A combination of hill cipher and RC4 methods for text security

Azanuddin[1], Rikie Kartadie[2], Fauzi Erwis[3], Ahmad Fitri Boy[4], Asyahri Hadi Nasyuha[5]
[1]Department of Computer Engineering, Faculty of Computer Engineering and Informatics, Politeknik Negeri Medan, Medan, Indonesia
[2]Department of Computer Engineering, Faculty of Information Technology, Universitas Teknologi Digital Indonesia, Yogyakarta, Indonesia
[3]Department of Information Technology Education, Faculty of Education, Universitas Rokania, Riau, Indonesia
[4]Department of Information System, Faculty of Information System, STMIK Triguna Dharma, Medan, Indonesia
[5]Departmen of Information of System, Faculty of Information Technology, Universitas Teknologi Digital Indonesia, Yogyakarta, Indonesia

Article Info

Article history:

Received Aug 21, 2023
Revised Dec 22, 2023
Accepted Jan 12, 2024

Keywords:

Combination
Data security
Hill chiper
Rivest Cipher
Rivest Cipher 4

ABSTRACT

To hide confidential messages from people who are not responsible or who can access the messages, a way is needed to hide the messages. One way to hide messages in transmission is to change the data into something unintelligible by encoding and embedding it using cryptography and steganography techniques. This application was built using the hill cipher algorithm and the Rivest Cipher 4 (RC4) method. This algorithm is a symmetric key algorithm which has several advantages in data encryption. The hill chiper algorithm uses a mxm matrix as the encryption and decryption key. Meanwhile, the RC4 symmetric key is in the form of a stream cipher which can process input data as well as messages or information. Input data is generally in the form of bytes or even bits. The results of this research show that hill cipher and RC4 have their respective advantages and disadvantages. However, currently, RC4 is generally considered less safe for use in security-critical scenarios due to its vulnerability to attack. It is highly recommended to use an encryption algorithm such as advanced encryption standard (AES) which is modern and strong and has been tested and proven to be more resilient.

Corresponding Author:

Azanuddin
Department of Computer Engineering, Faculty of Computer Engineering and Informatics
Politeknik Negeri Medan
Jl. Alamamater No. 1 Padang Bulan, Medan, Indonesia
Email: azanuddin@polmed.ac.id

1. INTRODUCTION

Security issues are an important aspect in sending data and information over the network [1]. This is due to computer networks with an open system concept that can make it easier for someone to enter the network, which can result in the process of sending data being insecure and can be used by people or other parties who are not responsible for taking information in the middle of the road. One way to maintain the security of confidential data or information is to use cryptography [2]. Cryptography is an encryption technique in which "original text" (plaintext) can be scrambled using an encryption key to become "random text that is hard to read" (ciphertext) [3]–[5]. Cryptography has many methods, namely hill cipher and Rivest Cipher 4 (RC4) [6]–[8]. Hill cipher is a symmetric key cryptographic algorithm that has several advantages in data encryption [9]. To avoid invertible key matrices, key matrices are generated using newton's binomial coefficients [10], [11]. The encryption and description process uses the same key, plaintext can use image or text media. RC4 is a stream cipher type so that it can process units or input data, messages or information. The unit or data is generally a byte or sometimes even a bit (a byte in the case of RC4) so that in this way the encryption or decryption can be carried out at variable lengths.

Cryptography is the science of encryption techniques where data is scrambled using an encryption key to be something that is difficult to read someone who doesn't have the decryption key. Decryption using the decryption key gets back the original data. The encryption process is carried out using an algorithm with several parameters. Encryption that relies on the secrecy of the algorithm is considered something that is not good. The secret lies in the several parameters used. It is the parameter that determines the decryption key that must be kept secret. Cryptography is a security method for protecting information by using passwords that can only be understood by people who have the right to access the information. Cryptography is the only method used to protect information through communication networks that use landlines, communication satellites, and microwave facilities. Cryptography is a science that studies mathematical techniques related to aspects of information security such as confidentiality, data integrity, and authentication of data senders/recipients. Cryptography is an art or science to maintain the confidentiality of a text so that it remains safe without being noticed by unauthorized parties. But now cryptography is not just art or secrecy but also integrity, authentication, and data validity.

2. CRYPTOGRAPHIC WORK SYSTEM

In general, cryptography is the practice and study of techniques for securing communication and data from third parties. Cryptographic systems are employed in various areas such as securing communication over the internet, protecting sensitive information, ensuring the integrity of data, and more. The keys that can be used for encryption and decryption need not be identical, depending on the system used [12], [13]. Mathematically the process of encryption and decryption can be written:

EK (M): C (encryption process)
DK (C): M (decryption process)
Where E: (encryption process)
K: key
M: original message
C: encrypted text
D: decryption process

During the encryption process, the message (M) will be coupled using the encryption key (K) to a password that cannot be understood (C) [14], [15]. While in the decryption process, the password that is not understood (C) will be deciphered using the decryption keyword (K) so that it can produce the same message (M) as the previous message. The fundamental functions in cryptography are encryption and decryption [16], [17].

a. Encryption

Encryption is the process of changing an original message (plaintext) into a message in coded language (ciphertext) [18], [19]. Encryption is part of cryptography and is very important so that the security of the data sent can be kept confidential [20]. Encryption can be interpreted as a cipher or code, where the original message (plaintext) is converted into separate codes according to the method agreed upon by the message and the recipient of the message. $C = E(M)$ where C=message in coded language (ciphertext), E=encryption process, and M=original message (plaintext). Figure 1 describes both the encryption process and the decryption process.

