Al-Jabar: Jurnal Pendidikan Matematika
Volume 14, Number 02, 2023, Pages 281 – 297
http://ejournal.radenintan.ac.id/index.php/al-jabar/index

Text security by using a combination of the vigenere cipher and the rubik's cube method of size 4×4×4

Raudhatun Safitri[1], Puguh Wahyu Prasetyo[2*], Dian Eka Wijayanti[1], Samsul Arifin[3], Fariz Setyawan[2], Joe Repka[4]

[1] Department of Mathematics, Universitas Ahmad Dahlan, Yogyakarta, Indonesia
[2] Department of Mathematics Education, Universitas Ahmad Dahlan, Yogyakarta, Indonesia
[3] Department of Statistic, School of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia
[4]Department of Mathematics, University of Toronto, Canada
✉ puguh.prasetyo@pmat.uad.ac.id*

Article Information
Submitted June 07, 2023
Revised Aug 09, 2023
Accepted Aug 12, 2023

Keywords
Technology;
Document files;
Cryptography;
Encryption and decryption;
Vigenere cipher;
Rubik’s cube.

Abstract

Background: With the rise in technology, information security is becoming increasingly crucial. Cryptography is identified as a key tool to protect data and information from growing technological threats.
Aim: The primary goal of this study is to improve text security by integrating the Vigenere cipher with the Rubik's cube algorithm. This innovative approach is aimed at strengthening the protection of textual data against unauthorized access and eavesdropping. The feasibility of this method is demonstrated through a simulation in the Python programming environment, specifically in Google Colab.
Method: The study adopts a qualitative research methodology, enhanced by an empirical simulation. It involves the application of the Vigenere Cipher and the Rubik's Cube algorithm in a 4×4×4 format for encrypting and decrypting text, with simulations performed on the Google Colab platform.
Results: The research indicates that the combination of the Vigenere Cipher and the Rubik's Cube algorithm in a 4×4×4 configuration effectively generates ciphertext. This result is substantiated by simulations in Google Colab, showcasing the method's efficiency and practicality.
Conclusion: The study presents a significant advancement in text security in the modern technological era. By merging the Vigenere Cipher with the 4×4×4 Rubik's Cube algorithm, it demonstrates a considerable improvement in the confidentiality of sensitive textual information. The practical application and effectiveness of this encryption method are validated through empirical simulations in Google Colab, highlighting its potential as a significant tool in information security.

INTRODUCTION

In the current technological era, information security is increasingly important (Al-Meer et al, 2023; Buser et al, 2023; Teranishi & Kogiso, 2023; Moosa et al, 2023; Dey & Dutta, 2023). Human activities are mostly related to data, information, and communication, and our activities are directly or indirectly related to computer technology devices (Syahib et al., 2017). Security is a basic human need; one vital aspect is the security of text. However, this has received less attention from designers and managers of information systems. The massive development of technology has led to higher levels of threat to the security of data and information dissemination (Luengo et al, 2023; Pirandola et al, 2023; Ye et al, 2023; Aissaoi et al, 2023; Schwiderowksi et al, 2023; Pocher et al, 2023; Chiu et al, 2023; Rehman et al, 2023; Huang et al, 2023; Rupa et al, 2023; Jaithunbi et al, 2022; Ahamed & Krishnamoorthy, 2020). Recent criminal wiretapping activities by other countries, and have caused serious problems, including

How to cite Safitri, R., Prasetyo, P. W., Wijayanti, D. E., Arifin, S., Setyawan, F., & Repka, J. (2023). Text security by using a combination of the vigenere cipher and the rubik's cube method of size 4×4×4. *Al-Jabar: Jurnal Pendidikan Matematika*, *14*(2), 281-297.
e-ISSN 2540-7562.
Published by Mathematics Education Department, UIN Raden Intan Lampung.

in Indonesia. If not addressed immediately, such activities will have a major impact. The sectors that have experienced the most hacks based on data from the National Cyber and Crypto Agency (BSSN) are shown in Figure 1

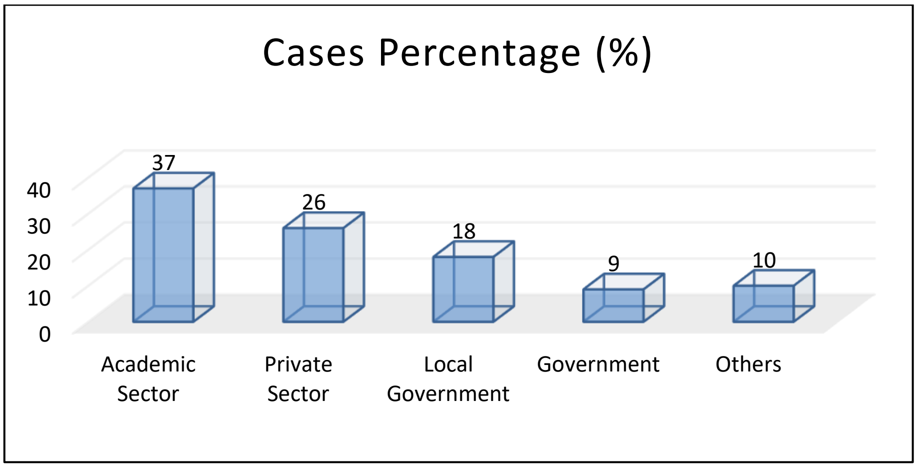


Figure 1. Column Chart of hacked sectors

The risk of data theft causes users to feel insecure if they have not taken action to secure stored documents (Tampubolon, 2021). The main target of theft is digital documents that have a high value (Aulia et al., 2019). Thus, a strong level of security for a file containing confidential data is highly expected for all users who store important data in it. Cryptography can be the right solution to protect against these crimes. Cryptography is a science that studies how to keep data or messages secure from interference by third parties; this is achieved by using a mathematical algorithm that converts data or information into a series of texts that are difficult to understand so that only the intended users can read and process it (Padhye et.al, 2018). Advances in computer science in the last 60 years have made cryptography a basic part of all aspects of contemporary life. Cryptography studies the transmission of data that is encoded in such a way that only the intended recipient can decode it (Baldoni et.al, 2008). Cryptography is proven to be secure against certain types of attacks.

The main purpose of cryptography is to keep plaintext secret from eavesdroppers. Third parties may try to modify messages that are in the data and are deemed to have full access to the communication channel. Therefore, cryptography is expected to guarantee the truth of the message (Delfs & Knebl, 2007). Cryptography converts data or messages into data that is encoded by the sender. This process is known as encryption. Encryption is the process of converting data or messages to be sent into a form that cannot be recognized by third parties. After the data or message reaches the recipient, the recipient performs decryption which is the opposite of encryption. Decryption can be interpreted as the process of changing the data or message back to its initial form so that the data or message can be conveyed to the recipient. The original data or message is called plaintext while messages that cannot be recognized by third parties are called ciphertext (Sumandri, 2017). Many cryptographic algorithms are designed to hide a message. Cryptographic algorithms are currently grouped into classical algorithms and modern algorithms (Fatonah et al., 2016). One of the cryptosystems of cryptography is the Vigenere Cipher. This cryptographic algorithm was published by a diplomat and cryptologist from France, namely Blaise de Vigenere, but in fact, this algorithm was previously described in the book La Cifra del Sig. by Giovan Batista Belaso (1553) (Minarni & Redha, 2020). The security of Vigenere Cipher depends on the number of keys used. One of the advantages of the Vigenere Cipher is the algorithm for character encryption is relatively simple but safe enough to guarantee confidentiality.

On the other hand, the cryptography of the 4 × 4 × 4 Rubik's Cube algorithm is famous for its complexity. This is because of the various configurations generated by randomization, which will increase the difficulty of guessing all keys. The basic idea of this research on symmetric cryptography design based on the 4 × 4 × 4 Rubik's Cube is complexity and an enormous number of possible guesses (Liwandouw & Wowor, 2016). Rubik's Cube was invented in 1974 by the Hungarian sculptor and professor of architecture, Erno Rubik. In 2016, Abitha and Pradeep proposed communication security based on the Rubik's Cube algorithm, in which one of the steps is the use of the Rubik's Cube principle for encrypting an image (Nana & Prasetyo, 2021).

Based on several previous studies, the authors propose an algorithm that combines the Vigenere Cipher and Rubik's Cube 4 × 4 × 4 methods. In the method, a message or text is first scrambled using the Vigenere Cipher method, and the ciphertext is obtained. The ciphertext obtained in the first process is scrambled again using a 4 × 4 × 4 Rubik's Cube. The 4 × 4 × 4 Rubik's Cube is chosen so it can accommodate 64 characters. Combining the Vigenere Cipher and the 4 × 4 × 4 Rubik's Cube to perform the encryption and decryption can improve data security because the complexity of the ciphertext is more complicated than that obtained using only one algorithm or using 3 × 3 × 3 Rubik's Cube.

METHODS

The method of this research is qualitative research with empirical simulation by using a google colab. The cryptography method is one of the methods used to improve text security because it can perform the process of encryption and decryption (Budiman & Paradise, 2019; Younus & Hussain, 2022). Someone without a decryption key will be unable to retrieve the encrypted document. The ciphertext will be decrypted using an agreed-upon key, and the original data will be returned. In the not-too-distant future, the chances of someone without a decryption key recovering the original text are extremely slim (Arifin et al., 2021).

The way the Vigenere cipher works is almost the same as the Caesar cipher. The Vigenere cipher is one of the classical cryptographic algorithms that use the compound alphabetic substitution method (Uniyal et al, 2021; Boussif et.al, 2020; Park et al, 2020; Grošek et al, 2019). Compound-alphabet substitution encrypts each letter using a different key, unlike the Caesar cipher which uses the single-alphabet substitution method, encrypting all letters using the same key (Hoerudin & Pratama, 2020; Bingöl, 2022). The Encryption and Decryption process in the Vigenere Cipher works by reading a word for a character, where if the message sent exceeds the length of the key used, the key will be repeated until the message sent gets its respective key (Amrulloh & Ujianto, 2019). Classical cryptography is a character-based cryptographic algorithm, namely encryption and decryption are used for each character of the message. The classical algorithm belongs to the symmetric cryptographic system (Permanasari, 2017), namely encryption and decryption techniques with the same technique or method or key (Irawan, 2017).

