Edusaintek: Jurnal Pendidikan, Sains dan Teknologi
Volume 10 Issue 1 2023 Pages 329 - 352
p-ISSN: 1858-005X e-ISSN: 2655-3392 DOI: https://doi.org/10.47668/edusaintek.v10i1.702
website: https://journalstkippgrisitubondo.ac.id/index.php/EDUSAINTEK

PENGGUNAAN OPERASI *BINER X-OR* DAN *N-OR* PADA KRIPTOGRAFI *HILL CIPHER*

Deby Erdriani[1*], Mishbah Ulhusna[2], Yulia Retno Sari[3]
[1,2,3]Universitas Putra Indonesia YPTK Padang, Indonesia
*e-mail: de2bye@gmail.com

Abstract : Cryptography is information that will be conveyed to other parties in a form that is not understood. At this time the science of cryptography is needed as an exchange of information / data. It is so important that the information that will be conveyed is kept secret and safe, making cryptography a science that needs to be learned which makes it a primary need in the field of technology. Cryptography consists of 2 things, namely encryption (data/information in a form that is not understood) and description (information that is not understood is changed again to its original form). *Hill Cipher* is an algorithm that is always growing by using a matrix size with all orders, be it rectangular or square. In this study, the rectangular order is used as the key matrix. With the addition of X-OR and X-NOR logic functions. In the process it is longer and here the key sentence used is also the plaintext length of 24 characters. The k value taken here is independent, the k value used is k = 117.

Keywords: Matrix, Hill cipher, Binary Number X-OR and X-NOR

Abstrak : Ilmu kriptografi merupakan sebuah informasi yang akan disampaikan kepada pihak lain dengan bentuk yang tidak dimengerti. Pada saat ini ilmu kriptografi sangat diperlukan sebagai pertukaran informasi/data. Begitu pentingan informasinya yang akan disampaikan agar tetap rahasia dan aman, menjadikan ilmu kriptografi ini suatu ilmu yang perlu dipelajari yang menjadikan kebutuhuan primer dalam bidang teknologi . Kriptografi terdiri dari 2 hal yaitu enskripsi (data/informasi dalam bentuk yang tidak dimengerti) dan deskripsi (informasi yang tidak dimengerti dirubah lagi kebentuk asli). *Hill Cipher* merupakan algoritma yang selalu berkembang dengan menggunakan ukuran matriks dengan semua ordo baik itu persegi panjang ataupun persegi. Pada penelitian ini memakai ordo persegi panjang sebagai matriks kunci. Dengan ada penambahaan fungsi logika X-OR dan X-NOR. Pada pengerjaannya lebih panjang dan disini kalimat kunci yang digunakan juga panjang *plaintext*nya 24 karakter. Nilai k yang digunakan disini bebas, nilai k yang digunakan k = 117.

Kata Kunci: Matriks, Hill cipher, Bilangan biner X-OR dan X-NOR

PENDAHULUAN

Pertukaran informasi menjadi hal yang biasa di era digital saat ini. Namun, informasi yang bersifat rahasia membutuhkan keamanan data yang tinggi agar data yang dikirim tidak dapat dibaca atau dipahami oleh pihak yang tidak berkepentingan (Manaor et al., 2017a). Konsep penyandian merupakan salah satu cara untuk menjaga kerahasiaan data tersebut (Gusti Awang Aritonang et al., 2019). Seni dalam merahasiakan pesan yang terdapat dalam data disebut kriptografi (Arif & Fanani, 2016; Manaor et al., 2017b). Kriptografi merupakan ilmu yang mempelajari teknik-teknik matematika yang

berhubungan dengan keamanan informasi, seperti kerahasiaan data, keabsahan data, integritas data dan autentikasi data (Danny Wowor, 2013) (Gunawan et al., 2018; Juliana Pangaribuan, 2018).

Penelitian yang dilakukan oleh Abdul Halim Hasugian yang berjudul "Implementasi Algoritma *Hill Cipher* dalam Penyajian Data" membahas tentang algoritma kriptografi klasik *Hill Cipher* yang sulit dipecahkan kriptanalis apabila yang diketahui hanya berkas cipherteks saja, karena *Hill Cipher* tidak mengganti setiap abjad yang sama pada plainteks dengan abjad lainnya yang sama pada cipherteks tetapi menggunakan perkalian matriks pada dasar enkripsi dan dekripsinya (Hasugian, 2013)(Puji Widodo et al., 2016). Dasar teknik *Hill Cipher* adalah aritmatika modulo pada matriks, dengan menggunakan teknik perkalian matriks dan teknik invers terhadap matriks (Informasi & Logika, 2021) (Warni Hasibuan & Budhiati Veronica, 2022). Kunci (K) pada *Hill Cipher* adalah matriks n x n dengan n merupakan ukuran blok dan matriks K haruslah matriks yang invertible (memiliki invers) (Harefa et al., 2020)(Yasmin et al., 2017).