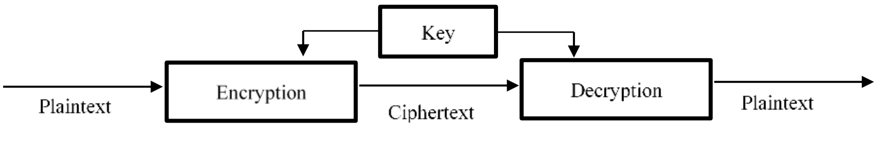

Figure 1. Encryption process and description

b. Decryption

Decryption is the process of changing messages in a coded language (ciphertext) into original messages (plaintext) [21], [22]. Decryption is the reverse process of encryption which returns the passwords or information that has been traced to the original file form by using a key or code. $M = D\ (C)$ where M=original message (plaintext), D=decryption process, and C message in coded language (ciphertext). In addition to using certain functions for encryption and decryption, these functions are often given additional parameters called keys.

Cryptographic algorithms can also be called ciphers, namely the rules for encrypting and decoding, or the functions used for encryption and decryption. The security of cryptographic algorithms is often measured by the amount of work required to break ciphertext into plaintext without knowing the key used. If the more processes needed mean the longer it takes, the stronger the algorithm is and the more secure it is used to encode messages. In cryptography there are various cryptographic algorithms based on the key, namely symmetric algorithms, asymmetric algorithms, hill cipher algorithms, and RC4 algorithms.

c. Symmetric algorithm

This algorithm is also often called the classical algorithm because it uses the same key for encryption and decryption [23]. To send messages using this algorithm, the recipient must be informed of the key to the message so that it can decrypt the message to be sent. In Figure 2, it explains the security of messages using this algorithm depending on the key, if the key is known by someone else then that person can encrypt and decrypt the message.

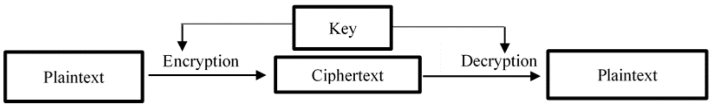

Figure 2. Symmetrical algorithm work process

d. Asymmetric algorithm

Asymmetric cryptography algorithms are algorithms that use different keys for the encryption and decryption processes [24]. Figure 3 explains the asymmetric cryptography algorithm, also known as the public key algorithm, because the key for encryption is public (public key) or can be known by everyone, but the key for decryption can only be known by authorized people who know it with the encoded data. Or often called private key.

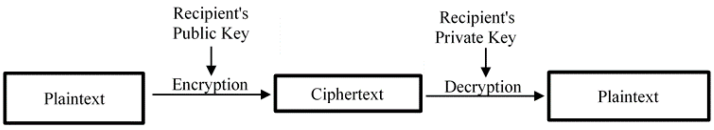

Figure 3. Asymmetric algorithm scheme

e. Hill cipher algorithm

Hill cipher is an application of modulo arithmetic to cryptograph [25]. This technique uses a square matrix as a key that can be used to perform encryption and decryption. Hill cipher cryptography does not replace every other letter that is the same in the ciphertext because it uses matrix multiplication based on encryption and description. Hill cipher is a polyalphabetic cipher which can be categorized as a block cipher because the selected text will be processed and divided into blocks of a certain size. Each character has a block that can influence other characters in the encryption and decryption process, so that the same character is not differentiated into the same character. The hill cipher was created by Lester S. Hill in 1929. The hill cipher does not replace every letter that is the same in plaintext with another alphabet in the ciphertext, because hill cipher uses matrix multiplication in its encryption and description. Hill cipher is a vulnerable symmetric encryption algorithm against known-plaintext attacks. If cryptanalysis can collect plaintext and ciphertext with the same key, then cryptanalysis can find out the key from hill.

f. RC4 stream algorithm

The RC4 algorithm is a symmetric key in the form of a stream cipher that can process input data or messages or information [26]. Input data is generally a byte or even bits. In this algorithm you don't have to wait for a certain amount of input data or information to add bytes to encrypt. The RC4 algorithm has two S-boxes, namely, an array of length 256 which contains permutations from 0 to 255, and the second S-box is a function of keys with variable lengths. The way the RC4 algorithm works is to initialize the first S-box array, S[0], S[1] ,...., S[255], with numbers 0 to 255. Fill in the first thing sequentially S[0]=0, S[1]=1 ,..., S[255]=255. Whereas the second S-box is for example an array K with a length of 256. An array of K with keys that can be repeated until the entire array is K[0], K[1] ,..., K[255] is completely filled. In general, in the encryption-decryption process, there are two types of ciphers based on how the encoding works, namely a stream cipher is a system where the encryption and decryption process are done bit by bit. In this system the key bit stream is generated by a random bit generator. This key stream is subjected to an XOR operation with a stream of plaintext bits to produce a stream of ciphertext bits.

3. RESULTS AND DISCUSSION

In a system required an analysis of the system to be designed. This study uses observation of data security by using a database and using the hill cipher and RC4 cryptographic systems. Hill cipher is a software that is used to secure data. The way the hill cipher works is quite simple and easy to understand so that it determines the security level of a cipher so that it cannot be dismantled. Meanwhile, RC4 is a byte-oriented stream cipher system, then it performs XOR operations with a byte key by generating cipher bytes.

a. Hill cipher cryptography

1) Encryption process on hill cipher

Hill cipher is a symmetric key algorithm that falls under the category of polygraphic substitution ciphers. It operates on blocks of plaintext, typically groups of two or three letters. The key for the hill cipher is a matrix. Cryptographic analysis using the 3×3 hill cipher encryption method is as:

Plaintext: “HARIMAUSUMATERA”

Key: “O R A N G U T A N”

If there is Plaintext “HARIMAUSUMATERA” and the key “ORANGUTAN” then it must be converted first to:

Plaintext: “7 0 17 8 12 0 20 18 20 12 0 19 4 17 0”

Key: “141701362019013”

