Paper ISSN: 2549-0389

Text Encryption Analysis of Modified Symmetric Keys using Genetic Algorithm on Caesar Cipher and One Time Pad

***Julia Kurniasih**
Program Studi Informatika
Universitas Sarjanawiyata Tamansiswa
julia.kurniasih@ustjogja.ac.id

Dian Tiara Rezalti
Program studi Informatika
Universitas Sarjanawiyata Tamansiswa
tiara@ustjogja.ac.id

Abstract:

The problem of data security is essential because data must be maintained securely and in its integrity, starting with the sending procedure and ending with the intended recipient. One way to maintain data security is to use cryptographic techniques, which use data encryption characteristics to transmit messages in secret (information encoding). Information can be encrypted using the Caesar Cipher and One Time Pad, symmetric key cryptographic algorithms. The key must be changed as a critical component of the cryptographic procedure to enhance the data or text's security. This study aims to analyse text encryption on the Caesar Cipher and One Time Pad (OTP) algorithms using a modified symmetric key by implementing a Genetic Algorithm, an optimisation algorithm. The research phase begins by adjusting the key using two types of Genetic Algorithm crossover operators, one-point crossover and two-point crossover, applied to the Caesar Cipher and OTP algorithms. Then testing and analysis of the ciphertext strength are carried out by comparing the frequency of character repetition with the results of implementing the two modification keys to determine which encoding model provides a better data security strength.

Keywords: Cryptography, Caesar Cipher, One Time Pad, Genetic Algorithm.

DOI: http://dx.doi.org/10.20961/ijie.v6i2.71434

Introduction

The increase in data transactions due to the use of information and communication technology on the internet raises data security problems in the transaction process. The foundation for this is the necessity that data be kept safe and intact from the time of delivery until the intended recipient receives it. Based on this situation, a science developed, namely the field of Cryptography, known as the science of encoding and hiding information, intended so that messages sent can be kept confidential and secure (Ariyus, 2008).

Numerous encryption algorithms, such as Caesar Cipher and One Time Pad (OTP), were created. Caesar Cipher is one of the oldest algorithms and is a type of substitution cypher, which forms a cypher by shifting all the characters in the plaintext with the same shift value. The Caesar Cipher's flaw is that it is possible to decipher the original message by employing brute force and analysing the proportion of letters that commonly appear in sentences. (Rachmawati & Candra, 2015). Meanwhile, One Time Pad is a symmetric cryptographic algorithm that contains a randomly generated vital character sequence. The security of both algorithms depends on the protection of the key. Keys are a vulnerability for symmetrical algorithms since they are easily predictable. The network security factor when exchanging keys also determines message security (Rachmawati, Sharif, & Sianipar, 2018). Furthermore, it is necessary to reinforce text security with additional algorithms because utilising just one traditional cryptographic technique carries a significant risk. Combining two or more classical cryptographic algorithms supports the guarantee that the messages sent are safe and not easily discovered by other parties (Saputro, 2020). Therefore, to cover these weaknesses, in this study, a symmetric essential modification was carried out in the combined implementation of the two algorithms using a genetic algorithm.

Genetic Algorithm (GA) is an optimisation algorithm that works based on the mechanism of natural selection and natural genetics. The desired outcome is a population of individuals (chromosomes) able to adapt to a specific environment and behave naturally. Chromosomes are typically binary-coded in GA. In general, GA starts by creating a random population of individuals. Applying reproduction operators (crossover and mutation) on the prior population will make new people. (Jhingran, Thada, & Dhaka, 2015).

Combining several cryptographic algorithms and modifying them through optimisation algorithms is expected to overcome the shortcomings of each part of the algorithm so that the encoding of information can be more assertive. This study modified the symmetry key using a genetic algorithm in one of two crossover stage variations—one-point and two-point crossover. The strength of each text encryption (ciphertext) created by altering the symmetric key of the Caesar Cipher algorithm and One Time Pad will then be checked and examined.

This research is a continuation of previous research by Ariyus, Kurniasih, & Profesi (2019). That study focuses on modifications with one-point crossover only. Meanwhile, in this study, an amendment was also carried out with a two-point crossover to determine which amendment has better strength.