Penelitian yang diangkat oleh Deby Erdriani dan Dewi Devita juga membahas mengenai kriptografi dengan judul "Aplikasi Matriks pada Ilmu Kriptografi dengan Menggunakan Matlab". Hubungan antara matriks dan sandi diaplikasikan menggunakan pemograman Matlab, dimana operasi matriks yang digunakan adalah perkalian matriks dan invers matriks dengan ordo berapapun (Erdriani et al., 2021; Makhomah et al., 2021). Nikken Prima Puspita dan Nurdin Bahtiar juga melakukan penelitian tentang kriptografi dengan judul "Kriptografi *Hill Cipher* dengan Menggunakan Operasi Matriks". Pada penelitian ini, setiap karakter pada pesan terlebih dahulu harus dikonversikan ke dalam angka-angka yang bersesuaian dengan American Standart Code for Information Interchange (ASCII) sebelum masuk pada algoritma enkripsi dan dekripsi Hill Cipher. Matriks yang digunakan pada penelitian ini adalah matriks berukuran 2x2 dengan determinan matriks 1 atau -1. (Puspita, 2010)

Pemrograman yang digunakan untuk melakukan proses enkripsi dan dekripsinya adalah Delphi. Hal ini dilakukan jika teks yang dikirim cukup panjang dan menyulitkan pengguna *Hill Cipher* untuk melakukan perhitungan secara manual. Pengembangan penelitian masih bisa dilakukan dengan memperluas asumsi matriks kuncinya.

Penelitian lain yang dilakukan oleh Alz Danny Wowor berjudul "Modifikasi

Kriptografi *Hill Cipher* Menggunakan Convert Between Base" membahas tentang sebuah teknik kriptografi klasik yaitu *Hill Cipher* yang menggunakan matriks sebagai kunci. Pada tulisan ini, kriptografi *Hill Cipher* dimodifikasi menggunakan Convert Between Base (CBB) dan perkalian n-matriks kunci untuk setiap iterasi karena *Hill Cipher* memiliki beberapa kekurangan, diantaranya: 1) Algoritmanya dirancang hanya dapat mengenkripsi karakter alfabet saja. 2) Cipherteks yang dihasilkan hanya dalam karakter abjad. 3) Jumlah elemen plainteks sama dengan cipherteks. Dengan adanya modifikasi ini dapat menahan kriptanalis know-plaintext attack yang sudah memecahkan *Hill Cipher* dengan teknik perkalian matriks dan fungsi linier. (Danny Wowor, 2013)

Penggunaan CBB merupakan kunci awal dan terakhir pada modifikasi Hill Cipher, yang dapat membuat elemen cipherteks lebih banyak dari plainteks serta dapat mempersulit kriptanalis untuk mencari hubungan antara plainteks dengan cipherteks karena plainteks dalam bentuk karakter tetapi cipherteks berbentuk bit biner.

Berdasarkan literatur di atas, penulis tertarik membahas mengenai kriptografi *Hill Cipher* yang merupakan salah satu teknik kriptografi klasik yang menggunakan operasi matriks dan modulo yang cukup kompleks. Namun, seiring dengan perkembangan ilmu pengetahuan, matriks kunci yang digunakan dapat dimodifikasi dengan pseudo invers dan operasi biner yang menghasilkan cipherteks yang lebih rumit dan menyulitkan kriptanalis untuk menemukan kuncinya (Dwitiyanti & Satria Setiawan, 2021).

Tabel 1. Korespondesi karakter dengan angka desimal.