The testing process in this study was carried out with two encryption processes. The first encryption uses Vigenere Cipher then the second encryption uses the 4 × 4 × 4 Rubik's Cube algorithm. The Rubik’s Cube has 6 surfaces and has 6 different colors with dimensions of 4 × 4 × 4 (Yudanto & Suartana, 2022). Notation - movement notation on the 4 × 4 × 4 Rubik's Cube as follows:

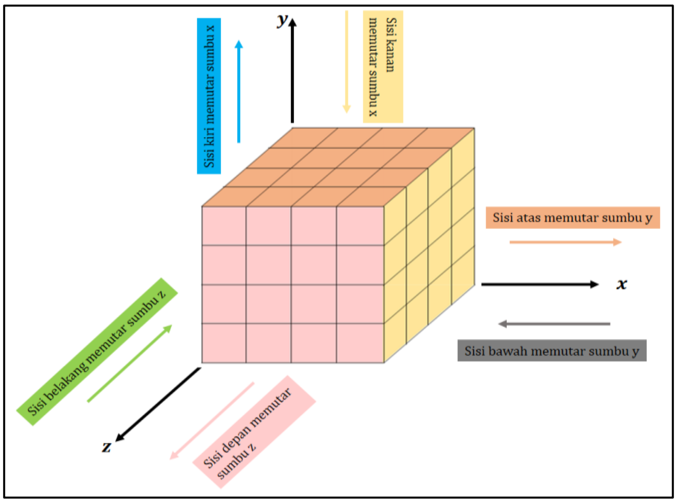

Sisi kiri memutar sumbu x
y
Sisi kanan memutar sumbu x
Sisi atas memutar sumbu y
x
Sisi bawah memutar sumbu y
Sisi belakang memutar sumbu z
z
Sisi depan memutar sumbu z

Figure 2. Cartesian coordinates of the movement of the 4 x 4 × 4 Rubik's Cube

After the decryption process using 4 × 4 × 4 Rubik's Cube method, a second decryption was carried out using the Vigenere Cipher algorithm.

Table 1. 4 × 4 × 4 Rubik's Cube Movement Notation

Side	The direction of Movement on Rubik's Cube 4 × 4 × 4
F (*Front*)	Front side rotation rotates around the z-axis by 90° clockwise
F’ (F accent)	Front side rotation rotates around the z-axis by 90°counterclockwise
F2 (F *double*)	Front side rotation rotates around the z-axis by 180°
f (*inner front*)	Inner front side rotation rotates around the z-axis by 90° clockwise
f’ (f accent)	Inner front side rotation rotates around the z-axis by 90°counterclockwise
f2 (f *double*)	Inner front side rotation rotates around the z-axis by 180°
B (*Back*)	Rear side rotation rotates around the z-axis by 90° clockwise
B’ (B accent)	Rear side rotation rotates around the z-axis by 90° counterclockwise
B2 (B *double*)	Rear side rotation rotates around the z-axis by 180°
b (*inner back*)	Inner back side rotation rotates around the z-axis by 90° clockwise
b’ (b accent)	Inner back side rotation rotates around the z-axis by 90° counterclockwise
b2 (b *double*)	Inner back side rotation rotates around the z-axis by 180°
U (*Up*)	Top side rotation rotates around the y-axis by 90° clockwise
U’ (U accent)	Top side rotation rotates around the y-axis by 90° counterclockwise
U2 (U *double*)	The rotation of the top side rotates around the y-axis by 180°
u (*inner up*)	The inner top side rotates around the y-axis by 90° clockwise
u’ (u accent)	The inner top side rotates around the y-axis by 90° counterclockwise
u2 (u *double*)	Inner top side rotation rotates around the y-axis by 180°
D (*Down*)	Bottom side rotation rotates around the y-axis by 90° clockwise
D’ (D accent)	Bottom side rotation rotates around the y-axis by 90° counterclockwise
D2 (D *double*)	Bottom side rotation rotates around the y-axis by 180°
d (*inner down*)	The inner bottom side rotates around the y-axis by 90° clockwise
d’ (d accent)	The inner bottom side rotates around the y-axis by 90° counterclockwise
d2 (d *double*)	Inner bottom side rotation rotates around the y-axis by 180°
L (*Left*)	Left side rotation rotates around the x-axis by 90° clockwise
L’ (L accent)	Left side rotation rotates around the x-axis by 90° counterclockwise
L2 (L *double*)	Left side rotation rotates around the x-axis by 180°
l (*inner left*)	Inner left side rotation rotates around the x-axis by 90° clockwise

Side	The direction of Movement on Rubik's Cube 4 × 4 × 4
l’ (l accent)	The inner left side rotation rotates around the x-axis by 90° counterclockwise
l2 (l *double*)	The inner left side rotation rotates around the x-axis by 180°
R (*Right*)	Right side rotation rotates around the x-axis by 90° clockwise
R’ (R accent)	Right side rotation rotates around the x-axis by 90° counterclockwise
R2 (R *double*)	Right side rotation rotates around the x-axis by 180°
r (*inner right*)	Inner right side rotation rotates around the x-axis by 90° clockwise
r’ (r accent)	Inner right side rotation rotates around the x-axis by 90° counterclockwise
r2 (r *double*)	The inner right side rotation rotates around the x-axis by 180°

The Vigenere Cipher encryption process is carried out in the following steps:

1. Determine the encrypted text, then divide it per block.
2. Determine the lock and key space to be used.
3. Convert letters by following ASCII table rules.
4. Text encryption with the Vigenere Cipher method.
$$C_i = e_k (P_i) = ((P_i + K_i - 64) \, mod \, 95) + 32$$
5. Ciphertext generated.

After the encryption process with the Vigenere Cipher method, the preliminary ciphertext is obtained. The text will be used as the initial text in the second encryption process, namely by using the 4 × 4 × 4 Rubik's Cube algorithm.

The 4 × 4 × 4 Rubik's Cube encryption process is carried out in the following steps:

1. Determine the position of the plaintext to be encrypted in 4 × 4 × 4 Rubik's Cube blocks.
2. Determine the plaintext per character in the specified block.
3. Fill all the remaining blocks with an “X”.
4. Determine the key to use.
5. Obtaine ciphertext.

The encryption process is carried out by combining two methods, namely Vigenere Cipher and the 4 × 4 × 4 Rubik's Cube. The first process uses the Vigenere Cipher method, and the second uses the 4 × 4 × 4 Rubik's Cube. The process is carried out sequentially and the ciphertext obtained from the first process becomes the plaintext of the second process, which is the final ciphertext. As for the decryption process, it is done the other way around.

The decryption process in Rubik's Cube algorithm can be done by the reverse of the encryption process (Ma’rifah, 2022). The decryption process using the 4 × 4 × 4 Rubik's Cube method is carried out with the following algorithm:

1. Form a Rubik based on the given key.
2. Insert ciphertext sequentially based on a predetermined order.
3. Determine the inverse of the key.
4. Returns the Rubik's Cube with the inverse of the key.
5. Enter the plaintext that matches the ciphertext into the decryption of the Vigenere Cipher.

The Vigenere Cipher decryption process is carried out in the following steps:

1. Insert the decrypted ciphertext, then divide it per block.
2. Determine the lock and key space to be used.
3. Convert letters by following ASCII table rules.
4. Text decryption with the Vigenere Cipher method.
$$P_i = d_k (C_i) = ((C_i - K_i) \, mod \, 95) + 32$$
5. Get plaintext.

RESULTS AND DISCUSSION

Encryption Process by Using Vigenere Cipher

Plaintext : Universitas Ahmad Dahlan memperoleh Akreditasi Institusi “A” tahun 2017.
Key : Matematika_2018

Plaintext and keys are substituted into numeric form based on the values in the American Standard Code for Information Interchange (ASCII) table.

Table 2. Substitution of plaintext and key into ASCII table

P_i	U	n	i	v	e	r	s	I	t	a
ASCII Value	85	110	105	118	101	114	115	105	116	97
K_i	M	a	t	e	m	a	t	I	k	a
ASCII Value	77	97	116	101	109	97	116	105	107	97
P_i	S	(space)	A	h	m	a	d	(space)	D	a
ASCII Value	115	32	65	104	109	97	100	32	68	97
K_i	_	2	0	1	8	M	a	T	e	m
ASCII Value	95	50	48	49	56	77	97	116	101	109
P_i	H	l	a	n	(space)	m	e	M	p	e
ASCII Value	104	108	97	110	32	109	101	109	112	101
K_i	A	t	i	k	a	_	2	0	1	8
ASCII Value	97	116	105	107	97	95	50	48	49	56
P_i	R	o	l	e	h	(space)	A	K	r	e
ASCII Value	114	111	108	101	104	32	65	107	114	101
K_i	M	a	t	e	m	a	t	I	k	a
ASCII Value	77	97	116	101	109	97	116	105	107	97
P_i	D	i	t	a	s	i	(space)	I	n	s
ASCII Value	100	105	116	97	115	105	32	73	110	115
K_i	_	2	0	1	8	M	a	T	e	m
ASCII Value	95	50	48	49	56	77	97	116	101	109
P_i	T	i	t	u	s	i	(space)	“	A	“
ASCII Value	116	105	116	117	115	105	32	34	65	34
K_i	A	t	i	k	a	_	2	0	1	8
ASCII Value	97	116	105	107	97	95	50	48	49	56
P_i	(space)	t	a	h	u	n	(space)	2	0	1
ASCII Value	32	116	97	104	117	110	32	50	48	49
K_i	M	a	t	e	m	a	t	I	k	a
ASCII Value	77	97	116	101	109	97	116	105	107	97
P_i	7	.								
ASCII Value	55	46								
K_i	_	2								
ASCII Value	95	50								