Related Works

Several studies related to the Caesar Cipher algorithm, One Time Pad. Although evolutionary algorithms have been employing studies on cryptography, crossover operator modifications still need to be investigated. Gunawan (2018) combines the Caesar Cipher and RSA algorithms to secure document files and text messages. This combination serves to get around the Caesar Cipher algorithm's flaws. The results of his research state that the combination of the Caesar Cipher and the RSA algorithm can improve the data security system by combining the calculation of the alphabetic structure and factoring of prime numbers. Harahap & Khairina (2017) analysed the application of the One Time Pad algorithm and the transposition cypher algorithm for text security. The results of their research stated that for long messages, the implementation of the transposition cypher algorithm was better than the One Time Pad because the transposition cypher algorithm could perform a complete description.

In comparison, the One Time Pad had broken decryption results. Jhingran et al. (2015) proposed a new approach/method for e-security by using genetic algorithms and pseudorandom series to generate encryption and decryption keys for data. The results show that the proposed algorithm provides a better throughput rate. Srikanth et al. (2017) implement genetic algorithm operations for cryptographic encryption and decryption processes. The outcomes demonstrate that data in a file may be secured using the genetic algorithm's mode of operation. Nazeer et al. (2018) proposed an algorithm called Genetic Crypto. Genetic Crypto development uses a Genetic Algorithm to increase key strength, improving the overall algorithm. Three steps make up genetic crypto: key generation, data diffusion, and data encryption. Their research shows that Genetic Crypto has better results regarding crucial strength but is still weak in computation (less efficiency). There is also research conducted by Kalsi et al.(2018), which introduces the concept of DNA Deep Learning Cryptography. They propose a method and implementation of crucial generation based on the theory of natural selection using Genetic Algorithms with the Needleman-Wunsch (NW) algorithm and ways of implementing encryption and decryption based on DNA computing using the biological operations Transcription, Translation, DNA Sequencing, and Deep Learning. Mittal & Gupta (2019) developed an algorithm for encrypting and decrypting a message based on a symmetric key cryptosystem involving Genetic Algorithms. This research uses substitution algorithms, genetic crossover and mutation techniques.

Research Method

This study uses the Caesar Cipher cryptographic algorithm, One Time Pad (OTP,) and Genetic Algorithm to develop symmetric essential modification. The hexadecimal conversion method utilising the ASCII table converts plaintext to ciphertext and vice versa. Figure 1 displays the flowchart.

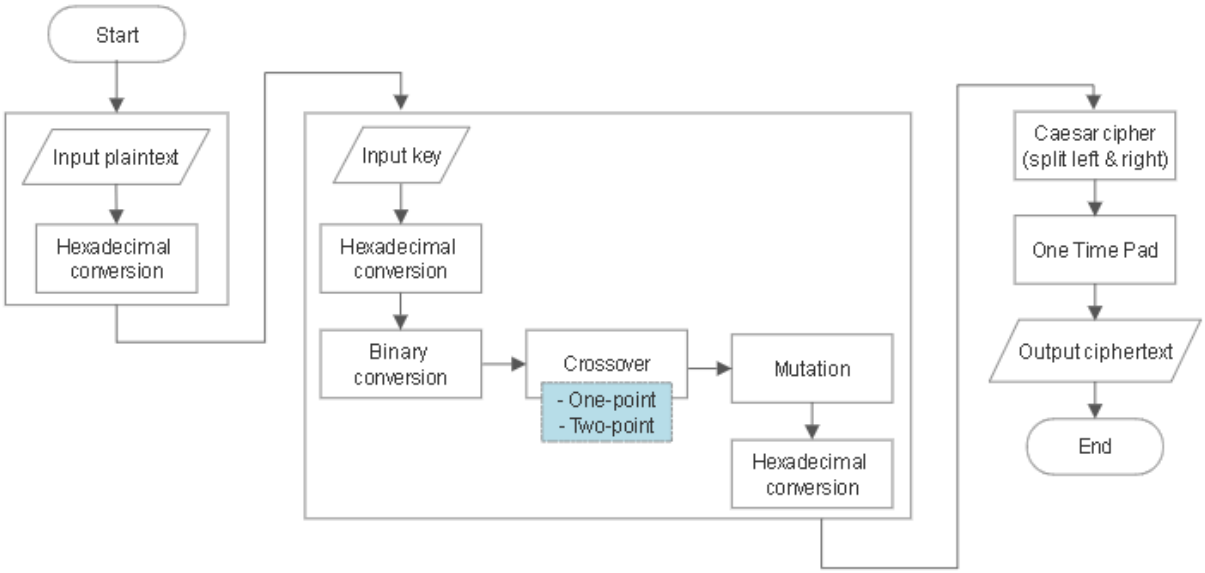

Figure 1. Flowchart of the Modification Process

The stages of the research carried out are:

1. Application of Genetic Algorithm (GA) on symmetric keys.

   Starting with making a random population of individuals and then creating new people by applying the crossover and mutation reproduction operators to the previous population. The crossover operators used in this study are one-point and two-point crossover, whereby the parent chromosome crossing occurs in pairs. The result of the crossover process will experience a mutation process. In this study, the standard mutation operator used is a binary string by inverting the mutated bit '0' to '1' and vice versa.
2. Application of modified symmetric key on Caesar Cipher algorithm for the text encryption.

   The key has two pieces: the left side and the right side of the Caesar Cipher key. The Caesar Cipher plaintext will then be encrypted using One Time Pad (OTP).
3. Application of OTP to encrypt Caesar Cipher plaintext and generate the ciphertext.

   An altered symmetric key and a hexadecimal random sequence generate a random series of key characters (pads) to do this. Messages are encrypted only once (once) using a single residence. Messages are encrypted only once (once) using a single pad. It again constitutes encrypting and decrypting further communications, repeating the randomisation process.
4. Stages 1 to 3 are carried out for one-point and two-point crossover processes, respectively. As a result, the ciphertext for each updated symmetric key used in the two crossovers
5. operations will be obtained.
6. The two updated symmetric keys' ciphertexts will undergo a comparison for character repetition rates to determine which encoding scheme offers the highest level of data security.

Result and Analysis

Result

The schematic of the text encryption process refers to in Figure 1. Table 1 shows an example of text encryption utilising plaintext and its translation to hexadecimal form. Table 2 shows the key that was used.

Table 1. The plaintext and its hexadecimal conversion

S	A	Y	A		M	A	U		B	E	K	E	R	J	A
53	41	59	41	20	4D	41	55	20	42	45	4B	45	52	4A	41

Table 2. The encryption keys

K	R	I	P	T	O	G	R	A	F	I
4B	52	49	50	54	4F	47	52	41	46	49

The keys in Table 2 were modified using a genetic algorithm through crossover and mutation operations. Table 3 displays the outcomes of changing the key from hexadecimal to binary format.

Table 3. Convert key to binary form

0100	0101	0100	0101	0101	0100	0100	0101	0100	0100	0100	0000
1011	0010	1001	0000	0100	1111	0111	0010	0001	0110	1001	0000

1. Modification with One-Point Crossover

The process carried out at the modification stage is as follows:

a. Determine the crossover point.

   The two halves of the two chosen chromosomes swap places to create a new chromosome after the selected chromosome

b. breaks in half.

c. The one-point crossover method performs a crossover operation on the binary key (Table 3), and the outcomes appear in Table 4. As stated in Table 5, the outcomes of the crossover operation will go through mutation and turn back into hexadecimal form.

Table 4. One-point crossover operation

0101	0100	0101	0100	0100	0101	0101	0100	0100	0100	0000	0100
1011	0010	1001	0000	0100	1111	0111	0010	0001	0110	1001	0000

Table 5. Mutation operation

1010	1011	1010	1011	1011	1010	1010	1011	1011	1011	1111	1011
0100	1101	0110	1111	1011	0000	1000	1101	1110	1001	0110	1111
A4	BD	A6	BF	BB	A0	A8	BD	BE	B9	F6	BF

The modified key using the one-point crossover is called the o-genetic key, and then it will be used for the text encryption process using the Caesar Cipher algorithm and One Time Pad. The left and right halves of the o-genetic

fundamental divide for the Caesar Cipher procedure. The left and right side of the Caesar Cipher key sequence is shown in Figure 2 and Figure 3, respectively.

