JOSCEX Journal of Soft Computing Exploration

Simulations of text encryption and decryption by applying vertical bit rotation algorithm

Dwika Ananda Agustina Pertiwi[1], Djuniadi[2]
[1]Department of Computer Science, Universitas Negeri Semarang, Indonesia
[2]Faculty of Engineering, Universitas Negeri Semarang, Indonesia

Article Info

Article history:

Received Jan 14, 2021
Revised Feb 20, 2021
Accepted Aug 29, 2021

Keywords:

Cryptography
Vertical bit
Rotation algorithm
Text encryption

ABSTRACT

Cryptography is the study of hiding text and numbers in the form of codes. Vertical Bit Rotation (VBR) is one of the most widely implemented cryptographic algorithms as a one-way hash function that simplifies the encryption process with a high degree of difficulty in decryption. The purpose of this study is to apply VBR hash algorithm modeling to binary value characters with bit rotation keys 10, 11, 7, 3, 2, 7, 5, and 4. Thus, generating a passcode. The results of the encryption simulation show the code in the form of letters and characters, then the result of the decryption with the opposite rotation to the encryption process returns the value from ciphertext to plaintext based on ASCII characters. Cryptographic algorithms are applied to avoid cryptanalytic experiments in opening encryption codes.

Corresponding Author:

Dwika Ananda Agustina Pertiwi,
Department of Computer Science, Universitas Negeri Semarang, Indonesia.
Email: dwikapertiwi13@gmail.com

1. INTRODUCTION

The development of digitalization and people's habits in communicating have an impact on the emergence of data security threats [1]. Humans began to look for solutions to avoid the threat of digitization such as experiments on cryptanalysis of encryption codes. Encryption and decryption methods are needed as a way of securing data or called cryptography.

Cryptography is a study or art that studies ways and techniques to keep messages or information secret in the form of passwords or numeric codes with various encryption methods and algorithms [2]. Encryption is an idea that is applied as risk management in authentication and integrity issues that are now widely applied [3]. Cryptography first appeared applying a symmetric algorithm or the so-called secret key algorithm or secret cipher. Where, this algorithm also has the same encryption key as the decryption key [4].

Encryption and decryption are the main functions of cryptography. The benefits of these two functions are data security against those who do not have the authority to access information. The encryption process converts plain text into a complex code called ciphertext. The decryption process returns or changes the ciphertext to the original text form [5].

Regarding the security of the world of computer technology, there are several security aspects, including [6] Availability is related to the availability of data and information systems when needed. Integrity is the protection of data so that it cannot be changed by unauthorized users. Privacy/confidentiality is to protect data and information from threats and disturbances from unauthorized users.

The hash function is an encrypted code consisting of numbers and letters at random. One of the hash functions is the Vertical Bit Rotation Algorithm [7]. The Vertical Bit Rotation (VBR) algorithm is a symmetric algorithm, which uses a 64-bit key and has 32 rounds to process each encrypt or decrypt [8], [9]. The Clipper-Chip is a commercial chip made by the NSA for encryption and uses the VBR algorithm.

In the encryption simulation by applying the Vertical Bit Rotation algorithm modeling using bit rotation keys 10, 11, 7, 3, 2, 7, 5, and 4 with text characters in binary form. Then, the decryption simulation with the rotation of the bit opposite the encryption process will produce a binary value that will be converted into ASCII characters. Research related to the application of the Vertical Bit Rotation algorithm as an encryption code algorithm for online files [10]. The purpose of this study is the implementation of the Vertical Bit Rotation hash algorithm on text files with bit rotation keys 10, 11, 7, 3, 2, 7, 5, 4.

2. METHOD

This study uses a methodology or simulation method which can be seen in Figure 1.

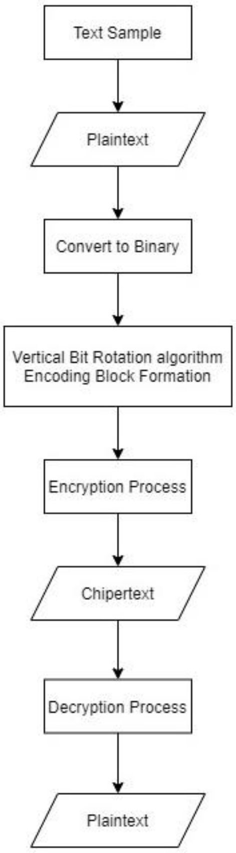


Figure 1. Simulations method

The sample text is converted into binary before the encryption process is carried out, the encryption technique uses the VBR algorithm with a bit key that is created as a simulation. Then the ciphertext generated from the encryption process will be returned to plaintext in the decryption process.