Karakter	Nilai Konversi	Karakter	Nilai Konversi	Karakter	Nilai Konversi	Karakter	Nilai Konversi	Karakter	Nilai Konversi
A	0	T	19	m	38	5	57	}	76
B	1	U	20	n	39	6	58	\	77
C	2	V	21	o	40	7	59	|	78
C	3	W	22	p	41	8	60	`	79
E	4	X	23	q	42	9	61	~	80
F	5	Y	24	r	43	spasi	62	!	81
G	6	Z	25	s	44	,	63	@	82
H	7	a	26	t	45	<	64	#	83
I	8	b	27	u	46	.	65	$	84
J	9	c	28	v	47	>	66	%	85
K	10	d	29	w	48	/	67	^	86
L	11	e	30	x	49	?	68	&	87
M	12	f	31	y	50	;	69	*	88
N	13	g	32	z	51	:	70	(	89
O	14	h	33	0	52	‘	71	)	90

P	15	i	34	1	53	“	72	-	91
Q	16	j	35	2	54	[	73	—	92
R	17	k	36	3	55	{	74	=	93
S	18	l	37	4	56	]	75	+	94

METODE

Pada penelitian ini kami menggunakan metode studi literatur dengan mengumpulkan artikel dari jurnal-jurnal yang berhubungan dengan algoritma kriptografi Hill Cipher, aritmatika modulo, teori dasar matriks dan pseudo invers. (Aribowo et al., 2021) Algoritma kriptografi terbagi atas dua, yaitu algoritma kriptografi simetrik dan algoritma kriptografi asimetrik (Alawiyah, 2016; Putera & Siahaan, 2016). *Hill Cipher* dengan Pseudo Invers merupakan algoritma yang cukup kuat pada algoritma kriptografi simetris. Pada algoritma ini terdapat tiga proses yang dioperasikan atas bilangan modulo *p* yaitu :

1. Proses Algoritma Inisialisasi Kunci

Proses ini bermaksud untuk menyelidiki matriks yang digunakan sebagai kunci. Matriks kunci yang diselidiki adalah matriks yang memiliki *invers* atau *pseudo invers* yang akan digunakan untuk proses deskripsi. Seandainya matriks tersebut tidak memiliki *invers* atau *pseudo invers* maka matriks tersebut tidak dapat digunakan sebagai matriks kunci. Algoritma Inisialisasi kunci terdiri atas beberapa bagian ;

a. Tentukan sebuah matriks A_{mxn}, hitung rank matriks A. Jika $rank\ (A) \neq m$ atau $rank\ (A) \neq n$, maka proses berhenti dan matriks tidak dapat dijadikan matriks kunci.

b. Tentukan *pseudo invers* matriks $A(A^{\#})$

i. Jika $m = n$ dan $rank(A) = m$, maka *pseudo invers* matriks A sama dengan *invers* matriks $A(A^{\#} = A^{-1})$

ii. Jika $m > n$ dan $rank(A) = n$, maka $A^{\#} = (A^T A)^{-1} A^T$

iii.Jika $m < n$ dan $rank\ (A) = m$, maka $A^{\#} = A^T (AA^T)^{-1}$

c. *Pseudo invers* matriks $A(A^{\#})$ harus memenuhi syarat berikut ini :

i. $AA^{\#}A = A$

ii. $A^{\#}AA^{\#} = A^{\#}$

iii. $(AA^{\#})^{*} = AA^{\#}$, $(AA^{\#})^{*}$ adalah matriks hermitian

iv. $(A^{\#}A)^{*} = A^{\#}A$, $(A^{\#}A)^{*}$ adalah matriks hermitian

Jika syarat-syarat diatas dipenuhi oleh matriks A, maka matriks A dapat digunakan sebagai matriks kunci.

2. Proses Algoritma Enkripsi

Pada proses algoritma enkripsi dengan menggunakan *pseudo invers* ada beberapa langkah sebagai berikut :

a. Hitung panjang *plaintext l*, jika $l \; mod \; r \neq 0$, maka tambahkan karakter hingga didapatkan $l \; mod \; r = 0$. Dimana *r* adalah $rank$ matriks kunci.

b. Konversi *plaintext* kedalam bilangan desimal sesuai data pada tabel.

c. Bagi *plaintext* kedalam blok-blok $P_1, P_2, \dots, P_i$, dimana $i = 1, \dots, \frac{l}{r}$ dengan masing-masing blok terdiri dari *r* elemen.

d. Hitung $C_1, C_2, \dots, C_i$ dengan syarat ;

i. $C_i = (A(P_i)^T)^T$ jika $m \geq n$

ii. $C_i = P_i A$ jika $m < n$

e. Gabungkan $C_1, C_2, \dots, C_i$ lalu konversikan ke dalam bentuk karakter sesuai tabel sehingga didapatkan sebuah c*iphertext.*

3. Proses Algoritma Deskripsi

Proses Algoritma Deskripsi pada *Hill Cipher* dengan *pseudo invers* terdapat beberapa tahapan, adalah ;

a. Konversikan *ciphertext* kedalam bentuk bilangan desimal sesuai tabel.