From the plaintext it can be added to the key K through the:

$$C_{1,2,3} = \begin{pmatrix} 14 & 17 & 0 \\ 13 & 6 & 20 \\ 19 & 0 & 13 \end{pmatrix} \begin{pmatrix} 7 \\ 0 \\ 17 \end{pmatrix} = \begin{pmatrix} 14 \text{ x } 7 & + & 17 \text{ x } 0 & + & 0 \text{x } 17 \\ 13 \text{ x } 7 & + & 6 \text{ x } 0 & + & 20 \text{ x } 17 \\ 19 \text{ x } 7 & + & 0 \text{ x } 0 & + & 13 \text{ x } 17 \end{pmatrix}$$

$$= \begin{pmatrix} 98 \\ 431 \\ 354 \end{pmatrix} (\text{mod } 26) \begin{pmatrix} 20 \\ 15 \\ 16 \end{pmatrix}$$

$$= \begin{pmatrix} 20 \\ 15 \\ 16 \end{pmatrix} = \begin{pmatrix} \text{U} \\ \text{P} \\ \text{Q} \end{pmatrix}$$

The encryption characters corresponding to 7, 0, and 17 are U, P and Q. Then the HAR characters in the plaintext change to the U P Q characters in the ciphertext. The third and fourth characters have the result of calculating numbers that do not correspond to the letters, so they must be modulo 26 in the result.

$$C_{4,5,6} = \begin{pmatrix} 14 & 17 & 0 \\ 13 & 6 & 20 \\ 19 & 0 & 13 \end{pmatrix} \begin{pmatrix} 8 \\ 12 \\ 0 \end{pmatrix} = \begin{pmatrix} 14 \text{ x } 8 & + & 17 \text{ x } 12 & + & 0 \text{ x } 0 \\ 13 \text{ x } 8 & + & 6 \text{ x } 12 & + & 20 \text{ x } 0 \\ 19 \text{ x } 8 & + & 0 \text{ x } 12 & + & 13 \text{ x } 0 \end{pmatrix}$$

$$= \begin{pmatrix} 316 \\ 176 \\ 152 \end{pmatrix} (\text{mod } 26) \begin{pmatrix} 4 \\ 20 \\ 22 \end{pmatrix}$$

$$= \begin{pmatrix} 4 \\ 20 \\ 22 \end{pmatrix} = \begin{pmatrix} \text{E} \\ \text{U} \\ \text{W} \end{pmatrix}$$

$$C_{7,8,9} = \begin{pmatrix} 14 & 17 & 0 \\ 13 & 6 & 20 \\ 19 & 0 & 13 \end{pmatrix} \begin{pmatrix} 20 \\ 18 \\ 20 \end{pmatrix} = \begin{pmatrix} 14 \text{ x } 20 & + & 17 \text{ x } 18 & + & 0 \text{ x } 20 \\ 13 \text{ x } 20 & + & 6 \text{ x } 18 & + & 20 \text{ x } 20 \\ 19 \text{ x } 20 & + & 0 \text{ x } 18 & + & 13 \text{ x } 20 \end{pmatrix}$$

$$= \begin{pmatrix} 586 \\ 768 \\ 640 \end{pmatrix} (\text{mod } 26) \begin{pmatrix} 14 \\ 14 \\ 16 \end{pmatrix}$$

$$= \begin{pmatrix} 14 \\ 14 \\ 16 \end{pmatrix} = \begin{pmatrix} \text{O} \\ \text{O} \\ \text{Q} \end{pmatrix}$$

$$C_{10,11,12} = \begin{pmatrix} 14 & 17 & 0 \\ 13 & 6 & 20 \\ 19 & 0 & 13 \end{pmatrix} \begin{pmatrix} 12 \\ 0 \\ 19 \end{pmatrix} = \begin{pmatrix} 14 \text{ x } 12 & + & 17 \text{ x } 0 & + & 0 \text{ x } 19 \\ 13 \text{ x } 12 & + & 6 \text{ x } 0 & + & 20 \text{ x } 19 \\ 19 \text{ x } 12 & + & 0 \text{ x } 0 & + & 13 \text{ x } 19 \end{pmatrix}$$

$$= \begin{pmatrix} 168 \\ 536 \\ 475 \end{pmatrix} (\text{mod } 26) \begin{pmatrix} 12 \\ 16 \\ 7 \end{pmatrix}$$

$$= \begin{pmatrix} 12 \\ 16 \\ 7 \end{pmatrix} = \begin{pmatrix} \text{M} \\ \text{Q} \\ \text{H} \end{pmatrix}$$

$$C13{,}14{,}15 = \begin{pmatrix} 14 & 17 & 0 \\ 13 & 6 & 20 \\ 19 & 0 & 13 \end{pmatrix}\begin{pmatrix} 4 \\ 17 \\ 0 \end{pmatrix} = \begin{pmatrix} 14 \times 4 & + & 17 \times 17 & + & 0 \times 0 \\ 13 \times 4 & + & 6 \times 17 & + & 20 \times 0 \\ 19 \times 4 & + & 0 \times 17 & + & 13 \times 0 \end{pmatrix}$$

$$= \begin{pmatrix} 345 \\ 154 \\ 76 \end{pmatrix} (\text{mod } 26) \begin{pmatrix} 7 \\ 24 \\ 24 \end{pmatrix}$$

$$= \begin{pmatrix} 7 \\ 24 \\ 24 \end{pmatrix} = \begin{pmatrix} \text{H} \\ \text{Y} \\ \text{Y} \end{pmatrix}$$

Applying the encryption process to the original text (plain text) “HARIMAUSUMATRA” using the hill cipher method, obtains an encrypted text (ciphertext) which reads “UPQEUWOOQMQHHYY”. The hill cipher method uses linear algebra principles to convert blocks of text into blocks of encrypted text, producing a ciphertext that does not show any obvious or easily recognizable patterns from the original text.