After the substitution, the encryption process is carried out one by one as follows:

$$C_i = e_k\,(P_i) = ((P_i + K_i - 64)\ mod\ 95) + 32$$

$$\begin{aligned}
C_1 &= ((P_1 + K_1 - 64)\ mod\ 95) + 32 \\
&= ((85 + 77 - 64)\ mod\ 95) + 32 = 35 = \# \\
C_2 &= ((P_2 + K_2 - 64)\ mod\ 95) + 32 \\
&= (110 + 97 - 64)\ mod\ 95 = 80 = P \\
C_3 &= ((P_3 + K_3 - 64)\ mod\ 95) + 32 \\
&= \big((105 + 116 - 64)\ mod\ 95\big) + 32 = 92 = \ \wedge
\end{aligned}$$

......

$$\begin{aligned}
C_{68} &= ((P_{68} + K_{68} - 64)\ mod\ 95) + 32 \\
&= ((50 + 105 - 64)\ mod\ 95) + 32 = 123 = \{
\end{aligned}$$

$$
\begin{aligned}
C_{69} &= ((P_{69} + K_{69} - 64)\ mod\ 95) + 32 \\
&= ((48 + 107 - 64) mod\ 95) + 32 = 123 = \{ \\
C_{70} &= ((P_{70} + K_{70} - 64)\ mod\ 95) + 32 \\
&= \big((49 + 97 - 64)\ mod\ 95\big) + 32 = 114 = r \\
C_{71} &= ((P_{71} + K_{71} - 64)\ mod\ 95) + 32 \\
&= \big((55 + 95 - 64)\ mod\ 95\big) + 32 = 118 = v \\
C_{72} &= ((P_{72} + K_{72} - 64)\ mod\ 95) + 32 \\
&= ((46 + 50 - 64)\ mod\ 95) + 32 = 64 = @
\end{aligned}
$$

Based on the encryption process above, the ciphertext obtained is as follows:

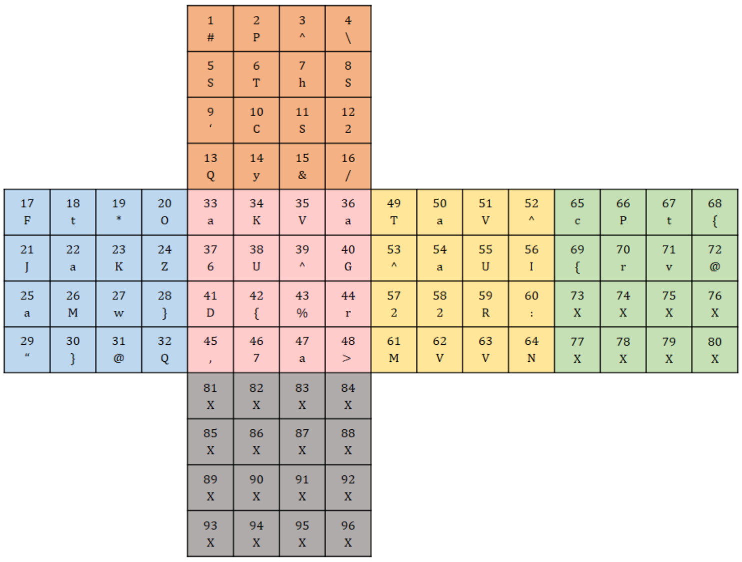
$$\#P\hat{}\backslash SThS'CS2Qy\&/Ft*OjaKZaMw\}"\}@QaKVa6U\hat{}GD\{\%r,7a$$
$$>TaV\hat{}\hat{}aUI22R:MVVNcPt\{\{rv@$$

$4 \times 4 \times 4$ *Rubik's Cube*

Plaintext :

$$\#P\hat{}\backslash SThS'CS2Qy\&/Ft*OJaKZaMw\}"\}@QaKVa6U\hat{}GD\{\%r,7a$$
$$>TaV\hat{}\hat{}aUI22R:MVVNcPt\{\{rv@$$

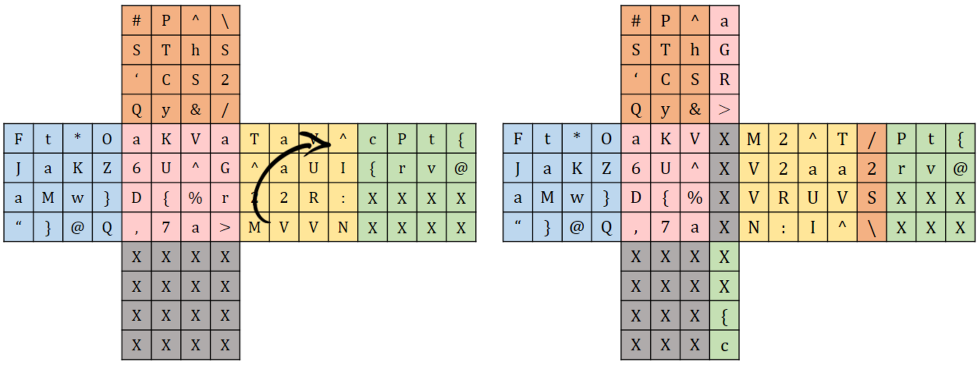
Key : $R - L' - U2 - D - F - B2 - U'$



1 # | 2 P | 3 ^ | 4 \
5 S | 6 T | 7 h | 8 S
9 ‘ | 10 C | 11 S | 12 2
13 Q | 14 y | 15 & | 16 /
17 F | 18 t | 19 * | 20 O | 33 a | 34 K | 35 V | 36 a | 49 T | 50 a | 51 V | 52 ^ | 65 c | 66 P | 67 t | 68 {
21 J | 22 a | 23 K | 24 Z | 37 6 | 38 U | 39 ^ | 40 G | 53 ^ | 54 a | 55 U | 56 I | 69 { | 70 r | 71 v | 72 @
25 a | 26 M | 27 w | 28 } | 41 D | 42 { | 43 % | 44 r | 57 2 | 58 2 | 59 R | 60 : | 73 X | 74 X | 75 X | 76 X
29 “ | 30 } | 31 @ | 32 Q | 45 , | 46 7 | 47 a | 48 > | 61 M | 62 V | 63 V | 64 N | 77 X | 78 X | 79 X | 80 X
81 X | 82 X | 83 X | 84 X
85 X | 86 X | 87 X | 88 X
89 X | 90 X | 91 X | 92 X
93 X | 94 X | 95 X | 96 X

Figure 3. Initialize encryption process with the 4 × 4 × 4 Rubik's Cube

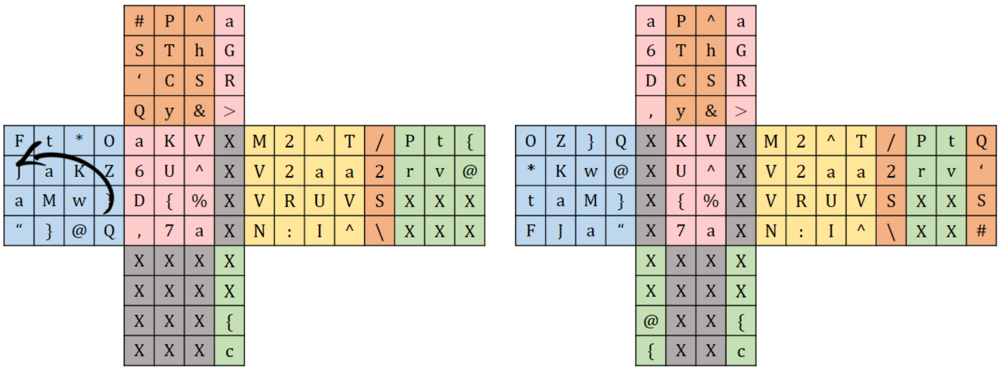
The first step, at the initialization stage of the Rubik's Cube, rotates around the x-axis by 90° clockwise on the right side (R). So the resulting Rubik's Cube block position is as follows:



# P ^ \
S T h S
‘ C S 2
Q y & /
F t * O a K V a T a V ^ c P t {
J a K Z 6 U ^ G ^ a U I { r v @
a M w } D { % r 2 2 R : X X X X
“ } @ Q , 7 a > M V V N X X X X
X X X X
X X X X
X X X X
X X X X
# P ^ a
S T h G
‘ C S R
Q y & >
F t * O a K V X M 2 ^ T / P t {
J a K Z 6 U ^ X V 2 a a 2 r v @
a M w } D { % X V R U V S X X X
“ } @ Q , 7 a X N : I ^ \ X X X
X X X X
X X X X
X X X {
X X X c

Figure 4. Initialize encryption process with the 4 × 4 × 4 Rubik's Cube first step

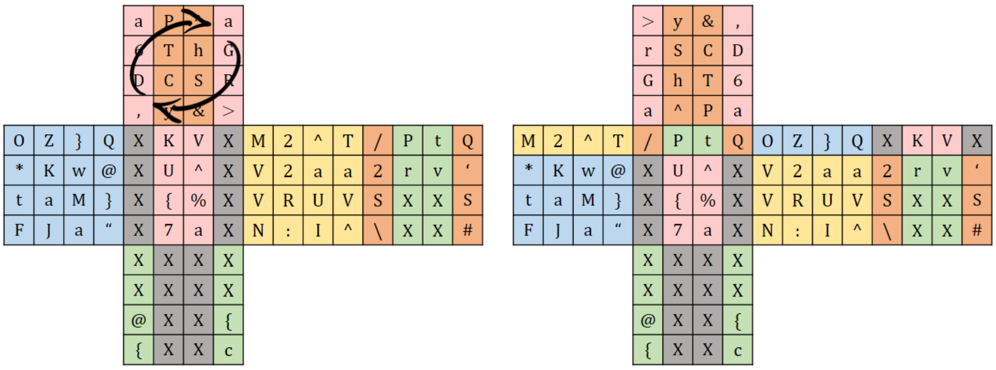
The second step results in the first stage, the Rubik's Cube is rotated on the left side rotating around the x-axis by 90° counterclockwise (L'). So that the following results are obtained:



# P ^ a
S T h G
' C S R
Q y & >
F t * O a K V X M 2 ^ T / P t {
J a K Z 6 U ^ X V 2 a a 2 r v @
a M w } D { % X V R U V S X X X
" } @ Q , 7 a X N : I ^ \ X X X
X X X X
X X X X
X X X {
X X X c
a P ^ a
6 T h G
D C S R
, y & >
O Z } Q X K V X M 2 ^ T / P t Q
* K w @ X U ^ X V 2 a a 2 r v '
t a M } X { % X V R U V S X X S
F J a " X 7 a X N : I ^ \ X X #
X X X X
X X X X
@ X X {
{ X X c