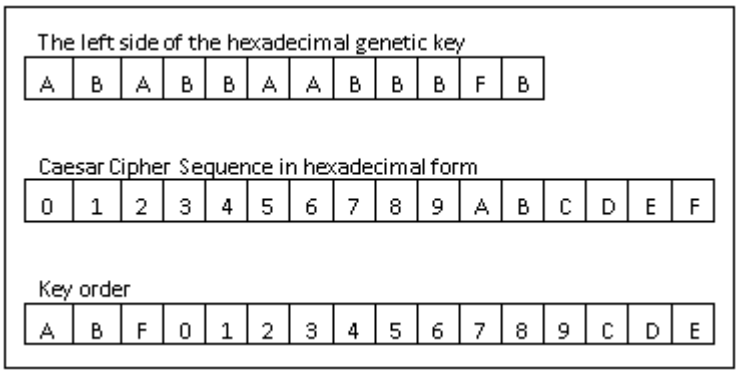

Figure 2. Left side Caesar Cipher key

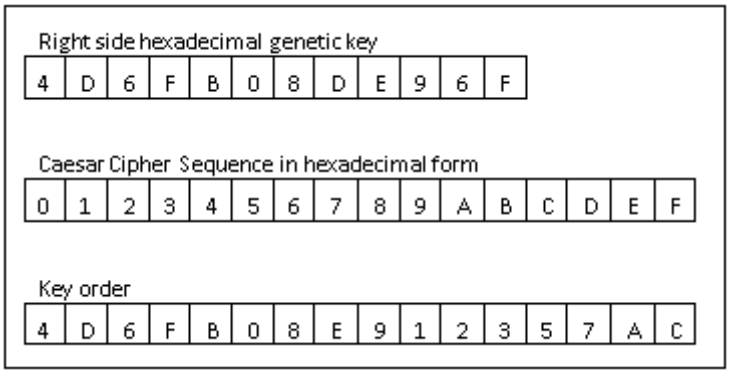

Figure 3. Right side Caesar Cipher key

The plaintext encryption process in Table 1 with the left and right Caesar Cipher keys in Figure 2 and Figure 3 produces the encryption text as shown in Table 6.

Table 6. Caesar Cipher plaintext

2F	1D	21	1D	F4	17	1D	20	F4	16	10	13	10	26	12	1D

The plaintext from Caesar Cipher will then undergo an encryption process using the One Time Pad algorithm. Using the o-genetic key in Table 5 and a random hexadecimal sequence, as shown in Table 7, the plaintext resulting from the Caesar Cipher will be encrypted to produce a ciphertext, as shown in Table 8.

Table 7. Hexadecimal random sequence scheme for One Time Pad algorithm

Normal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Random	A	4	B	D	6	F	0	8	E	9	1	2	3	5	7	C
Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Table 8. The ciphertext of the One Time Pad algorithm (in hexadecimal)

20	30	27	3E	8D	16	11	59	89	35	C1	34	18	58	1C	3E

The repeat frequency of characters in the generated ciphertext applies to perform the o-genetic symmetric critical test. The test results appear in Table 9 with the crucial text rendered as "penelitian," and the underscore symbol separates terms in the plaintext. (_).

Table 9. Test results of symmetric essential modification using one-point crossover