b. Bagi *ciphertext* kedalam blok-blok $C_1, C_2, \dots, C_i$ dengan $i = 1, \dots, \frac{l}{j}$, dimana setiap blok terdiri dari *j* elemen.

i. $j = m$ jika $m \geq n$

ii. $j = n$ jika $m < n$

c. Hitung $P_1, P_2, \dots, P_i$ dengan ketetapan ;

i. $P_i = (A^{\#}(C_i)^T)^T$ jika $m \geq n$

ii. $P_i = C_i A^{\#}$ jika $m < n$

d. Gabungkan $P_1, P_2, \dots, P_i$ lalu konversikan ke dalam bentuk karakter sesuai tabel sehingga didapatkan *plaintext.*

HASIL DAN PEMBAHASAN

Pada bagian hasil dan pembahasan untuk Kriptografi Hill Chiper ini terdapat tiga proses yang dioperasikan atas bilangan modulo *p* yaitu :

1. Proses Algoritma Inisialisasi Matriks Kunci.
2. Proses Algoritma Enkripsi
3. Proses Algoritma Deskripsi

Selanjutnya kami akan memaparkan proses pembahasan pada algoritma Hill Chiper ini adalah sebagai berikut :

1. Proses Algoritma Inisialisasi Matriks Kunci

Matriks kunci hasil yang digunakan memiliki ordo 2 x 3, yang terdiri 2 baris dan 3 kolom.

$$A_{2x3} = \begin{bmatrix} 1 & 2 & 1 \\ 4 & 3 & 1 \end{bmatrix}$$

Jika m < n dan Rak (A) = m, maka

$$A^{\#} = A^T\left((A.A^T)\right)^{-1}$$

Matriks A^T dan $A.A^T$ diperoleh sebagai berikut :

$$A^T = \begin{bmatrix} 1 & 4 \\ 2 & 3 \\ 1 & 1 \end{bmatrix} \qquad dan \quad A.A^T = \begin{bmatrix} 6 & 11 \\ 11 & 26 \end{bmatrix}$$

Determinan dari $|A.A^T| = 35$

fpb (p, det $|A.A^T|$) $\rightarrow$fpb(95, 35)

$\rightarrow$95 = (35 x 2) + 20

$\rightarrow$35 = (20 x 1) + 15

$\rightarrow$20 = (15 x 1) + 5

$\rightarrow$15 = (5 x 3) + 0

fpb (95, 35) = 3, berarti memiliki invers

$$(A.A^T)^{-1} = \begin{bmatrix} \frac{26}{35} & -\frac{11}{35} \\ -\frac{11}{35} & \frac{6}{35} \end{bmatrix}$$

Maka diperoleh matriks $A^{\#}$ adalah pseudo invers dari matriks kunci A dengan rumus :

$$A^{\#} = A^T (A.A^T)^{-1}$$

$$A^{\#} = \begin{bmatrix} -\frac{18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & -\frac{1}{7} \end{bmatrix}$$

Syarat-syarat p-invers :

1. $A.A^{\#}.A$ dipenuhi
2. $A^{\#}.A.A^{\#} = A^{\#}$ dipenuhi
3. $(A.A^{\#})^{*} = A.A^{\#}$ dipenuhi
4. $(A^{\#}A)^{*} = A^{\#}A$ dipenuhi

Terpenuh syarat-syarat diatas dapat disimpulkan bahwa matrik A dapat digunakan sebagai matriks kunci.

2. Proses Algoritma Enkripsi

Nilai sembarang k ditentukan adalah 117, dapat ditulis datanya sebagai berikut :

Plaintext : UPI Melangkah Lebih Maju

k = 117 = 01110101

rank(A) = 2

Panjang Plaintext = 24

Korespondensikan Plaintext kedalam angka karakter angka desimal.

Tabel 2. Konversi Bilangan decimal Plaintext kedalam bilangan biner

Plaintext	Bilangan Biner	Plaintext	Bilangan Biner
20	10100	33	100001
15	1111	62	111110
8	1000	11	1011
62	111110	30	11110
12	1100	27	11011
30	11110	34	100010
37	100101	33	100001
26	11010	62	111110
39	100111	12	1100
32	100000	26	11010
36	100100	35	100011
26	11010	46	101110

P = [20 15 8 62 12 30 37 26 39 32 36 26 33 62 11 30 27 34 33 62 12 26 35 46]

Konversi bilangan decimal plaintext kedalam bilangan biner. Hasil konversi dapat dilihat pada table berikut. Kemudian lakukan proses enkripsi menggunakan fungsi logika X-OR antara bilangan biner dengan kunci k. Partisi setiap blok biner hasil dari X-OR menjadi 2 bagian terdiri dari 4 bit biner. Kemudian konversi ke bilangan desimal.