2) Decryption process on hill cipher

To carry out cryptographic analysis and decrypt messages that have been encrypted using the hill cipher method, the first step is to change the encryption key in the form of the word “ORANGUTAN” into a series of numbers according to the position of the letters in it, where A=0, B=1, …, Z = 25. The key “ORANGUTAN” is converted into a set of numbers K= (14,17,0,13,6,20,19, 0, 13), based on the alphabet.

$$K^{-1} = \begin{pmatrix} 14 & 17 & 0 \\ 13 & 6 & 20 \\ 19 & 0 & 13 \end{pmatrix}^{-1} = \begin{pmatrix} 78 & -221 & 340 \\ 211 & 182 & -280 \\ -144 & 323 & -137 \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix} \text{ mod } 26 \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix}$$

In proving the inverse of matrix K with inverse K-1, the proof step is carried out by performing a multiplication operation between matrix K and its inverse, namely K-1. The goal of this step is to ensure that the multiplication produces an identity matrix. The success of producing the identity matrix after the multiplication operation is proof of the validity that K-1 is the appropriate inverse for the K matrix, thus reaffirming the truth of the inverse relationship between the two, the proof is as:

$$K = \begin{pmatrix} 14 & 17 & 0 \\ 13 & 6 & 20 \\ 19 & 0 & 13 \end{pmatrix} \quad K^{-1} = \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix}$$

$$K.K^{-1} = \begin{pmatrix} 14 & 17 & 0 \\ 13 & 6 & 20 \\ 19 & 0 & 13 \end{pmatrix} \text{ x } \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix}$$

$$= \begin{pmatrix} 14x0 + 17x23 + 0x12 & 14x13 + 17x0 + 0x15 & 14x24 + 17x6 + 0x19 \\ 13x0 + 6x23 + 20x12 & 13x13 + 6x0 + 20x15 & 13x24 + 6x6 + 20x19 \\ 19x0 + 0x23 + 13x12 & 19x13 + 0x0 + 13x15 & 19x24 + 0x6 + 13x19 \end{pmatrix}$$

$$= \begin{pmatrix} 391 & 182 & 438 \\ 378 & 469 & 728 \\ 156 & 442 & 703 \end{pmatrix} \text{ mod } 26 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

In the context of information security, the process of decrypting an encrypted note or data requires the initial step, namely changing the ciphertext that has been generated during the encryption process. The process of decrypting the record must first change the ciphertext C = “UPQEUWOOQMQHHYY” to letters in numerical sequence C = “20 15 16 4 20 22 14 14 16 12 16 7 7 24 24”.

First record:

$$P_{1,2,3} = \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix}\begin{pmatrix} 20 \\ 15 \\ 16 \end{pmatrix} = \begin{pmatrix} 579 \\ 546 \\ 771 \end{pmatrix} \text{ mod } 26 \begin{pmatrix} 7 \\ 0 \\ 17 \end{pmatrix} = \begin{pmatrix} \text{H} \\ \text{A} \\ \text{R} \end{pmatrix}$$

Second record:

$$P_{4,5,6} = \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix}\begin{pmatrix} 4 \\ 20 \\ 22 \end{pmatrix} = \begin{pmatrix} 788 \\ 220 \\ 754 \end{pmatrix} \text{ mod } 26 \begin{pmatrix} 8 \\ 12 \\ 0 \end{pmatrix} = \begin{pmatrix} \text{I} \\ \text{M} \\ \text{A} \end{pmatrix}$$

$$P_{7,8,9} = \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix}\begin{pmatrix} 14 \\ 14 \\ 16 \end{pmatrix} = \begin{pmatrix} 566 \\ 434 \\ 670 \end{pmatrix} \text{s mod } 26 \begin{pmatrix} 20 \\ 18 \\ 20 \end{pmatrix} = \begin{pmatrix} \text{U} \\ \text{S} \\ \text{U} \end{pmatrix}$$

$$P10{,}11{,}12 = \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix} \begin{pmatrix} 12 \\ 16 \\ 7 \end{pmatrix} = \begin{pmatrix} 376 \\ 312 \\ 513 \end{pmatrix} \bmod 26 \begin{pmatrix} 12 \\ 0 \\ 19 \end{pmatrix} = \begin{pmatrix} \mathrm{M} \\ \mathrm{A} \\ \mathrm{T} \end{pmatrix}$$

$$P13{,}14{,}15 = \begin{pmatrix} 0 & 13 & 24 \\ 23 & 0 & 6 \\ 12 & 15 & 19 \end{pmatrix} \begin{pmatrix} 7 \\ 24 \\ 24 \end{pmatrix} = \begin{pmatrix} 888 \\ 303 \\ 884 \end{pmatrix} \bmod 26 \begin{pmatrix} 4 \\ 17 \\ 0 \end{pmatrix} = \begin{pmatrix} \mathrm{E} \\ \mathrm{R} \\ \mathrm{A} \end{pmatrix}$$

After all records have been decrypted, the plaintext results are obtained: C = "20 15 16 4 20 22 14 14 16 12 16 7 7 24 24" P = "HARIMAUSUMATERA".

b. Cryptography RC4

RC4 is a type of symmetric stream cipher encryption that has been widely used in various applications to secure data transmission. To understand how RC4 works, use a case example by taking the text "TIGER" as the data you want to encrypt, and using "LEUSER" as the encryption key. RC4 encrypts this data into a ciphertext that cannot be read without the appropriate key.

In Table 1 there are the results of cryptographic analysis carried out using the RC4 algorithm. This algorithm has been implemented to secure data by generating random keys that are used in the encryption process. The analysis includes an evaluation of the performance of the algorithm, the strength of the encryption provided, and the possible vulnerability to cryptanalysis attacks.

1) Initialize an S-box with a length of 256 bytes, with S[0]=0, S[1]=1, S[2]=2, S[3]=3,....., S[255] =255 so array S becomes:

Table 1. S-box

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

2) Initialize the 7-byte Ki array key, for example the key consists of 7 bytes namely "LEUSER" then the sentence to be converted into decimal form "76 69 85 83 69 82". Repeat the key until it fills the entire K array so that the K array becomes something different, as shown in Table 2.