Figure 5. Initialize encryption process with the 4 × 4 × 4 Rubik's Cube second step

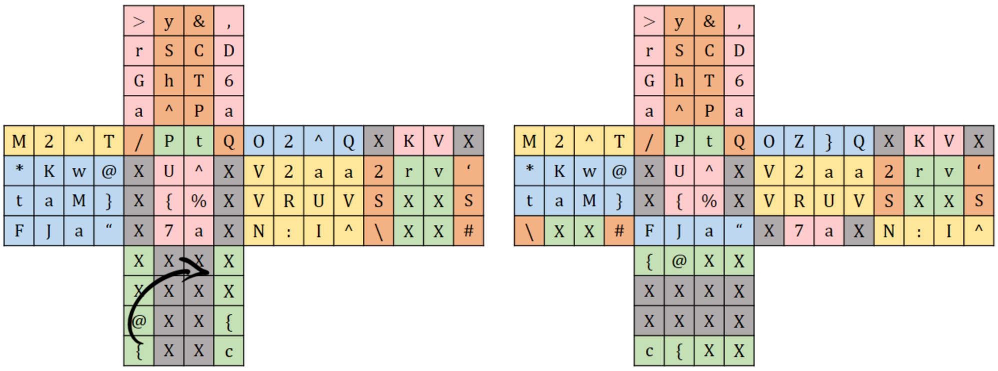
In the third step, the Rubik's Cube generated in the second stage of rotation of the top side rotates around the y-axis by 180° (U2). The following results were obtained:



a P ^ a
6 T h G
D C S R
, y & >
O Z } Q X K V X M 2 ^ T / P t Q
* K w @ X U ^ X V 2 a a 2 r v '
t a M } X { % X V R U V S X X S
F J a " X 7 a X N : I ^ \ X X #
X X X X
X X X X
@ X X {
{ X X c
> y & ,
r S C D
G h T 6
a ^ P a
M 2 ^ T / P t Q O Z } Q X K V X
* K w @ X U ^ X V 2 a a 2 r v '
t a M } X { % X V R U V S X X S
F J a " X 7 a X N : I ^ \ X X #
X X X X
X X X X
@ X X {
{ X X c

Figure 6. Initialize encryption process with the 4 × 4 × 4 Rubik's Cube third step

In the fourth step, the Rubik's Cube generated in the third stage of rotation of the bottom side rotates around the y-axis by 90° clockwise(D). The following results were obtained:



> y & ,
r S C D
G h T 6
a ^ P a
M 2 ^ T / P t Q O 2 ^ Q X K V X
* K w @ X U ^ X V 2 a a 2 r v '
t a M } X { % X V R U V S X X S
F J a " X 7 a X N : I ^ \ X X #
X X X X
X X X X
@ X X {
{ X X c
> y & ,
r S C D
G h T 6
a ^ P a
M 2 ^ T / P t Q O Z } Q X K V X
* K w @ X U ^ X V 2 a a 2 r v '
t a M } X { % X V R U V S X X S
\ X X # F J a " X 7 a X N : I ^
{ @ X X
X X X X
X X X X
c { X X

Figure 7. Initialize encryption process with the 4 × 4 × 4 Rubik's Cube fourth step

In the fifth step, the Rubik's Cube generated in the fourth stage of rotation of the front side rotates around the z-axis by 90° clockwise (F). The following results were obtained:

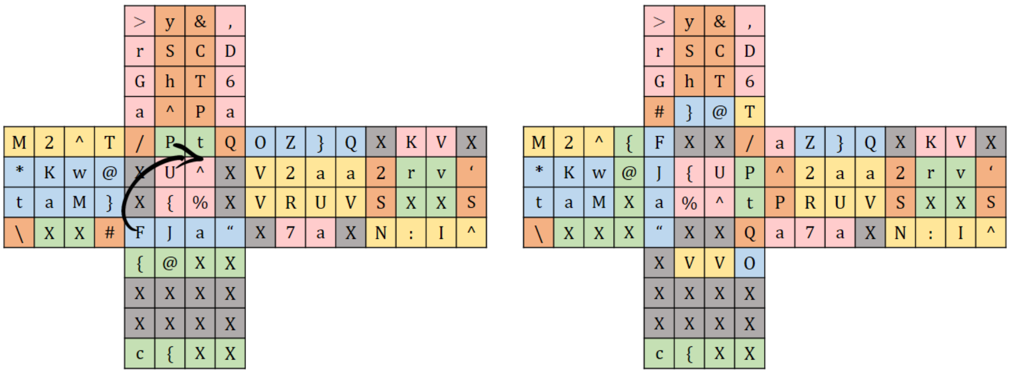


Figure 8. Initialize the encryption process with the 4 × 4 × 4 Rubik's Cube fifth step

In the sixth step, the back side rotation rotates around the z-axis by 180° (B2). The following results were obtained:

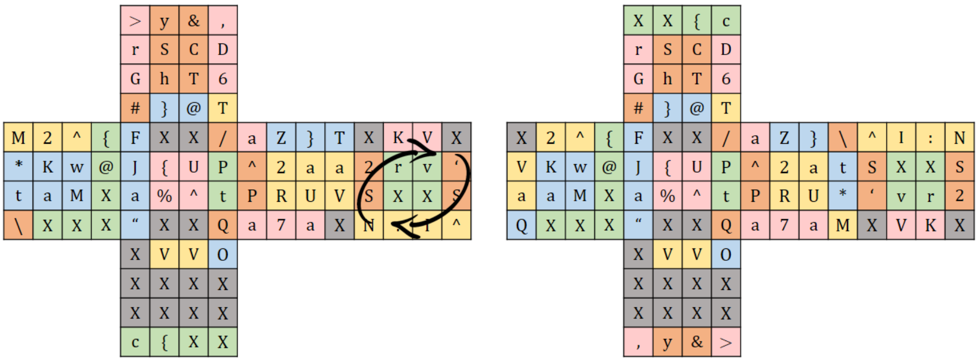


Figure 9. Initialize encryption process with the 4 x 4 × 4 Rubik's Cube sixth step

In the last step, the Rubik's Cube generated in the sixth stage is rotated around the z-axis by 180° on the back side (U'). So that the ciphertext results are obtained as follows:

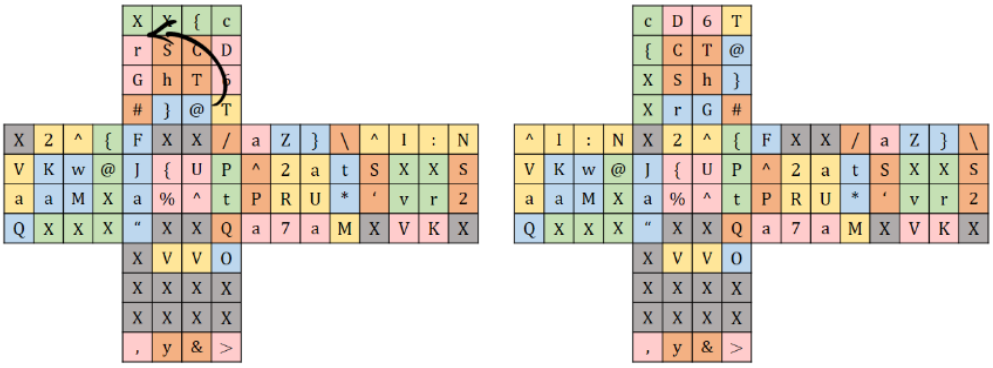


Figure 10. Initialize encryption process with the 4 × 4 × 4 Rubik's Cube seventh step

Based on the encryption process above, the ciphertext obtained is as follows:

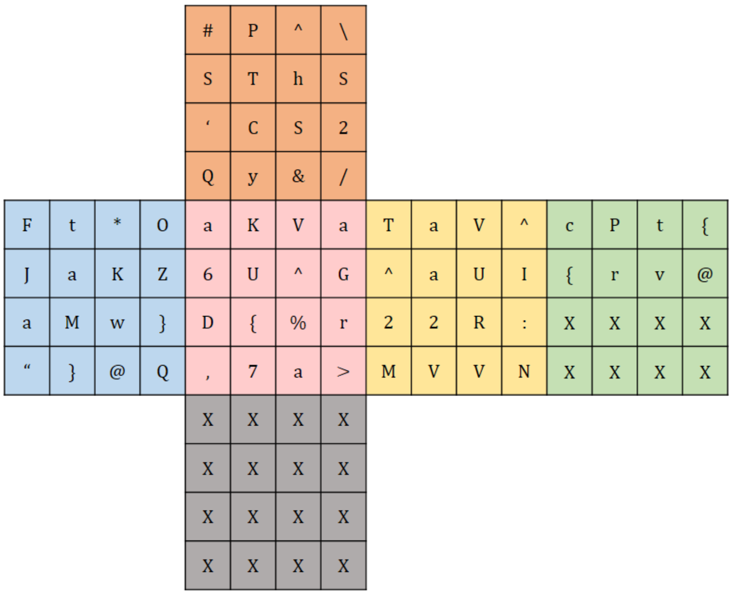
$$cD6T\{CT@XSh\}XrG\#^{\prime}:NVKw@aaMXQXXXX2^\wedge\{J\{UPa\%^\wedge t"XXQFXX/^\wedge 2atPRU$$
$$*\,a7aMaZ\}\backslash SXXS'vr2XVKXXVVOXXXXXXXX,y\&>$$

Decryption Process For the 4 × 4 × 4 Rubik's Cube

Ciphertext :

$$cD6T\{CT@XSh\}XrG\#^{\prime}:NVKw@aaMXQXXXX2^\wedge\{J\{UPa\%^\wedge t"XXQFXX/^\wedge 2atPRU$$
$$*\,a7aMaZ\}\backslash SXXS'vr2XVKXXVVOXXXXXXXX,y\&>$$

Key : $R - L' - U2 - D - F - B2 - U'$
Key inverse : $U - B2 - F' - D' - U2 - L - R'$