Vertical Bit Rotation is a cryptographic algorithm that belongs to the type of symmetric cryptography in the block cipher category [11]. Block cipher is a form of bit block cryptographic algorithm that operates on plaintext/ciphertext. The VBR algorithm uses blocks with a maximum size of 256 bytes [12]. The flow of the symmetric cryptography simulation can be seen in Figure 2.

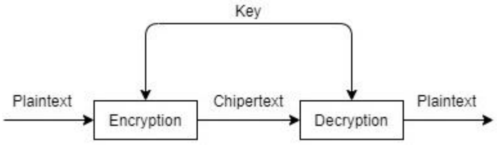
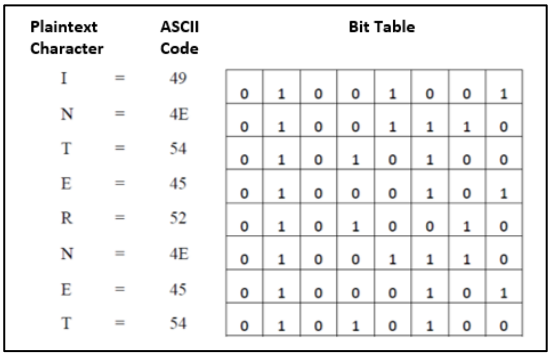


Figure 2. Simetryc cryptography simulation

In the block formation stage, the text encoding process is carried out by reading the experimental text bits, then categorized based on the code block size of N bytes [13], [14]. In one block will be divided by character into 8bits sorted vertically. For the VBR algorithm model with a size of 256 bytes, a bit table with an area of 8 columns and 256 rows is obtained [15], [16]. The encoding block character is shown in Figure 3.



Plaintext Character		ASCII Code	Bit Table							
I	=	49	0	1	0	0	1	0	0	1
N	=	4E	0	1	0	0	1	1	1	0
T	=	54	0	1	0	1	0	1	0	0
E	=	45	0	1	0	0	0	1	0	1
R	=	52	0	1	0	1	0	0	1	0
N	=	4E	0	1	0	0	1	1	1	0
E	=	45	0	1	0	0	0	1	0	1
T	=	54	0	1	0	1	0	1	0	0

Figure 3. VBR algorithm encoding block

In the symmetric algorithm process, there are two processes, namely encryption and decryption. In the encryption process there are aspects based on the technique of randomizing matrix operations in the form of bits [3].

The encryption process uses a vertical shift pattern from top to bottom in making encoding blocks in columns and rows of bits. In the same column, the number of shifts up and down is set at r bits in each of the same columns, but for each other column, a different amount of shift can be done [17]. In the bit table there are 8-bit columns, 1 byte = 8-bit with 1 character, so the simulation technique requires 8 value variables to shift the bits in each column (r1, r2, r3, … r8).

The encryption technique is carried out in 3 stages, namely selecting the plaintext to be used as simulation material, then each character in the plaintext is converted into binary numbers into an 8-bit table. Then the encryption process for bit rotation positions r1, r2, r3, r4, r5, r6, r7, r8 the simulation flow of the encryption process can be seen in Figure 4.

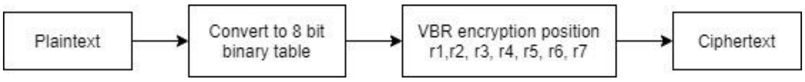

Plaintext
Convert to 8 bit binary table
VBR encryption position r1,r2, r3, r4, r5, r6, r7
Ciphertext

Figure 4. Encryption simulation

The decryption process is carried out through 3 stages, namely choosing the Ciphertext which is the simulation material, then each character in the Ciphertext is converted into binary numbers into an 8-bit table. Then the process of decrypting the bit rotation positions r1, r2, r3, r4, r5, r6, r7, r8. The simulation flow of the encryption process can be seen in Figure 5.

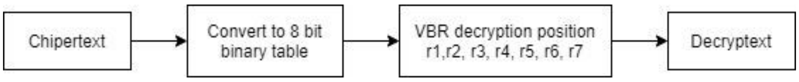

Chipertext
Convert to 8 bit binary table
VBR decryption position r1,r2, r3, r4, r5, r6, r7
Decryptext

Figure 5. Decryption simulation

3. RESULT AND DISCUSSION

The size of each binary column depends on the variable r rotation. The encryption process with VBR requires a variable as a bit column rotation that corresponds to the number of bit tables. Block encoding up to 256 numbers defined in ASCII characters. So, the rotation variable comes from the result of character values in ASCII.

The number of columns in the bit table is 8-bit columns, with each column requiring a variable rotation of the key in decimal form with a length of 8 characters.

3.1 Encryption Process

The encryption process requires a key to hide the character of each associated bit. In this simulation, the key that is used as a vertical rotation of the bit value is adjusted to the number of columns, namely 8 decimal shifters. The key in this simulation is decimal (10, 11, 7, 3, 2, 7, 5, 4).