3) Next mixes the operation which will apply variables i and j to the index array $S[i]$ and $K[i]$. The first step is initialized for i and j with 0. The mixing operation is repeating the formula ($j + S[i] + K[i]$) mod 256 followed by swapping $S[i]$ for $S[j]$. Because it uses an array with a length of 256 bytes the algorithm becomes:

$$For\ i = 0\ to\ 256$$
$$j = (j + S[i] + K[i] mod\ 256$$
$$Swap\ S[i] dan\ S[j]$$

1st iteration:

$i = 0, then$

$j = (j + S[i] + K[i]) mod\ 256$

$= (j + S[0] + K[0]) mod\ 256$

$= (0 + 0 + 76) mod\ 256$

$= 76$

$Swap\ S[0] and\ S[76]$

2nd iterarion

$i = 1, then$

$j = (j + S[i] + K[i]) mod\ 256$

$= (j + S[1] + K[1]) mod\ 256$

$= (76 + 1 + 69) mod\ 256$

$= 146$

$Swap\ S[1] and\ S[146]$

3rd iterarion

$i = 2, then$

$j = (j + S[i] + K[i]) mod\ 256$

$= (j + S[2] + K[2]) mod\ 256$

$= (146 + 2 + 85) mod\ 256$

$= 233$

$Swap\ S[2] and\ S[233]$

4th iteration:
$i = 3, then$
$j = (j + S[i] + K[i]) mod\ 256$
$= (j + S[3] + K[3]) mod\ 256$
$= (233 + 3 + 83) mod\ 256$
$= 63$
$Swap\ S[3] and\ S[76]$

5th iterarion
$i = 4, then$
$j = (j + S[i] + K[i]) mod\ 256$
$= (j + S[4] + K[4]) mod\ 256$
$= (63 + 4 + 69) mod\ 256$
$= 136$
$Swap\ S[4] and\ S[146]$

6th iterarion
$i = 5, then$
$j = (j + S[i] + K[i]) mod\ 256$
$= (j + S[5] + K[5]) mod\ 256$
$= (136 + 5 + 82) mod\ 256$
$= 223$
$Swap\ S[5] and\ S[223]$

7th iteration:
$i = 3, then$
$j = (j + S[i] + K[i]) mod\ 256$
$= (j + S[6] + K[6]) mod\ 256$
$= (223 + 6 + 49) mod\ 256$
$= 49$
$Swap\ S[6] and\ S[49]$

……….
……….
……..…
……….
……….

254th iterarion
$i = 253, then$
$j = (j + S[i] + K[i]) mod\ 256$
$= (j + S[253] + K[253]) mod\ 256$
$= (242 + 253 + 69) mod\ 256$
$= 52$
$Swap\ S[253] and\ S[52]$

255th iteration:
$i = 254, then$
$j = (j + S[i] + K[i]) mod\ 256$
$= (j + S[254] + K[254]) mod\ 256$
$= (52 + 254 + 85) mod\ 256$
$= 135$
$Swap\ S[254] and\ S[135]$

256th iterarion
$i = 255, then$
$j = (j + S[i] + K[i]) mod\ 256$
$= (j + S[255] + K[255]) mod\ 256$
$= (135 + 255 + 83) mod\ 256$
$= 217$
$Swap\ S[255] and\ S[217]$

Table 2. Array initialization

Inter-i	Key-char	Key[i]	Sbox[i]
1	L	76	0
2	E	69	1
3	U	85	2
4	S	83	3
5	E	69	4
6	R	82	5
7	L	76	6
8	E	69	7
9	U	85	8
10	S	83	9
11	E	69	10
12	R	82	11
13	L	76	12
14	E	69	13
15	U	85	14
16	S	83	15
17	E	69	16
18	R	82	17
19	L	76	18
20	E	69	19
21	U	85	20
...			
256	S	83	255

Iterations are carried out to perfect each step in the process, while S-box exchange (swap) becomes a key strategy in increasing resistance to possible attacks. After carrying out iterations from 0 to 255 iterations and S-box exchanges, the results obtained after carrying out all iterations from 0 to 255 iterations and S-box exchanges (swaps) are shown in Table 3. RC4 encryption is an encryption process that is XORing bytes with the plaintext "HARIMAU". Plaintext consists of 7 characters, so 7 iterations occur. The previous iteration must be converted into binary form as shown in Table 4.

Table 3. Iteration results 0/256

76	146	233	63	136	223	49	125	218	54	133	226	58	140	239
81	166	9	42	103	191	40	144	235	84	184	22	133	243	84
195	45	145	6	122	225	86	198	48	160	9	135	249	101	227
89	212	68	201	66	181	58	183	57	176	60	188	54	194	70
207	77	224	103	232	123	6	150	27	181	67	203	101	247	142
26	187	80	223	128	25	183	74	242	142	36	204	108	17	171
90	253	154	73	240	156	61	243	157	65	247	165	88	0	189
110	25	214	139	69	244	184	112	34	230	162	99	25	228	163
92	39	234	178	111	65	7	199	153	99	50	246	270	156	99
60	13	227	174	142	98	48	16	232	197	151	126	89	46	21
244	216	177	159	129	93	75	49	28	252	241	218	189	178	159
145	120	116	100	78	74	62	55	37	40	31	16	19	14	14
3	13	11	3	13	15	22	18	35	40	39	56	65	79	82
106	18	124	148	164	185	195	226	245	2	33	56	84	101	139
165	185	223	253	32	56	101	134	161	206	243	29	60	112	152
186	238	26	75	113	172	219	4	63	114	170	215	25	79	127
193	58	110	183	244	43	116	181	251	170	242	202	52	135	217

Table 4. Character results to binary numbers

Plaintext	Decimal	Biner
H	72	01001000
A	65	01000001
R	82	01010010
I	73	01001001
M	77	01001101
A	82	01010010
U	65	01000001

In the initial step of the algorithm, the initial values of variables i and j are initialized as 0. This initialization is a critical step that prepares the algorithm's iterative process for key setting and data randomization. Initialize i and j with $i = 0; j = 0$.