				#	P	^	\								
				S	T	h	S								
				‘	C	S	2								
				Q	y	&	/								
F	t	*	O	a	K	V	a	T	a	V	^	c	P	t	{
J	a	K	Z	6	U	^	G	^	a	U	I	{	r	v	@
a	M	w	}	D	{	%	r	2	2	R	:	X	X	X	X
“	}	@	Q	,	7	a	>	M	V	V	N	X	X	X	X
				X	X	X	X								
				X	X	X	X								
				X	X	X	X								
				X	X	X	X								

Figure 11. Decryption result of the 4 × 4 × 4 Rubik's Cube

Based on the the 4 × 4 × 4 Rubik's Cube decryption process, the following plaintext is obtained:

$$\begin{gathered}\#P^\wedge\backslash SThS'CS2Qy\&/Ft*OjaKZaMw\}''\}@QaKVa6U^\wedge GD\{\%r,7a\\ > TaV^\wedge{}^\wedge aUI22R:MVVNcPt\{\{rv@\end{gathered}$$

Vigenere Cipher

The Vigenere Cipher decryption process is almost the same as the encryption process, using the Matematika_2018 key.
Ciphertext and keys are substituted into numbers based on the values in the American Standard Code for Information Interchange (ASCII) table which can be seen in Table 3 below.
Ciphertext :

$$\begin{gathered}\#P^\wedge\backslash SThS'CS2Qy\&/Ft*OjaKZaMw\}''\}@QaKVa6U^\wedge GD\{\%r,7a\\ > TaV^\wedge{}^\wedge aUI22R:MVVNcPt\{\{rv@\end{gathered}$$

Key : Matematika_2018

Table 3. ASCII Symbols and Character Convertion

C_i	#	P	^	\	S	T	h	S	‘	C
Nilai ASCII	35	80	94	92	83	84	104	83	96	67
K_i	M	a	T	e	m	A	t	i	k	a
Nilai ASCII	77	97	116	101	109	97	116	105	107	97
C_i	S	2	Q	Y	&	/	F	t	*	O
Nilai ASCII	83	50	81	121	38	47	70	116	42	79
K_i	_	2	0	1	8	M	a	t	e	m
Nilai ASCII	95	50	48	49	56	77	97	116	101	109
C_i	J	A	K	Z	a	M	w	}	“	}
Nilai ASCII	74	97	75	90	97	77	119	125	34	125
K_i	a	t	I	k	a	_	2	0	1	8
Nilai ASCII	97	116	105	107	97	95	50	48	49	56
C_i	@	Q	A	K	V	a	6	U	^	G
Nilai ASCII	64	81	97	75	86	97	54	85	94	71
K_i	M	A	T	e	m	a	t	i	k	a

Nilai ASCII	77	97	116	101	109	97	116	105	107	97
$\boldsymbol{C_i}$	D	{	%	R	,	7	a	>	T	a
Nilai ASCII	68	123	37	114	44	55	97	62	84	97
$\boldsymbol{K_i}$	_	2	0	1	8	M	a	t	e	m
Nilai ASCII	95	50	48	49	56	77	97	116	101	109
$\boldsymbol{C_i}$	V	^	^	a	U	I	2	2	R	:
Nilai ASCII	86	94	94	97	85	73	50	50	82	58
$\boldsymbol{K_i}$	a	t	I	k	a	_	2	0	1	8
Nilai ASCII	97	116	105	107	97	95	50	48	49	56
$\boldsymbol{C_i}$	M	V	V	N	c	P	t	{	{	r
Nilai ASCII	77	86	86	78	99	80	116	123	123	114
$\boldsymbol{K_i}$	M	a	T	e	m	a	t	i	k	a
Nilai ASCII	77	97	116	101	109	97	116	105	107	97
$\boldsymbol{C_i}$	v	@								
Nilai ASCII	118	64								
$\boldsymbol{K_i}$	_	2								
Nilai ASCII	95	50								

After being substituted, the following decryption process is carried out one by one:

$$P_i = d_k\ (C_i\ ) = (\ (C_i - K_i\ )\ mod\ 95\ )\ + 32$$

$$\begin{aligned}
P_1 &= ((\ C_1 - K_1\ )\ mod\ 95\ )\ + 32 \\
&= \big((\ 35 - 77\ )\ mod\ 95\ \big) + 32 = 85 = U \\
P_2 &= ((\ C_2 - K_2\ )\ mod\ 95\ )\ + 32 \\
&= \big((\ 80 - 97\ )\ mod\ 95\ \big) + 32 = 110 = n \\
P_3 &= ((\ C_3 - K_3\ )\ mod\ 95\ )\ + 32 \\
&= \big((\ 94 - 116\ )\ mod\ 95\ \big) + 32 = 105 = i \\
P_4 &= ((\ C_4 - K_4\ )\ mod\ 95\ )\ + 32 \\
&= \big((\ 92 - 101\ )\ mod\ 95\ \big) + 32 = 118 = v \\
P_5 &= ((\ C_5 - K_5\ )\ mod\ 95\ )\ + 32 \\
&= (\ (\ 83 - 109\ )\ mod\ 95\ ) + 32 = 101 = e \\
\dots \\
P_{68} &= ((\ C_{68} - K_{68}\ )\ mod\ 95\ )\ + 32 \\
&= ((\ 123 - 105\ )\ mod\ 95\ )\ + 32 = 50 = 2 \\
P_{69} &= ((\ C_{69} - K_{69}\ )\ mod\ 95\ )\ + 32 \\
&= ((\ 123 - 107\ )\ mod\ 95\ )\ + 32 = 48 = 0 \\
P_{70} &= ((\ C_{70} - K_{70}\ )\ mod\ 95\ )\ + 32 \\
&= ((\ 114 - 97\ )\ mod\ 95\ )\ + 32 = 49 = 1 \\
P_{71} &= ((\ C_{71} - K_{71}\ )\ mod\ 95\ )\ + 32 \\
&= ((\ 118 - 95\ )\ mod\ 95\ )\ + 32 = 55 = 7 \\
P_{72} &= ((\ C_{72} - K_{72}\ )\ mod\ 95\ )\ + 32 \\
&= \big((\ 64 - 50\ )mod\ 95\ \big) + 32 = 46 = .
\end{aligned}$$

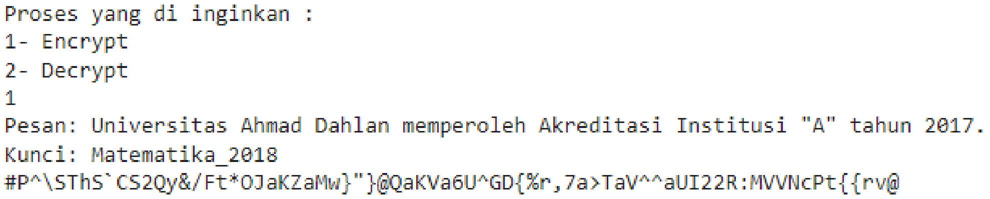
Based on the Vigenere Cipher decryption process, the plaintext is recovered as follows: Universitas Ahmad Dahlan memperoleh Akreditasi Institusi “A” pada tahun 2017.

Encryption Simulation Process Using Python Base Language, Google colab

Python is a simple programming language. Unlike other languages that are difficult to read and understand, Python places more emphasis on code readability to make it easier to understand syntax. This makes Python very easy to learn both for beginners and for those who have mastered other programming languages. Python is a high-level programming language (High-

Level Language). Python is a programming language that is ranked as the 5th most used programming language in the world (Trisno, 2016).

The encryption process in Python is divided into two stages, namely encoding with the Vigenere Cipher and the 4 x 4×4 Rubik's Cube methods. In the process of encoding the Vigenere Cipher method in Python, the following display will appear.

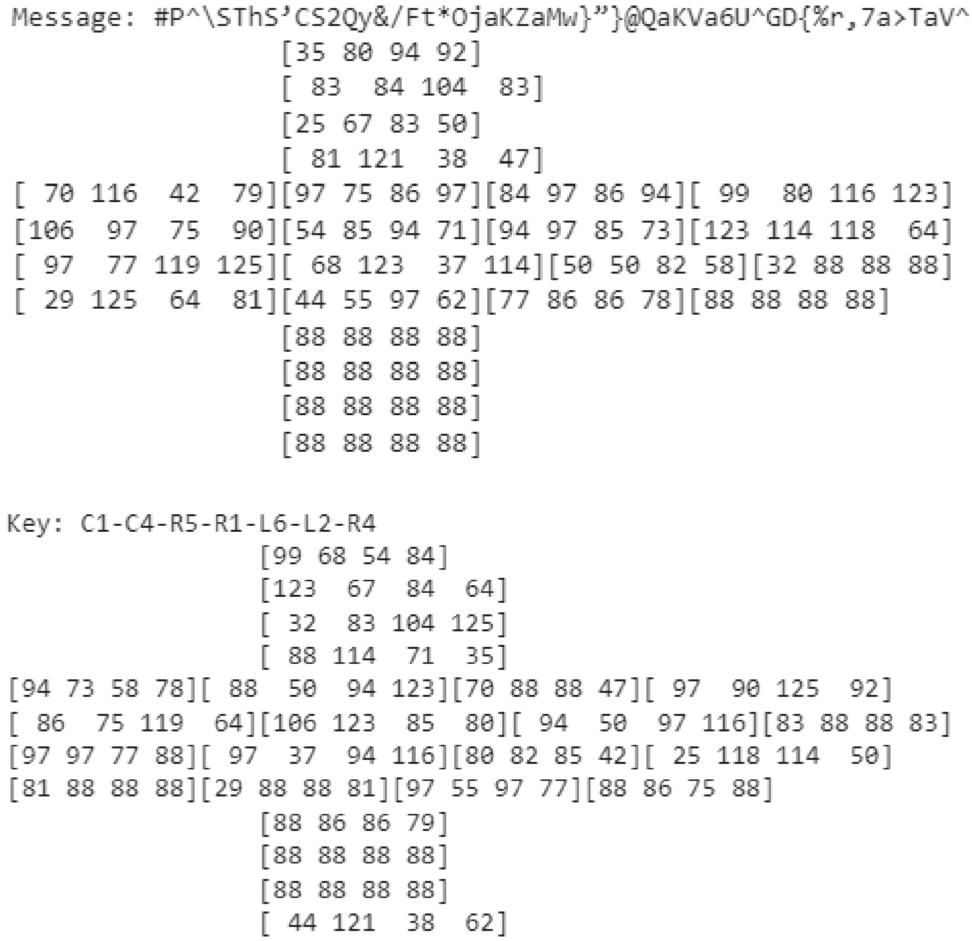


```
Proses yang di inginkan :
1- Encrypt
2- Decrypt
1
Pesan: Universitas Ahmad Dahlan memperoleh Akreditasi Institusi "A" tahun 2017.
Kunci: Matematika_2018
#P^\SThS`CS2Qy&/Ft*OJaKZaMw}"}@QaKVa6U^GD{%r,7a>TaV^^aUI22R:MVVNcPt{{rv@
```

Figure 12. Vigenere Cipher encryption process in Python

Figure 12 will encrypt the sentence that Ahmad Dahlan University obtained Institutional Accreditation "A" in 2017. The encoding process begins by entering the desired process, 1 for the encryption process and 2 for the decryption process. Then enter the desired message and key, then the ciphertext message appears: $\#P\hat{}\backslash SThS'CS2Qy\&/Ft * OjaKZaMw\}''\}@QaKVa6U\hat{}GD\{\%r,7a > TaV\hat{}\hat{}aUI22R\colon MVVNcPt\{\{rv@$.