The initial stage is the conversion of plaintext into Binary form, shown in Table 1.

Table 1. Convert plaintext to binary

Plaintext character	ASCII				Bit Table				
I	49	0	1	0	0	1	0	0	1
N	4E	0	1	0	0	1	1	1	0
T	54	0	1	0	1	0	1	0	0
E	45	0	1	0	0	0	1	0	1
R	52	0	1	0	1	0	0	1	0
N	4E	0	1	0	0	1	1	1	0
E	45	0	1	0	0	0	1	0	1
T	54	0	1	0	1	0	1	0	0

Next, the key formation process for each column of the bit table is presented in Table 2.

Table 2. Key formation stage

Plaintext character	ASCII	10	11	7	3	2	7	5	4
I	49	0	1	0	0	1	0	0	1
N	4E	0	1	0	0	1	1	1	0
T	54	0	1	0	1	0	1	0	0
E	45	0	1	0	0	0	1	0	1
R	52	0	1	0	1	0	0	1	0
N	4E	0	1	0	0	1	1	1	0
E	45	0	1	0	0	0	1	0	1
T	54	0	1	0	1	0	1	0	0

Furthermore, the process of rotating each column of bits with a decimal key in each column, the rotation of each column is carried out as many as the number of keys.

Column 1(r1) with key 10 then r1 is rotated vertically down 10 times, r2 with key 11, then rotated vertically downwards 11 times, r3 is rotated 7 times, r4 is rotated 3 times, r5 is rotated downwards 2 times, r6 is rotated vertically downwards 7 times, r7 is rotated vertically downwards 5 times, and r8 is rotated vertically downwards 4 times, then the binary result in the rotation process is defined into ASCII characters. The results of the encryption step cycle with the VBR algorithm are in Table 3.

Table 3. Binary rotation results in the encryption process

			Bit Table					ASCII
0	1	0	0	0	1	0	0	44
0	1	0	0	0	1	1	0	46
0	1	0	1	1	1	1	1	5F
0	1	0	0	1	0	0	0	48
0	1	0	0	0	1	0	1	45
0	1	0	1	0	1	0	0	54
0	1	0	0	0	1	1	0	46
0	1	0	1	1	0	0	1	59

Next, the ASCII characters in Table 3 are defined as letter characters as encryption codes or called ciphertext. The cliphertext results can be seen in Table 4.

Table 4. Ecryption result with VBR

Cliphertext	ASCII	Bit Table							
D	44	0	1	0	0	0	1	0	0
F	46	0	1	0	0	0	1	1	0
_	5F	0	1	0	1	1	1	1	1
H	48	0	1	0	0	1	0	0	0
E	45	0	1	0	0	0	1	0	1
T	54	0	1	0	1	0	1	0	0
F	46	0	1	0	0	0	1	1	0
Y	59	0	1	0	1	1	0	0	1

Based on Table 4, the encryption result of the INTERNET text is D F _ H E T F Y.

3.2 Decryption Process

The decryption stage is different from the encryption stage, namely by shifting the bits of the cliphertext vertically upwards. So every r in Table 4 is shifted or rotated vertically up as much as the key in each binary column. The results of the rotation at the decryption stage are in Table 5.

Table 5. Rotation results in the decryption process

Bit Table								ASCII
0	1	0	0	1	0	0	1	49
0	1	0	0	1	1	1	0	4E
0	1	0	1	0	1	0	0	54
0	1	0	0	0	1	0	1	45
0	1	0	1	0	0	1	0	52
0	1	0	0	1	1	1	0	4E
0	1	0	0	0	1	0	1	45
0	1	0	1	0	1	0	0	54

In Table 5, the bit values have been converted to ASCII characters, then to find the original text or plaintext, the ASCII characters are defined in character form. The results of the decryption with the VBR algorithm are in Table 6.

Table 6. Decryption result with VBR

Cliphertext	ASCII	Tabel Bit							
I	49	0	1	0	0	1	0	0	1
N	4E	0	1	0	0	1	1	1	0
T	54	0	1	0	1	0	1	0	0
E	45	0	1	0	0	0	1	0	1
R	52	0	1	0	1	0	0	1	0
N	4E	0	1	0	0	1	1	1	0
E	45	0	1	0	0	0	1	0	1
T	54	0	1	0	1	0	1	0	0

Then the result of the description is back with the character INTERNET.

4. CONCLUSION

Based on the results of this study, simulation of encryption and description by applying the Vertical Bit Rotation (VBR) algorithm of sample text characters using 10, 11, 7, 3, 2, 7, 5, and 4 bit rotation keys produces encryption codes with variations of ASCII characters. which is quite complicated. So that the VBR algorithm can be trusted as a data security algorithm and the use of the VBR hash function can guarantee data security to avoid attempts to open encryption by cryptanalysis.