1st iteration:

$$i = (i+1) mod\ 256 = (0+1) mode\ 256 = 1$$

$$j = (j + S[1] mod\ 256 = (0 + S[1] mod 256 = (0+146) mod 256 = 146$$

$$Swaps(S[1] and\ S[146]$$

$$t = (S[i] + S[j] mod 256 = S[1] + S[146] mod 256 = (146+81) mod 256 = 227$$

$$Keys[0] = S[227] = 246$$

2nd iteration:

$$i = (i+1) mod\ 256 = (1+1) mode\ 256 = 2$$

$$j = (j + S[i] mod\ 256 = (146 + S[2] mod 256 = (146+233) mod 256 = 123$$

$$Swaps(S[2] and\ S[123]$$

$$t = (S[i] + S[j] mod 256 = S[2] + S[123] mod 256 = (233+67) mod 256 = 44$$

$$Keys[1] = S[44] = 166$$

3rd iteration:

$$i = (i+1) mod\ 256 = (2+1) mode\ 256 = 3$$

$$j = (j + S[1] mod\ 256 = (123 + S[3] mod 256 = (123+63) mod 256 = 186$$

$$Swaps(S[3] and\ S[186]$$

$$t = (S[i] + S[j] mod 256 = S[1] + S[146] mod 256 = (63+107) mod 256 = 170$$

$$Keys[2] = S[170] = 109$$

4th iteration:

$$i = (i+1) mod\ 256 = (3+1) mode\ 256 = 4$$

$$j = (j + S[i] mod\ 256 = (186 + S[4] mod 256 = (186+136) mod 256 = 66$$

$$Swaps(S[4] and\ S[66]$$

$$t = (S[i] + S[j] mod 256 = S[4] + S[66] mod 256 = (136+223) mod 256 = 103$$

$$Keys[3] = S[103] = 77$$

5th iteration:

$$i = (i+1) mod\ 256 = (4+1) mode\ 256 = 5$$

$$j = (j + S[1] mod\ 256 = (66 + S[5] mod 256 = (66+223) mod 256 = 33$$

$$Swaps(S[5] and\ S[33]$$

$$t = (S[i] + S[j] mod 256 = S[5] + S[33] mod 256 = (223+122) mod 256 = 89$$

$$Keys[4] = S[89] = 213$$

6th iteration:

$$i = (i+1) mod\ 256 = (5+1) mode\ 256 = 6$$

$$j = (j + S[i] mod\ 256 = (33 + S[6] mod 256 = (33+49) mod 256 = 82$$

$$Swaps(S[6] and\ S[82]$$

$$t = (S[i] + S[j] mod 256 = S[6] + S[82] mod 256 = (49+89) mod 256 = 138$$

$$Keys[6] = S[138] = 115$$

7th iteration:

$$i = (i+1) mod\ 256 = (6+1) mode\ 256 = 7$$

$$j = (j + S[1] mod\ 256 = (82 + S[7] mod 256 = (82+125) mod 256 = 207$$

$$Swaps(S[7] and\ S[207]$$

$$t = (S[i] + S[j] mod 256 = S[7] + S[207] mod 256 = (125+241) mod 256 = 110$$

$$Keys[6] = S[110] = 95$$

By performing seven iterations using the "TIGER" key, the results were recorded and presented in Table 5. This table provides a detailed description of the key changes after each iteration, showing sequential data transformations. After successfully finding the key for each character, the next step is the XOR operation between the character in the plaintext and the key that has been generated. This process is an integral part of the encryption algorithm, where XOR is used to combine information from both sources and produce a ciphertext that cannot be easily reconstructed without knowing the correct key. The resulting key is as shown in Table 6.

The decryption process is XORing pseudorandom bytes with the ciphertext being % Ҁ? - ȳ ä - . Ciphertext consists of 7 iterations which will be converted into characters in the form of binary numbers as in the data contained in Table 7.

Table 5. Key formation results

Index	Key	Biner
0	246	11110110
1	166	10100110
2	109	01101101
3	77	01001101
4	213	11010101
5	155	10011011
6	95	01011111

Table 6. XOR iterasi

Index	P XOR K	Ciphertext (C)	Des (C)	ASCI
0	01001000 θ 11110110	10111110	190	%
1	01000001 θ 10100110	11100111	231	Ҁ
2	01010010 θ 01101101	00111111	63	?
3	01001001 θ 01001101	00000100	4	-
4	01001101 θ 11010101	10011000	152	ȳ
5	01010010 θ 10011011	11001001	201	ä
6	01000001 θ 01011111	00011110	30	-

Table 7. XOR iterasi

Index	C XOR K	Ciphertext (P)	Des (P)	ASCI
0	11110110 θ10111110	01001000	72	H
1	10100110 θ11100111	01000001	65	A
2	01101101 θ 00111111	01010010	82	R
3	01001101 θ 00000100	01001001	73	I
4	11010101 θ 10011000	01001101	77	M
5	10011011 θ 11001001	01010010	82	A
6	01011111 θ 00011110	01000001	65	U

4. CONCLUSION

Hill cipher is a cryptographic substitution encryption method that works with matrices. This method converts bright text letter blocks into cipher letter blocks using the key matrix. This method is capable of encrypting messages in block form, so it can overcome some of the vulnerabilities in the simple substitution method. This method is more powerful than the simple substitution method because the correlations between letters are not clearly visible in encrypted text. RC4 is a cryptographic flow algorithm that uses a key to generate a stream of random bytes, which are used to encrypt messages. RC4 is an algorithm that is fast enough to perform encryption and decryption. and this algorithm is relatively easy to understand and implement. Both hill cipher and RC4 have their own strengths and weaknesses. However, at this time, RC4 is generally considered less secure for use in security-critical scenarios because of its vulnerability to attacks. In practice, for higher data security, it is recommended to use modern and stronger encryption algorithms such as advanced encryption standard (AES) which has been tested and proven to be more robust against modern attacks.