After the encryption process is carried out using the Vigenere Cipher method, it will then be encrypted with a $4 \times 4 \times 4$ Rubik's Cube. The key to be used is “$R - L' - U2 - D - F - B2 - U'$ ”. So the display will appear as shown below:

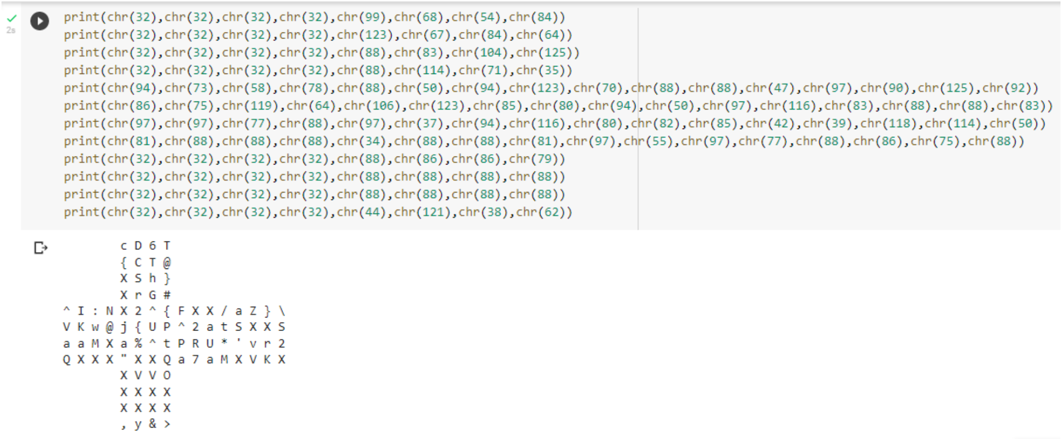


```
Message: #P^\SThS’CS2Qy&/Ft*OjaKZaMw}”}@QaKVa6U^GD{%r,7a>TaV^
                 [35 80 94 92]
                 [ 83  84 104  83]
                 [25 67 83 50]
                 [ 81 121  38  47]
[ 70 116  42  79][97 75 86 97][84 97 86 94][ 99  80 116 123]
[106  97  75  90][54 85 94 71][94 97 85 73][123 114 118  64]
[ 97  77 119 125][ 68 123  37 114][50 50 82 58][32 88 88 88]
[ 29 125  64  81][44 55 97 62][77 86 86 78][88 88 88 88]
                 [88 88 88 88]
                 [88 88 88 88]
                 [88 88 88 88]
                 [88 88 88 88]


Key: C1-C4-R5-R1-L6-L2-R4
                 [99 68 54 84]
                 [123  67  84  64]
                 [ 32  83 104 125]
                 [ 88 114  71  35]
[94 73 58 78][ 88  50  94 123][70 88 88 47][ 97  90 125  92]
[ 86  75 119  64][106 123  85  80][ 94  50  97 116][83 88 88 83]
[97 97 77 88][ 97  37  94 116][80 82 85 42][ 25 118 114  50]
[81 88 88 88][29 88 88 81][97 55 97 77][88 86 75 88]
                 [88 86 86 79]
                 [88 88 88 88]
                 [88 88 88 88]
                 [ 44 121  38  62]
```

Figure 13. 4 × 4 × 4 Rubik's Cube encryption process in Python

Based on the picture above, the encryption results are obtained as follows:
99 68 54 84 123 67 84 64 32 83 104 125 88 114 71 35 94 73 58 78 86 75 119 64 97 97 77 88 81 88 88 88 88 50 94 123 106 123 85 80 97 37 94 116 29 88 88 81 70 88 88 47 94 50 97 116 88 82 85 42 97 55 97 77 97 90 125 92 83 88 88 83 25 118 114 50 88 86 75 88 88 86 86 79 88 88 88 88 88 88 88 88 44 121 38 62
Then the results of the encryption are converted to ASCII characters as shown below:



```
print(chr(32),chr(32),chr(32),chr(32),chr(99),chr(68),chr(54),chr(84))
print(chr(32),chr(32),chr(32),chr(32),chr(123),chr(67),chr(84),chr(64))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(83),chr(104),chr(125))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(114),chr(71),chr(35))
print(chr(94),chr(73),chr(58),chr(78),chr(88),chr(50),chr(94),chr(123),chr(70),chr(88),chr(88),chr(47),chr(97),chr(90),chr(125),chr(92))
print(chr(86),chr(75),chr(119),chr(64),chr(106),chr(123),chr(85),chr(80),chr(94),chr(50),chr(97),chr(116),chr(83),chr(88),chr(88),chr(83))
print(chr(97),chr(97),chr(77),chr(88),chr(97),chr(37),chr(94),chr(116),chr(80),chr(82),chr(85),chr(42),chr(39),chr(118),chr(114),chr(50))
print(chr(81),chr(88),chr(88),chr(88),chr(34),chr(88),chr(88),chr(81),chr(97),chr(55),chr(97),chr(77),chr(88),chr(86),chr(75),chr(88))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(86),chr(86),chr(79))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(88),chr(88),chr(88))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(88),chr(88),chr(88))
print(chr(32),chr(32),chr(32),chr(32),chr(44),chr(121),chr(38),chr(62))
```

```
        c D 6 T
        { C T @
        X S h }
        X r G #
^ I : N X 2 ^ { F X X / a Z } \
V K w @ j { U P ^ 2 a t S X X S
a a M X a % ^ t P R U * ' v r 2
Q X X X " X X Q a 7 a M X V K X
        X V V O
        X X X X
        X X X X
        , y & >
```

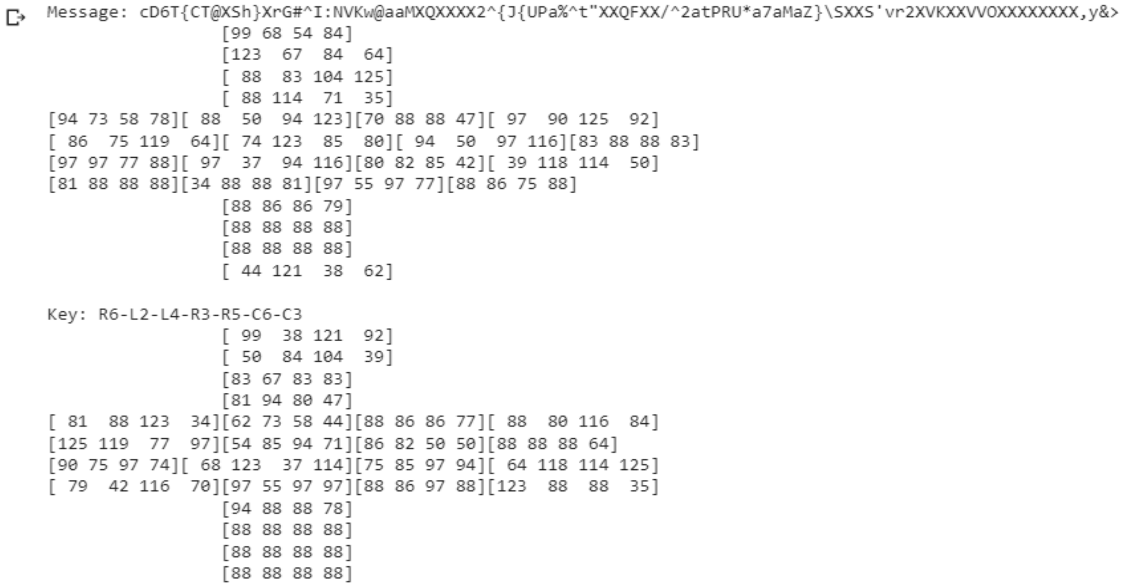
Figure 14. Conversion to ASCII characters

So the encryption results are obtained as follows:

$$cD6T\{CT@XSh\}XrG\#\hat{}I{:}NVKw@aaMXQXXXX2\hat{}\{J\{UPa\%\hat{}t"XXQFXX/\hat{}2atPRU$$
$$* \, a7aMaZ\}\backslash SXXS'vr2XVKXXVVOXXXXXXXX,y\& >$$

Decryption Process

In the decryption process, the first thing to do is to decrypt it using a 4 × 4 × 4 Rubik's Cube. The plaintext of the second encryption is:
$cD6T\{CT@XSh\}XrG\#\hat{}I{:}NVKw@aaMXQXXXX2\hat{}\{J\{UPa\%\hat{}t"XXQFXX/\hat{}2atPRU *$
$a7aMaZ\}\backslash SXXS'vr2XVKXXVVOXXXXXXXX,y\& >$.
Then the key used is the reverse of the key in the previous encryption process The following key is obtained "$U - B2 - F\hat{}' - D - U2 - L - R'$". The process can be seen in the following image:

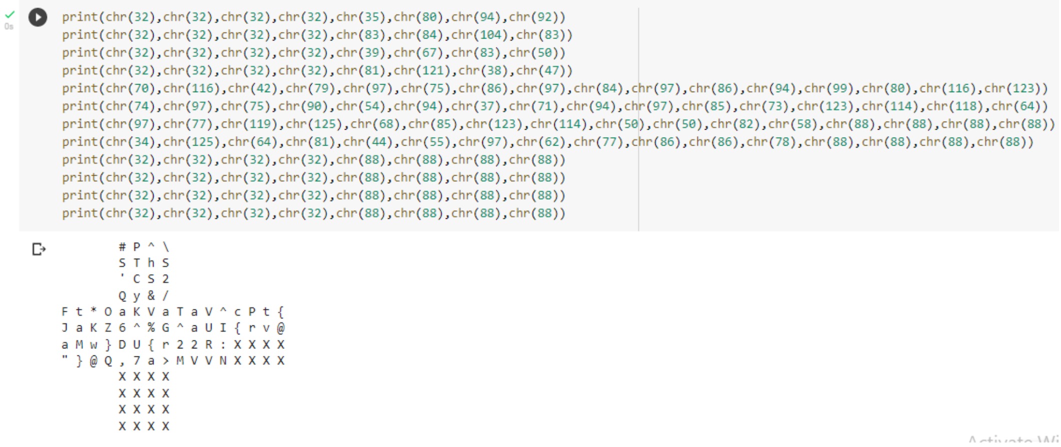


```
Message: cD6T{CT@XSh}XrG#^I:NVKw@aaMXQXXXX2^{J{UPa%^t"XXQFXX/^2atPRU*a7aMaZ}\SXXS'vr2XVKXXVVOXXXXXXXX,y&>
                 [99 68 54 84]
                 [123  67  84  64]
                 [ 88  83 104 125]
                 [ 88 114  71  35]
[94 73 58 78][ 88  50  94 123][70 88 88 47][ 97  90 125  92]
[ 86  75 119  64][ 74 123  85  80][ 94  50  97 116][83 88 88 83]
[97 97 77 88][ 97  37  94 116][80 82 85 42][ 39 118 114  50]
[81 88 88 88][34 88 88 81][97 55 97 77][88 86 75 88]
                 [88 86 86 79]
                 [88 88 88 88]
                 [88 88 88 88]
                 [ 44 121  38  62]

Key: R6-L2-L4-R3-R5-C6-C3
                 [ 99  38 121  92]
                 [ 50  84 104  39]
                 [83 67 83 83]
                 [81 94 80 47]
[ 81  88 123  34][62 73 58 44][88 86 86 77][ 88  80 116  84]
[125 119  77  97][54 85 94 71][86 82 50 50][88 88 88 64]
[90 75 97 74][ 68 123  37 114][75 85 97 94][ 64 118 114 125]
[ 79  42 116  70][97 55 97 97][88 86 97 88][123  88  88  35]
                 [94 88 88 78]
                 [88 88 88 88]
                 [88 88 88 88]
                 [88 88 88 88]
```

Figure 15. 4 × 4 × 4 Rubik's Cube decryption process in Python

Based on Figure 15, the results of the decription are as follows:
99 38 121 92 50 84 104 39 83 67 83 83 81 94 80 47 81 88 123 34 125 119 77 97 90 75 97 74 79 42 116 70 62 73 58 44 54 85 94 71 123 37 114 97 55 97 97 88 86 86 77 86 82 50 50 75 85 97 94 88 86 97 88 88 80 116 84 88 88 88 64 64 118 114 125 123 88 88 35 94 88 88 78 88 88 88 88 88 88 88 88 88 88 88 88
Then the results of the decryption are converted to ASCII characters as shown below:

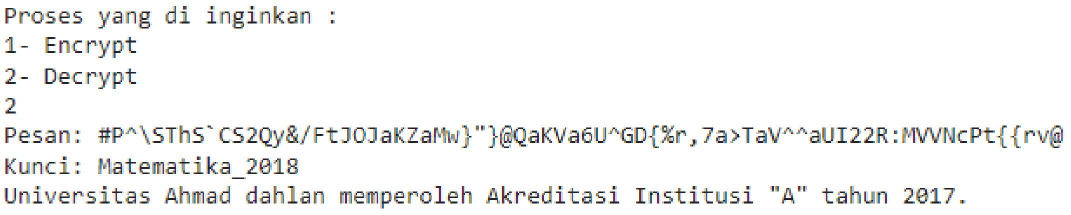


```
print(chr(32),chr(32),chr(32),chr(32),chr(35),chr(80),chr(94),chr(92))
print(chr(32),chr(32),chr(32),chr(32),chr(83),chr(84),chr(104),chr(83))
print(chr(32),chr(32),chr(32),chr(32),chr(39),chr(67),chr(83),chr(50))
print(chr(32),chr(32),chr(32),chr(32),chr(81),chr(121),chr(38),chr(47))
print(chr(70),chr(116),chr(42),chr(79),chr(97),chr(75),chr(86),chr(97),chr(84),chr(97),chr(86),chr(94),chr(99),chr(80),chr(116),chr(123))
print(chr(74),chr(97),chr(75),chr(90),chr(54),chr(94),chr(37),chr(71),chr(94),chr(97),chr(85),chr(73),chr(123),chr(114),chr(118),chr(64))
print(chr(97),chr(77),chr(119),chr(125),chr(68),chr(85),chr(123),chr(114),chr(50),chr(50),chr(82),chr(58),chr(88),chr(88),chr(88),chr(88))
print(chr(34),chr(125),chr(64),chr(81),chr(44),chr(55),chr(97),chr(62),chr(77),chr(86),chr(86),chr(78),chr(88),chr(88),chr(88),chr(88))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(88),chr(88),chr(88))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(88),chr(88),chr(88))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(88),chr(88),chr(88))
print(chr(32),chr(32),chr(32),chr(32),chr(88),chr(88),chr(88),chr(88))
```

```
        # P ^ \
        S T h S
        ' C S 2
        Q y & /
F t * O a K V a T a V ^ c P t {
J a K Z 6 ^ % G ^ a U I { r v @
a M w } D U { r 2 2 R : X X X X
" } @ Q , 7 a > M V V N X X X X
        X X X X
        X X X X
        X X X X
        X X X X
```

Figure 16. Conversion to ASCII characters

After the decryption process is carried out using the 4 × 4 × 4, Rubik's Cube, then the Vigenere Cipher method is carried out as shown in the following image:



```
Proses yang di inginkan :
1- Encrypt
2- Decrypt
2
Pesan: #P^\SThS`CS2Qy&/FtJOJaKZaMw}"}@QaKVa6U^GD{%r,7a>TaV^^aUI22R:MVVNcPt{{rv@
Kunci: Matematika_2018
Universitas Ahmad dahlan memperoleh Akreditasi Institusi "A" tahun 2017.
```

Figure 17. Vigenere Cipher decryption process in Python

Based on Figure 17 we get the sentence

Universitas Ahmad Dahlan memperoleh Akreditasi Institusi "A" tahun 2017.

Limitation and Suggestion for Further Research

Further research is expected to improve existing deficiencies and is expected to further develop what has been done in this research. For this reason, further research is recommended to develop a document file security system for all types of documents with a security system using another cryptographic algorithm or the latest which is better than the Vigenere Cipher and 4 × 4 × 4 Rubik's Cube cryptography, as well as the development of a Python program by integrating the results of the Vigenere Cipher with the 4 × 4 × 4 Rubik's Cube.

CONCLUSIONS

Based on the discussion in this study, it can be concluded that the security in the encryption process lies in the length of the Vigenere Cipher key character and the notation of the 4 × 4 × 4 Rubik's Cube movement. The encryption process using the Vigenere Cipher and 4 × 4 × 4 Rubik's Cube method produces a more random and complicated ciphertext. The use of two types of ciphers allows the security of messages to be doubled.

The encryption technique used using Vigenere Cipher with ASCII substitution modification is performed by using the equation $C_i = e_k\,(P_i) = ((C_i - K_i - 64)\, mod\, 95) + 32$ encoding results then re-encoded using Rubik's Cube $4 \times 4 \times 4$. The message process uses $4 \times 4 \times 4$ Rubik's Cube decryption followed by Vigenere Cipher decryption using the equation $P_i = d_k\,(C_i) = ((C_i - K_i)\, mod\, 95) + 32$.

In this study, the encoding process manually takes a very long time. But with the help of Python, we can minimize the steps and errors made especially in the encoding process with the $4 \times 4 \times 4$ Rubik's Cube which uses a lot of rotation on the key.

ACKNOWLEDGMENT

Authors thank to all parties who support this research. This research is supported by the Institute of Research and Community Service of Universitas Ahmad Dahlan under number grant: PD-158/SP3/LPPM-UAD/VIII/2023 and the manuscript has been completed when the corresponding author visits the University of Toronto in Fall 2023.

AUTHOR CONTRIBUTIONS STATEMENT

The author's contribution is to collect an algorithm in a cryptographic system which combines a text message security system through the Vigenere Cipher and the $4 \times 4 \times 4$ Rubik's Cube and implements it through the Python online base language software, google colab.

REFERENCES

Ahamed, B. B., & Krishnamoorthy, M. (2020). SMS Encryption and Decryption Using Modified Vigenere Cipher Algorithm, *Journal of the Operations Research Society of China, 10*, 835-848. https://doi.org/10.1007/s40305-020-00320-x

Al-Meer, A., & Al-Kuwari, S. (2023). Physical unclonable functions (PUF) for IoT devices. *ACM Computing Surveys*, *55*(14s), 1-31. https://doi.org/10.1145/3591464

Amrulloh, A., & Ujianto, E. I. H. (2019). Kriptografi Simetris Menggunakan Algoritma Vigenere Cipher. *CoreIT*, ***5***(2), 71–77.