REFERENCES

[1] D. A. A. Pertiwi, T. Mustaqim, and M. A. Muslim, “Prediksi Rating Aplikasi Playstore Menggunakan Xgboost,” *Proceedings of Seminar Nasional Ilmu Komputer*, Semarang: September 2020, pp. 108–112.

[2] D. P. Timothy and A. K. Santra, “A hybrid cryptography algorithm for cloud computing security,” *2017 Int. Conf. Microelectron. Devices, Circuits Syst. ICMDCS 2017*, vol. 2017-Janua, pp. 1–5, 2017, doi: 10.1109/ICMDCS.2017.8211728.

[3] C. Wang, S., Wang, C., & Xu, “An image encryption algorithm based on a hidden attractor chaos system and the Knuth–Durstenfeld algorithm,” *Opt. Lasers Eng.*, vol. 128, no. 105995, 2020.

[4] S. Agarwal, “Symmetric Key Encryption using Iterated Fractal Functions,” *Int. J. Comput. Netw. Inf. Secur.*, vol. 9, no. 4, pp. 1–9, 2017, doi: 10.5815/ijcnis.2017.04.01.

[5] F. Maqsood, M. Ahmed, M. Mumtaz, and M. Ali, “Cryptography: A Comparative Analysis for Modern Techniques,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 6, 2017, doi: 10.14569/ijacsa.2017.080659.

[6] P. R. Kumar, P. H. Raj, and P. Jelciana, “Exploring Data Security Issues and Solutions in Cloud Computing,” *Procedia Comput. Sci.*, vol. 125, no. 2009, pp. 691–697, 2018, doi: 10.1016/j.procs.2017.12.089.

[7] D. N. Tutueva, A. V., Karimov, A. I., Moysis, L., Volos, C., & Butusov, “Construction of one-way hash functions with increased key space using adaptive chaotic maps,” *Chaos, Solitons & Fractals*, vol. 141, 11034, no. 110344, 2020.

[8] D. Pradhan, S. Som, and A. Rana, “Cryptography Encryption Technique Using Circular Bit Rotation in Binary Field,” *ICRITO 2020 - IEEE 8th Int. Conf. Reliab. Infocom Technol. Optim. (Trends Futur. Dir.*, pp. 815–818, 2020, doi: 10.1109/ICRITO48877.2020.9197845.

[9] S. Patel and K. Deb, “Study of Active Earth Pressure behind a Vertical Retaining Wall Subjected to Rotation about the Base,” *Int. J. Geomech.*, vol. 20, no. 4, p. 04020028, 2020, doi: 10.1061/(asce)gm.1943-5622.0001639.

[10] E. N. Citra, “Penerapan Algoritma Vertical Bit Rotation (VBR) Dalam Penyimpanan File Online,” *MEANS (Media Inf. Anal. dan Sist).*, vol. 3, no. 1, pp. 16–23, 2018.

[11] W. Ren *et al.*, “Privacy-preserving using homomorphic encryption in Mobile IoT systems,” *Comput. Commun.*, vol. 165, pp. 105–111, 2021, doi: 10.1016/j.comcom.2020.10.022.

[12] R. Primartha, “Penerapan Enkripsi Dan Dekripsi File Menggunakan Algoritma Data Encryption Standard (DES),” *J. Res. Comput. Sci. Appl. Informatics Eng. Dep. Sriwij. Univ.*, vol. 01, no. 01, pp. 1–19, 2011.

[13] D. Arisandi and B. Yusuf, “Pemeriksaan Integritas Dokumen Dengan Digital Signature Algorithm,” *J. Inf. Syst. Informatics Eng.*, vol. 4, no. 1, pp. 1–6, 2020.

[14] Y. Zhang and D. Xiao, “An image encryption scheme based on rotation matrix bit-level permutation and block diffusion,” *Commun. Nonlinear Sci. Numer. Simul.*, vol. 19, no. 1, pp. 74–82, 2014, doi: 10.1016/j.cnsns.2013.06.031.

[15] R. N. Ibrahim, “Perangkat Lunak Keamanan Data Menggunakan Algoritma Kriptografi Simetri Tiny Ecryption Algorithm (TEA),” *J. Comput.*, vol. 13, no. 1, pp. 1–10, 2019.

[16] C. Xu, J. Sun, and C. Wang, “A novel image encryption algorithm based on bit-plane matrix rotation and hyper chaotic systems,” *Multimed. Tools Appl.*, vol. 79, no. 9–10, pp. 5573–5593, 2020, doi: 10.1007/s11042-019-08273-x.

[17] R. Bhanot and R. Hans, “A review and comparative analysis of various encryption algorithms,” *Int. J. Secur. its Appl.*, vol. 9, no. 4, pp. 289–306, 2015, doi: 10.14257/ijsia.2015.9.4.27.