REFERENCES

[1] V. Adat and B. B. Gupta, "Security in Internet of Things: issues, challenges, taxonomy, and architecture," *Telecommunication Systems*, vol. 67, no. 3, pp. 423–441, 2018, doi: 10.1007/s11235-017-0345-9.

[2] S. Ghoul, R. Sulaiman, and Z. Shukur, "A Review on Security Techniques in Image Steganography," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 6, pp. 361–385, 2023, doi: 10.14569/IJACSA.2023.0140640.

[3] C. Gong *et al.*, "Grover algorithm-based quantum homomorphic encryption ciphertext retrieval scheme in quantum cloud

computing," *Quantum Information Processing*, vol. 19, no. 3, p. 105, Mar. 2020, doi: 10.1007/s11128-020-2603-0.
[4] H. Y. Lin and Y. R. Jiang, "A multi-user ciphertext policy attribute-based encryption scheme with keyword search for medical cloud system," *Applied Sciences (Switzerland)*, vol. 11, no. 1, pp. 1–14, Dec. 2021, doi: 10.3390/app11010063.
[5] S. Y. Wulandari, "Cryptography: A Combination of Caesar and Affine Cipher to Conceal the Message," *Proceeding International Conference on Science and Engineering*, vol. 3, pp. 741–744, Apr. 2020, doi: 10.14421/icse.v3.595.
[6] K. Prasad and H. Mahato, "Cryptography using generalized Fibonacci matrices with Affine-Hill cipher," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 25, no. 8, pp. 2341–2352, Nov. 2022, doi: 10.1080/09720529.2020.1838744.
[7] R. K. Hasoun, S. F. Khlebus, and H. K. Tayyeh, "A new approach of classical hill cipher in public key cryptography," *International Journal of Nonlinear Analysis and Applications*, vol. 12, no. 2, pp. 1071–1082, 2021, doi: 10.22075/ijnaa.2021.5176.
[8] H. Touil, N. E. L. Akkad, and K. Satori, "Text Encryption: Hybrid cryptographic method using Vigenere and Hill Ciphers.," in *2020 International Conference on Intelligent Systems and Computer Vision, ISCV 2020*, IEEE, Jun. 2020, pp. 1–6, doi: 10.1109/ISCV49265.2020.9204095.
[9] Nurhayati, A. Meizar, F. Tambunan, and E. Ginting, "Optimizing the Complexity of Time in the Process of Multiplying Matrices in the Hill Cipher Algorithm Using the Strassen Algorithm," in *2019 7th International Conference on Cyber and IT Service Management, CITSM 2019*, IEEE, Nov. 2019, pp. 1–4, doi: 10.1109/CITSM47753.2019.8965416.
[10] N. Aragon, P. Gaborit, A. Hauteville, O. Ruatta, and G. Zemor, "Low Rank Parity Check Codes: New Decoding Algorithms and Applications to Cryptography," *IEEE Transactions on Information Theory*, vol. 65, no. 12, pp. 7697–7717, Dec. 2019, doi: 10.1109/TIT.2019.2933535.
[11] F. Pitolli, "A fractional b-spline collocation method for the numerical solution of fractional predator-prey models," *Fractal and Fractional*, vol. 2, no. 1, pp. 1–16, Feb. 2018, doi: 10.3390/fractalfract2010013.
[12] C. L. Chowdhary, P. V. Patel, K. J. Kathrotia, M. Attique, P. Kumaresan, and M. F. Ijaz, "Analytical study of hybrid techniques for image encryption and decryption," *Sensors (Switzerland)*, vol. 20, no. 18, pp. 1–19, Sep. 2020, doi: 10.3390/s20185162.
[13] Y. Zhang, "The unified image encryption algorithm based on chaos and cubic S-box," *Information Sciences*, vol. 450, pp. 361–377, Jun. 2018, doi: 10.1016/j.ins.2018.03.055.
[14] L. Teng, X. Wang, and J. Meng, "A chaotic color image encryption using integrated bit-level permutation," *Multimedia Tools and Applications*, vol. 77, no. 6, pp. 6883–6896, Mar. 2018, doi: 10.1007/s11042-017-4605-1.
[15] Y. Luo, X. Ouyang, J. Liu, and L. Cao, "An Image Encryption Method Based on Elliptic Curve Elgamal Encryption and Chaotic Systems," *IEEE Access*, vol. 7, pp. 38507–38522, 2019, doi: 10.1109/ACCESS.2019.2906052.
[16] D. Liestyowati, "Public Key Cryptography," *Journal of Physics: Conference Series*, vol. 1477, no. 5, p. 052062, Mar. 2020, doi: 10.1088/1742-6596/1477/5/052062.
[17] A. A. Yazdeen, S. R. M. Zeebaree, M. M. Sadeeq, S. F. Kak, O. M. Ahmed, and R. R. Zebari, "FPGA Implementations for Data Encryption and Decryption via Concurrent and Parallel Computation: A Review," *Qubahan Academic Journal*, vol. 1, no. 2, pp. 8–16, Mar. 2021, doi: 10.48161/qaj.v1n2a38.
[18] M. S. Novelan, A. M. Husein, M. Harahap, and S. Aisyah, "SMS Security System on Mobile Devices Using Tiny Encryption Algorithm," *Journal of Physics: Conference Series*, vol. 1007, no. 1, p. 012037, Apr. 2018, doi: 10.1088/1742-6596/1007/1/012037.
[19] O. Laia, E. M. Zamzami, Sutarman, F. G. N. Larosa, and A. Gea, "Application of Linear Congruent Generator in Affine Cipher Algorithm to Produce Dynamic Encryption," *Journal of Physics: Conference Series*, vol. 1361, no. 1, p. 012001, Nov. 2019, doi: 10.1088/1742-6596/1361/1/012001.
[20] R. Hamza, Z. Yan, K. Muhammad, P. Bellavista, and F. Titouna, "A privacy-preserving cryptosystem for IoT E-healthcare," *Information Sciences*, vol. 527, pp. 493–510, Jul. 2020, doi: 10.1016/j.ins.2019.01.070.
[21] J. Shen, H. Ji, and J. Han, "Near-imperceptible neural linguistic steganography via self-adjusting arithmetic coding," in *EMNLP 2020 - 2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 303–313, doi: 10.18653/v1/2020.emnlp-main.22.
[22] M. Alruily, O. R. Shahin, H. Al-Mahdi, and A. I. Taloba, "Asymmetric DNA encryption and decryption technique for Arabic plaintext," *Journal of Ambient Intelligence and Humanized Computing*, Apr. 2021, doi: 10.1007/s12652-021-03108-w.
[23] E. Osaba, J. D. Ser, A. Sadollah, M. N. Bilbao, and D. Camacho, "A discrete water cycle algorithm for solving the symmetric and asymmetric traveling salesman problem," *Applied Soft Computing Journal*, vol. 71, pp. 277–290, Oct. 2018, doi: 10.1016/j.asoc.2018.06.047.
[24] M. A. Al-Shabi, "A Survey on Symmetric and Asymmetric Cryptography Algorithms in information Security," *International Journal of Scientific and Research Publications (IJSRP)*, vol. 9, no. 3, p. p8779, Mar. 2019, doi: 10.29322/ijsrp.9.03.2019.p8779.
[25] J. R. Paragas, A. M. Sison, and R. P. Medina, "Hill cipher modification: A simplified approach," in *2019 IEEE 11th International Conference on Communication Software and Networks, ICCSN 2019*, IEEE, Jun. 2019, pp. 821–825, doi: 10.1109/ICCSN.2019.8905360.
[26] E. J. Madarro-Capó, C. M. Legón-Pérez, O. Rojas, and G. Sosa-Gómez, "Measuring avalanche properties on RC4 stream cipher variants," *Applied Sciences (Switzerland)*, vol. 11, no. 20, p. 9646, Oct. 2021, doi: 10.3390/app11209646.