Arifin, S., Muktyas, I. B., & Prasetyo, P. W. (2021). Unimodular matrix and bernoulli map on text encryption algorithm using python. *Al-Jabar: Jurnal Pendidikan Matematika, 12*(2), 447–455. http://dx.doi.org/10.24042/ajpm.v12i2.10469

Aulia, R., Zakir, A., & Zulhafiz, M. (2019). Penerapan Algoritma One Time Pad & Linear Congruential Generator Untuk Keamanan Pesan Teks. *Jurnal Nasional Informatika Dan Ilmu Komputer*, 4(*1)*, 37-41. http://dx.doi.org/10.30743/infotekjar.v4i1.1590

Baldoni, M. W., Ciliberto, C., & Cattaneo G. M. P (2008). *Elementary Number Theory, Cryptography and Codes*. Springer. https://doi.org/10.1007/978-3-540-69200-3

Bingöl, S. (2022). The changes in Ottoman diplomatic cryptography and its methods during the 19th century (1811–1877), *Cryptologia 47*(6), 541-569. https://doi.org/10.1080/01611194.2022.2092916

Boussif, M., Aloui, N., & Cherif, A. (2020). Securing DICOM images by a new encryption algorithm using Arnold transform and Vigenère cipher, *IET Image Processing, 14*(6), 1209-1216. https://doi.org/10.1049/iet-ipr.2019.0042

Budiman, A., & Paradise. (2019). Modification of Vigenere Algorithm and One Time Pad Using Rivest Code 6 (RC6) Key Expansion. *Compiler, 8(2),* 149-155. http://dx.doi.org/10.28989/compiler.v8i2.481

Buser, M., Dowsley, R., Esgin, M., Gritti, C., Kermanshahi, S. K., Kuchta, V., Legrov, J., Liu, J., Phan, R., Sakzad, A., Steinfeld, R., & Yu, J. (2023). A Survey on Exotic Signatures for

Post-quantum Blockchain: Challenges and Research Directions. *ACM Computing Surveys, 55 (12), 251:1-32*. https://doi.org/10.1145/3572771

Chiu, R., Castañeda, C. E., Orozco-Lopez, O., & Mancilla, L. (2023). Images cipher based on convolution with chaotic maps and retrieving using. *Optics and Laser Technology*, *167*, 109680. https://doi.org/10.1016/j.optlastec.2023.109680

Delfs, H., & Knebl, H. (2007). *Introduction to Cryptography: Principles and Applications (Information Security and Cryptography* (Second Edition). Springer.

Dey, J., & Dutta, R. (2023). Progress inMultivariate Cryptography: Systematic Review, Challenges, and Research Directions. *ACM Computing Surveys, 55*(12), 1-34. https://doi.org/10.1145/3571071

Fatonah, S., Yulandari, A., & Ariyus, D. (2016). Analisis Penerapan Modifikasi Algoritma Vigenere Cipher, Caesar Cipher, Vernam Cipher dan Hill Cipher Untuk Penyisipan Pesan Dalam Gambar. *Voice of Informatic*, *8*(2), 19–30. http://dx.doi.org/10.30872/jim.v15i2.3746

Grošek, O., Antal, E., & Fabšič, T. (2019). Remarks on breaking the Vigenère autokey cipher. *Cryptologia*, *43*(6), 486-496. http://dx.doi.org/10.1080/01611194.2019.1596997

Huang, B., Gao, J., & Li, X. (2023). Efficient lattice-based revocable attribute-based encryption against decryption key exposure for cloud file sharing. *Journal of Cloud Computing*, *12*(1), 1–15. https://doi.org/10.1186/s13677-023-00414-w

Hoerudin, M., & Pratama, E. K. D. (2020). *Steganografi Teks di Audio Menggunakan Vigenere Cipher*. *January*.

Irawan, M. D. (2017). Implementasi Kriptografi Vigenere Cipher Dengan PHP. *Jurnal Teknologi Informasi 1(1)*, 11–21. http://dx.doi.org/10.36294/jurti.v1i1.21

Jaithunbi, A. K., Sabena, S., & Sairamesh, L. (2022). Preservation of Data Integrity in Public Cloud Using Enhanced Vigenere Cipher Based Obfuscation. *Wireless Personal Communications*, *129*, 271-284. http://dx.doi.org/10.1007/s11277-022-10097-2

Liwandouw, V. B., & Wowor, A. D. (2016). Kombinasi Algoritma Rubik, CPSRNG, Chaos, dan S-Box Fungsi Linier Dalam Perancangan Kriptografi Cipher Blok. *SESINDO 2015*, 2015. 207-214.

Luengo, E. A., Olivares, B. A., Villalba, L. V. G., & Hernandez-Castro, J. (2023). Further analysis of the statistical independence of the NIST SP statistical independence of the NIST SP. *Applied Mathematics and Computation*, 459: 128222. http://dx.doi.org/10.1016/j.amc.2023.128222

Ma'rifah, S. H. (2022). *Implementasi Algoritma Rubik's Cube dan Algoritma Rivest-Shamir-Adleman (RSA) pada Pengamanan Citra DIgital Iris Mata* (Undergraduate Theses). Universitas Islam Negeri Maulana Malik Ibrahim

Minarni, M., & Redha, R. (2020). Implementasi Least Significant Bit (LSB) dan Algoritma Vigenere Cipher Pada Audio Steganografi. *Jurnal Sains dan Teknologi*, *20(2)*, 168-174. http://dx.doi.org/10.36275/stsp.v20i2.268

Moosa, H., Ali, M., Alaswad, H., Elmedany, W., & Balakrishna, C. (2023). A combined Blockchain and zero-knowledge model for healthcare B2B and B2C data sharing, *Arab Journal of Basic and Applied Sciences*, *30*(1), 179-196. https://doi.org/10.1080/25765299.2023.2188701

Nana, & Prasetyo, P. W. (2021). An implementation of Hill Cipher and 3× 3 × 3 rubik ' s cube to enhance communication security. *Bulletin of Applied Mathematics and Mathematics Education*, *1*(2), 75–92. https://doi.org/10.12928/bamme.v1i2.4252

Padhye, S., Sahu., R. A., & Saraswat, V. (2018). *Introduction to Cryptography*. CRC Press

Park, S., Kim, J., Cho, K., & Yum D. H. (2020). Finding the key length of a Vigenère cipher: How to improve the twist algorithm, *Cryptologia*, *44*(3), 197-204. http://dx.doi.org/10.1080/01611194.2019.1657202

Permanasari, Y. (2017). Kriptografi Klasik Monoalphabetic. *Jurnal Teori dan Terapan Matematika*, *16*(1), 7–10. http://dx.doi.org/10.29313/jmtm.v16i1.2543

Pirandola, S., Andersen, U.L., Banchi, L., Berta, M., Bunanda, D., Colbeck, R., Englund, D., Gehring. T., Lupo, C., Attaviani, C., Pereira, J.L., Razavi, M., Shaari. S., Tommamichel. M., Usenko, V. C., Vallone, G., Villoresi, P., & Walden, P. (2020). Advances in quantum cryptography. *Advances in Optics and Photonics,* *12*(4), 1012-1236. http://dx.doi.org/10.1364/aop.361502

Pocher, N., Zichici, M., Merizzi, F., Shafiq. M. Z., & Ferretti, S. (2023). Detecting anomalous cryptocurrency transactions: An AML/CFT application of machine learning-based forensics, *Electronic Markets*, *33*(1), 37. https://doi.org/10.1007/s12525-023-00654-3

Rehman, M. U., Shafique, A., & Usman, A. B. (2023). Securing Medical Information Transmission Between IoT Devices: An Innovative Hybrid Encryption Scheme Based on Quantum Walk, DNA Encoding, and Chaos. *Internet of Things, 24, 100891*. http://dx.doi.org/10.1016/j.iot.2023.100891

Rupa, C., Greshmant., Shah, M. S. (2023). Novel secure data protection scheme using Martino homomorphic encryption, *Journal of Cloud Computing*, *12*(1), 47. http://dx.doi.org/10.1186/s13677-023-00425-7

Sumandri. (2017). Studi Model Algoritma Kriptografi Klasik dan Modern. *Prosiding Seminar Matematika Dan Pendidikan Matematika UNY 2017*, 265–272. http://dx.doi.org/10.29313/jmtm.v16i1.2543

Syahib, M. I., Riadi, I., & Umar, R. (2017). *Analisis Forensik Digital Aplikasi Beetalk Untuk Penanganan*. Cybercrime Menggunakan Metode NIST. Prosiding Semnasif *2017*, *1(1),* 134–139. http://dx.doi.org/10.30645/j-sakti.v4i1.196

Tampubolon, A. (2021). Implementasi Kombinasi Algoritma RSA dan Algoritma DES Pada Aplikasi Pengaman Pesan Teks. *Jurnal SAINTIKOM (Jurnal Sains Manajemen Informatika Dan Komputer)*, *20*(1), 38-43. http://dx.doi.org/10.53513/jis.v20i1.2598

Teranishi, K., & Kogiso, K. (2023). Optimal security parameter for encrypted control systems against eavesdropper and malicious server. *SICE Journal of Control, Measurement, and System Integration*, 16(1), 203-214. http://dx.doi.org/10.1080/18824889.2023.2215691

Trisno, I. B. (2016) Belajar Pemrograman Sulit? Coba Python. In: Buku Ajar. Ubahara Manajemen Press Surabaya.

Uniyal, N., Dobhal, G., Rawat, A., & Sikander. (2021). A Novel Encryption Approach Based on Vigenère Cipher for Secure Data Communication, *Wireless Personal Communications, 119*, 1577–1587. http://dx.doi.org/10.1007/s11277-021-08295-5

Ye, C. Q., Li, J., Chen, X. B., Hou, Y., & Wang, Z. (2023). Security and application of semi-quantum key distribution protocol for users with different quantum capabilities, *EPJ Quantum Technology*, *10*:21. http://dx.doi.org/10.1140/epjqt/s40507-023-00180-3

Younus, Z. S., & Hussain, M. K. (2022). Image steganography using exploiting modification direction for compressed encrypted data, *Journal of King Saud University-Computer and Information Sciences*, 34, 2951–2963. http://dx.doi.org/10.1016/j.jksuci.2019.04.008

Yudanto, Y. S., & Suartana, I. M. (2022). Analisis Kekuatan Enkripsi Data Pada Citra Digital Menggunakan Metode Rubiks Cube, *Journal of Informatics and Computer Science (JINACS)*, *3(4)*, 557–563. http://dx.doi.org/10.26740/jinacs.v3n04.p557-563