Plaintext	Ciphertext (hexadecimal)	Ciphertext (character)	Frequency of Repetition of Characters in Ciphertext
tuliskan_secara_ringkas_hasilnya	5B C2 46 4C 52 4A 42 71 3B 50 4E 7E 4B 5B 4E 33 54 7F 46 47 4C 76 50 38 4D 40 55 7F 42 48 55 76	[?FLRJBq;PN ~K[N3T□FG LvP8M@U□ BHUv	- The characters that appear the most are two times, namely '**[**', '**F**', '**P**', '**N**', '□', '**L**', '**v**', '**U**', and '**B**'. - The total number of characters is 32 characters. - Percentage of occurrence of 9 characters above = (2/32)x9 = 0.56%
PENYAJIAN_DATA_DAPAT_BERUPA_GAMBAR_DAN_TABEL	38 52 28 3C 2E 28 28 50 26 3A 2B 56 32 26 31 25 22 BB 24 32 39 5B 24 3B 35 3B 22 B3 2F 2B 2F 5B 2B 3B 31 25 22 51 3B 32 21 5B 24 23	8R(<.((P&:+V 2&1%"?$29[$;5;"?/+/[+;1%" Q;2![$#	- The character that appears the most is '**;**', four times. - The total number of characters is 44 characters. - Percentage of occurrences of the character '**;** ' = 4/44 = 0.09%.
Menentukan_jumlah_DATA_dan_sumber_DATA_yang_digunakan	2F 72 48 42 4A 55 5C 7A 44 48 39 7D 54 4A 4B 40 4F B3 2C 2B 3B 56 3A 40 4E 41 36 C4 5D 41 43 72 5E 38 24 20 37 50 3B 5D 41 7F 47 38 44 4F 49 C7 46 4B 4C 76 48	/rHBJU\zDH9 }TJK@O?,+; V:@NA6?]AC r^8$ 7P;]A□G8DO I?FKLvH	- The characters that appear the most are three times: '**H** ', '**A**', and '**?**'. - The total number of characters is 53 characters. - Percentage of occurrence of 3 characters above = (3/53)x3 = 0.17%.
rumus_untuk_menghitung_luas_segitiga=1/2(alas*tinggi)	53 C2 41 52 52 33 5C 71 5C 54 4C B8 41 42 4A 4D 4F 7F 5C 54 4D 75 3A 43 55 40 55 B3 57 44 42 7C 52 4C 40 1B 60 0B 1E 04 76 46 46 52 08 57 7F 46 47 42 7C 0D	S?ARR3\q\TL? ABJMO'□\TMu :CU@U?WDB| RLL@'□` -'□vFFR'□W'□ FGB|	- The character that appears the most is '□', which is five times. - The total number of characters is 53 characters. - Percentage of occurrence of the character '□' = 5/53 = 0.09%.
Jenis_Luaran_Yang_Diberikan_Pada_Laporan_Kemajuan_Ini_Masih_Bersifat_Sementara_Yaitu_Berupa_Hasil_Pengujian_Dar_	27 72 48 4C 52 33 20 C7 44 5E 41 7F 3A 3C 4E 41 49 B3 2C 4D 43 72 5E 4C 48 40 43 B3 30 4B 76 3A 23 4E 5B 46 C2 44 48 39 59 44 4A 4E 48 5C 70 46 3A 25 7F 4D 38 23 40 55 7F 48 3A 23 72 5E 49 4C 42 C5 3B 30 4E 7A 44 4F 54 40 54 70 3B 3D 41 7C 52 52 31 22 4C C2 5D 53 41 B8 2C 46 52 4F 40 B3 30 44 4D	'rHLR3 ?D^A□:<NAI ?,MCr^LH@C ?0KKv:#N[F? DH9YDJNH\p F:% □M8#@U□H: #r^^ILB?;0Nz DOT@Tp;=A|	- The character that appears the most is '**?**', 19 times. - The total number of characters is 234 characters. - Percentage of occurrence of the character '**?**' = 19/234 = 0.08%.

Sebagian_Proses_Penelitian_Sebagaimana_Terdapat_Pada_Poin_D,_Karena_Pelaksanaan_Penelitian_Belum_Selesai_Secara_Keseluruhan	75 54 4D 49 40 43 B3 2C 4B 53 B8 30 42 40 40 49 7F 44 48 39 B1 5E 48 52 47 55 B3 30 44 4D 72 46 4C 54 4F 42 71 3B 30 4E 7B 4B 45 4E 4F 4B 70 46 4B 39 B0 44 5B 44 40 5E 70 5C 3A 38 76 42 46 31 3B 46 7F 46 3A 2B 43 3A 29 4E 52 4C 71 44 3A 38 72 46 46 48 54 42 71 44 4B 4D B8 33 42 4A 47 40 7F 5C 4D 41 7F 3A 2B 45 4E 5C 76 3B 30 4E 73 44 5E 4E 4F 36 B4 4D 40 41 CB 4B 38 28 47 55 77 42 54 53 C2 4C 46 4A	RR1"L?]SA?, FRO@?0DMu TMI@C?,KS? 0B@@I□DH9 ?^HRGU?0D MrFLTOBq;0 N{KENOKpF K9?D[D@^p\: 8vBF1;F□F:+ C:)NRLqD:8r FFHTBqDKM ?3BJG@□\M A□:+EN\v;0N sD^NO6?M@ A?K8(GUwB TS?LFJ

2. Modification with Two-Point Crossover

As in the one-point crossover, the process of the two-point crossover carried out at the modification stage is as follows:

a. Determine the crossover point.