BIOGRAPHIES OF AUTHORS

Azanuddin is a lecturer at the Computer Engineering Department of Computer Engineering and Informatics, Medan State Polytechnic, Indonesia since 2022, before is lacturer at the Budi Darma University and STMIK Triguna Dharma since 2013-2022. He received her S.Kom in Informatics Engineering from Budi Darma University, Indonesia in 2011 and his M.Kom degree in Computer Science from Putra Indonesia YPTK Padang University Indoneisa in 2013. His research interests include network security, data security, encryption and decryption algorithms, mobile programming, and cloud computing. He can be contacted at email: azanuddin@polmed.ac.id.

Rikie Kartadie he received his Master of Computer Science from Universiti Amikom Yogyakarta (2014). Now a lecturer in Computer Science at the Information Systems Study Program, now a lecturer at the Department of Computer Engineering, Universitas Technologi Digital Indonesia. In addition, he also has an assessment of the national professional certification agency, computer network competence, he has a professional lecturer certification from the Ministry of Research, Technology, and Higher Education. In 2018 he was assigned by the Ministry of Education, Culture, Research and Technology as a lecturer workload assessor. He is also one of the reviewers of the SINTA 3, 4 indexed national journals. He has published more than 35 journal papers, 1 written book, and 1 journal scopus paper. His research interests are networking, artificial intelligence, internet of thing, and data mining. He can be contacted at email: rikie@utdi.ac.id.

Fauzi Erwis currently working on Doctorate (Doctorate in Vocational Technology Education) at the Faculty of Engineering from Padang State University starting in 2020. He is a lecturer in Information Technology Education Universitas Rokania, Rokan Hulu, Indonesia. He was head lectures at Information Technology Education from 2019 to 2020. His research interests are decision support systems, artificial intelligence, and expert system. He can be contacted at email: fauzierwis@gmail.com.

Ahmad Fitri Boy he earned a postgraduate degree in the Faculty of Computer Science from UPI (Universitas Putera Indonesia) YPTK Padang in 2009. He is a Computer Science lecturer in Information Systems at the STMIK Triguna Dharma Institute, Medan, Indonesia. In addition, he also served as head of the Informatics Management Program and Deputy Chair III of the Institute at STMIK Triguna Dharma from 2017 to 2022. Since 2018, he has a professional lecturer certification from the Ministry of Research, Technology and Higher Education. He has published more than 10 journal papers, 2 HKI, and 2 papers in conference proceedings. His research interests are decision support systems, artificial intelligence, and data mining. He can be contacted via email: ahmadfitriboy@gmail.com.

Asyahri Hadi Nasyuha he received his Doctorate (Doctorate in Vocational Technology Education) at the Faculty of Engineering from Padang State University in 2020. He is a lecturer in Computer Science at the Information Systems Study Program, Universitas Teknologi Digital Indonesia, Yogyakarta, Indonesia. In addition, he has also served as head of the Quality Assurance Institute at STMIK Triguna Dharma from 2020 to 2022. Since 2019, he has a professional lecturer certification from the Ministry of Research, Technology and Higher Education. In 2021 he was assigned by the Ministry of Education, Culture, Research and Technology as a lecturer workload assessor. In 2022, he will graduate as an assessor for the independent accreditation agency INFOKOM. He is also one of the reviewers of the SINTA 3 indexed national journal. He has published more than 100 articles in national and international journals, 3 books, 2 intellectual property rights and several articles in conference proceedings. His research interests are decision support systems, artificial intelligence, and data mining. He can be contacted at email: asyahrihadi@gmail.com.