Two points work to split a chromosome into three parts, and then one component of each section of each chromosome is swapped to create new chromosomes.

d. Undergoes a crossover operation with the two-point crossover method; the results appear in Table 10. As illustrated in Table 11, the outcomes of the crossover operation will go through mutation and transform back into hexadecimal form.

Table 10. Two-point crossover operation

0100	0101	0100	0101	0101	0100	0100	0101	0100	0100	0000	0100
1010	0011	1000	0001	0111	1100	0010	0111	0110	0001	1000	0001

Table 11. Mutation operation

1011	1010	1011	1010	1010	1011	1011	1010	1011	1011	1111	1011
0101	1100	0111	1110	1000	0011	1101	1000	1001	1110	0111	1110
B5	AC	B7	AE	A8	B3	BD	A8	B9	BE	F7	BE

The modified key using the two-point crossover is called the t-genetic key, and then it will be used for the text encryption process using the Caesar Cipher algorithm and One Time Pad. The left and right side keys are the t-genetic keys for the Caesar Cipher procedure. The left and right side of the Caesar Cipher key sequence is shown in Figure 4 and Figure 5, respectively.

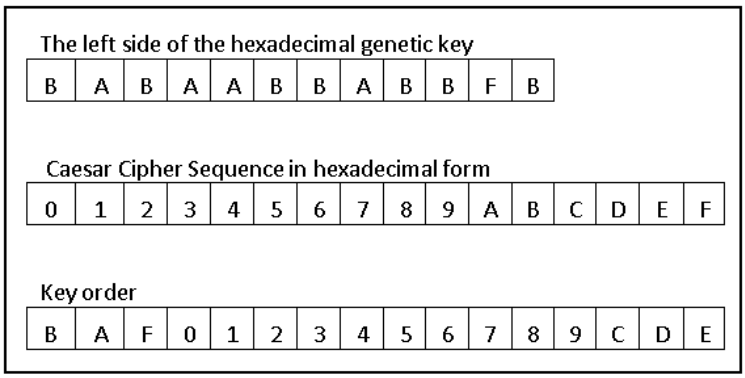

Figure 4. Left side Caesar Cipher key

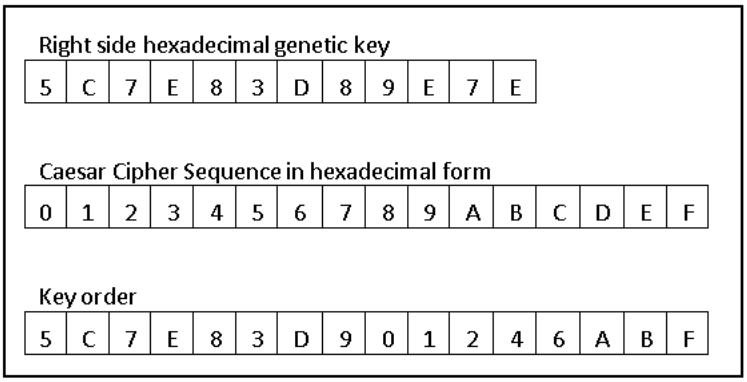

Figure 5. Right side Caesar Cipher key

The plaintext encryption process in Table 1 with the left and right Caesar Cipher keys in Figure 4 and Figure 5 produces the encryption text as shown in Table 12. The One Time Pad algorithm will then encrypt the plaintext Caesar Cypher result. Using the t-genetic key in Table 11 and a random hexadecimal sequence, as shown in Table 13, the plaintext resulting from the Caesar Cipher will be encrypted to produce a ciphertext, as shown in Table 14.

Table 12. Caesar Cipher plaintext

2E	1C	21	1C	F5	1A	1C	23	F5	17	13	14	13	27	12	1C

Table 13. Hexadecimal random sequence scheme for One Time Pad algorithm

Normal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Random	B	5	A	C	7	E	8	3	D	9	F	0	1	2	4	6
Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Table 14. The ciphertext of the One Time Pad algorithm (in hexadecimal)

28	48	2B	4D	13	19	10	62	F	19	80	1C	1D	63	15	4D

It is also possible to do the symmetric critical modification test with the t-genetics method by examining how frequently certain characters occur in the ciphertext that results from that. The test results can be seen in Table 15 with the critical text used as 'penelitian'. In this research, the separator between words in plaintext uses the underscore symbol (_).

Table 15. Test results of symmetric essential modification using two-point crossover

Plaintext	Ciphertext (hexadecimal)	Ciphertext (character)	Frequency of Repetition of Characters in Ciphertext
tuliskan_secara_ringkas_hasilnya	5A C1 46 4C 51 49 40 62 3E 54 4D 6D 4E 52 40 3E 57 69 48 4F 46 6A 52 3E 4C 4D 51 69 4B 47 54 6A	Z?FLQI@b>TMmNR@>WiHOFjR>LMQiKGTj	- The characters that appear the most are three times, namely '>'. - The total number of characters is 32 characters. - Percentage of occurrence of the character '>' = 3/32 = 0.09%
PENYAJIAN_DATA_DAPAT_BERUPA_GAMBAR_DAN_TABEL	38 A1 2C 3C 20 2F A1 28 3A 2A AA 37 2D 32 21 20 5D 2D 35 39 AE 2A 32 34 37 20 57 25 23 2E AE 2E 32 32 21 20 A2 3E 35 22 AE 2A 2B	8?,< //?(:*?7-2!]-59?*247W%#.?.22!?>5"	- The character that appears the most is '**?**', six times. - The total number of characters is 44 characters. - Percentage of occurrences of the character '? ' = 6/44 = 0.14%.
Menentukan_jumlah_DATA_dan_sumber_DATA_yang_digunakan	2E 61 4C 43 4E 51 54 68 4D 47 39 6C 5A 4A 48 4D 4C 57 21 23 3A AA 3F 41 40 48 32 C3 53 41 4B 61 59 3E 23 2D 33 A1 3E 5B 42 6B 40 3E 43 4C 46 C5 48 43 46 6A 4C	.aLCNQThMG9lZJHMLW!#:??A@H2?SAKaY>#-3?>[Bk@>CLF?HCFjL	- The characters that appear the most are five times, namely '**?**'. - The total number of characters is 53 characters. - Percentage of occurrence of the character '? ' = 5/53 = 0.09%.
rumus_untuk_menghitung_luas_segitiga=1/2(alas*tinggi)	5B C1 48 53 51 3E 54 62 51 56 46 58 48 43 4E 45 4C 69 51 56 4F 64 3F 4B 54 4D 51 57 50 46 41 66 57 4C 46 4D 1D 41 0E 1D 03 6A 46 4D 51 0F 53 69 48 4F 41 66 03	[?HSQ>TbQVFXHCNELiQVOd?KTMQWPFAfWLFM□A□□jFMQ□SiHOAf□	- The characters that appear the most are '**Q**' and '□' @5 times. - The total number of characters is 53 characters. - Percentage of characters 'Q' and '□' occurrence = (5/53)x2 = 0.19%.
Jenis_Luaran_Yang_Diberikan_Pada_Laporan_Kemajuan_Ini_Masih_Bersifat_Sementara_Yaitu_Berupa_Hasil_Pengujian_Dar_Sebagian_Proses_Penelitian_Sebagaimana_Terdapat_Pada_Poin_D,_Karena_Pelaksana	25 61 4C 51 3E 28 C5 4D 5D 42 6B 3F 3C 40 48 46 57 21 4B 61 59 4C 4B 4D 4E 57 37 43 4A 6A 3F 2B 40 57 42 CA 4D 47 39 AF 4A 4A 40 4F 54 61 48 3A 24 6B 43 3E 2D 4D 51 69 46 3A 2B 61 59 50 4F 44 40 C4 3E 34 4D 67 4A 48 53 4D 57 61 3E 3B 42 66 57 53 32 22 44 CA 53 50 42 58 21 4D 51 4C 48 57 37 46 4F 64 5A 4F 4F 4D 4E 57 21 43 5B 58 32 43 47 4D 46 69 4D 47 39 52 59 4E 51 43 51 57 37 46 4F 61 46 4C 53 4C 40 62 3E 34 4D 6E 4E 45 40 4C 4D 61 48 43 39 50 4A 52 43 4D 5A 61 51 3A 38 6A 47 4D 32 37 42	%aLLQ>(?M]Bk?<@HFW!KKaYLKMNW7CJj?+@WB?MG9?JJ@OTaH:$kC>-MQiF:+aYPOD@?>4MgJHSMWa>;BfWS2"D?SPBX!MQLHW7FOdZOOMNW!C[X2CGMFiMG9RYNQCQW7FOaFLS	- The character that appears the most is '**M**', 19 times. - The total number of characters is 234 characters. - Percentage of occurrence of the character ' **M** ' = 19/234 = 0.08%.

an_Penelitian_ Belum_Selesai _Secara_Kesel uruhan	69 48 3A 2A 39 3F 29 40 52 44 62 4D 3A 38 61 46 4D 4B 50 40 62 4D 43 4F 58 3B 43 4E 43 48 69 51 4B 42 6B 3F 22 44 4B 54 60 3E 34 4D 69 4A 50 40 4C 32 53 43 44 42 CE 4E 3E 2B 43 51 65 4B 56 5B C1 41 4D 4E	L@b>4MnNE @LMaHC9PJ RCMZaQ:8jG M27BiH:*9?) @RDbM:8aF MKP@bMCO X;CNCHiQK Bk?"DKT`>4 MiJP@L2SC DB?N>+CQe KV[?AMN

Analysis

The test results show that the average repetition of characters in the ciphertext of the Caesar Cipher and OTP cryptographic algorithms with symmetric essential modification using the o-genetic algorithm is 0.20%, and using the t-genetics algorithm is 0.12%. In the t-genetics implementation, character repetition is less frequent than in the o-genetics approach.

Conclusion

Based on the average percentage of occurrence of the characters, it can be concluded that the strength of text encoding using a modified symmetric key resulting from the t-genetics algorithm (two-point crossover) is better than the o-genetic algorithm (one-point crossover).
Recommendations for further research are symmetric critical optimisation for stages in the selection process in Genetic Algorithms.

Acknowledgement

We want to thank LP2M Universitas Sarjanawiyata Tamansiswa for the support given so that this research can be completed properly.

References

Ariyus, D. (2008). *Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi.* Yogyakarta: Andi Offset.

Ariyus, D., Kurniasih, J., & Profesi, D. E. (2019). Modifikasi Kunci Simetris Caesar Cipher dan OTP Menggunakan Algoritma Genetika Pada Steganografi. *CSRID (Computer Science Research and Its Development Journal)*.

Gunawan, I. (2018). Kombinasi Algoritma Caesar Cipher dan Algoritma RSA untuk Keamanan File Dokumen dan Pesan Teks. *Jurnal Nasional Informatika dan Teknologi Jaringan (InfoTekJar)*.

Harahap, M. K., & Khairina, N. (2017). Analisis Algoritma One Time Pad Dengan Algoritma Cipher Transposisi Sebagai Pengamanan Pesan Teks. *Sinkron : Jurnal Dan Penelitian Teknik Informatika*, 58-62.

Jhingran, R., Thada, V., & Dhaka, S. (2015). A Study on Cryptography using Genetic Algorithm. *International Journal of Computer Applications (IJCA)*.

Kalsi, S., Kaur, H., & Chang , V. (2018). DNA Cryptography and Deep Learning using Genetic Algorithm with NW algorithm for Key Generation. *Journal of Medical Systems*.

Mittal, A., & Gupta, R. K. (2019). Encryption and Decryption of a Message Involving Genetic Algorithm. *International Journal of Engineering and Advanced Technology (IJEAT)*.

Nazeer, M. I., Mallah, G. A., Shaikh, N. A., Bhatra, R., Memon, R. A., & Mangrio, M. I. (2018). The Implication of Genetic Algorithm in Cryptography to Enhance Security. *(IJACSA) International Journal of Advanced Computer Science and Applications*.

Rachmawati, D., & Candra, A. (2015). Implementasi Kombinasi Caesar Cipher dan Affine Cipher Untuk Keamanan Data Teks. *Jurnal Edukasi dan Penelitian Informatika (JEPIN), 1*(2).

Rachmawati, D., Sharif, A., & Sianipar, R. (2018). A combination of the vigenere algorithm and one-time pad algorithm in the three-pass protocol. *MATEC Web of Conferences.* EDP Sciences.

Saputro, F. A. (2020). *Implementasi Algoritma One Time Pad Cipher dan.* Malang: Jurusan Matematika Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim.

Srikanth, P., Mehta, A., Yadav, N., Singh, S., & Singhal, S. (2017). Encryption and Decryption Using Genetic Algorithm Operations and Pseudorandom Number. *IJCSN - International Journal of Computer Science and Network.*