Tabel 3. Proses Enskrpsi menggunakan fungsi X-OR

Plaintext	00010100	00001111	00001000	00111110	00001100
Kunci K	01110101	01110101	01110101	01110101	01110101
X-Or	01100001	01111010	01111101	1001011	01111001
Plaintext	00011110	00100101	00011010	00100111	00100000
Kunci K	01110101	01110101	01110101	01110101	01110101
X-Or	01101011	01010000	11101111	01010010	01010101
Plaintext	00100100	00011010	00100001	00111110	00001011
Kunci K	01110101	01110101	01110101	01110101	01110101
X-Or	01010001	01101111	01010100	01001011	01111110
Plaintext	00011110	00011011	00100010	00100001	00111110
Kunci K	01110101	01110101	01110101	01110101	01110101
X-Or	01101011	01101110	01010111	01010100	01001011
Plaintext	00001100	00011010	00100011	00101110	
Kunci K	01110101	01110101	01110101	01110101	
X-Or	01111001	01101111	01010110	01011011	

Didapatkan deret bilangan decimal dari hasil konversi biner yaitu :

P = [6 1 7 10 7 13 4 11 7 9 6 11 5 0 14 15 5 2 5 5 5 1 6 15 5 4 4 11 7 14 6 11 6 14 5 7 5 4 4 11 7 9 6 15 5 6 5 11].

P dipartisi menjadi beberapa blok yang masing-masing blok memiliki 2 elemen sesuai dengan rank pada matriks A,

P_1 = [6 1]

P_2 = [7 10]

⋮

P_{24} = [5 6]

Jika m < n, matriks kunci full row rank maka rumus yang digunakan $C_i = P_i . A$.

Tabel 4 . Proses Enkripsi konversi bilangan biner kedalam decimal

Biner	Desimal	Biner	Desimal	Biner	Desimal
0110	6	0101	5	0110	6
0001	1	0010	2	1110	14
0111	7	0101	5	0101	5

1010	10	0101	5	0111	7
0111	7	0101	5	0101	5
1101	13	0001	1	0100	4
0100	4	0110	6	0100	4
1011	11	1111	15	1011	11
0111	7	0101	5	0111	7
1001	9	0100	4	1001	9
0110	6	0100	4	0110	6
1011	11	1011	11	1111	15
0101	5	0111	7	0101	5
0000	0	1110	14	0110	6
1110	14	0110	6	0101	5
1111	15	1011	11	1011	11

Plaintext (l) = 48

Rank (A) = 2

$$C_1 = [6 \quad 1]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [10 \quad 15 \quad 7]$$
$$C_2 = [7 \quad 10]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [47 \quad 44 \quad 17]$$
$$C_3 = [7 \quad 13]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [59 \quad 53 \quad 20]$$
$$C_4 = [4 \quad 11]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [48 \quad 41 \quad 15]$$
$$C_5 = [7 \quad 9]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [43 \quad 41 \quad 16]$$
$$C_6 = [6 \quad 11]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [50 \quad 45 \quad 17]$$
$$C_7 = [5 \quad 0]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [5 \quad 10 \quad 5]$$
$$C_8 = [14 \quad 15]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [74 \quad 73 \quad 29]$$
$$C_9 = [5 \quad 2]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [13 \quad 16 \quad 7]$$
$$C_{10} = [5 \quad 5]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [25 \quad 25 \quad 10]$$
$$C_{11} = [5 \quad 1]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [9 \quad 13 \quad 6]$$
$$C_{12} = [6 \quad 15]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [66 \quad 57 \quad 21]$$
$$C_{13} = [5 \quad 4]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [21 \quad 22 \quad 9]$$
$$C_{14} = [4 \quad 11]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [48 \quad 41 \quad 15]$$
$$C_{15} = [7 \quad 14]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [63 \quad 56 \quad 21]$$
$$C_{16} = [6 \quad 11]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [50 \quad 45 \quad 17]$$
$$C_{17} = [6 \quad 14]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [62 \quad 54 \quad 20]$$

$$C_{18} = [5 \quad 7]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [33 \quad 31 \quad 12]$$
$$C_{19} = [5 \quad 4]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [21 \quad 22 \quad 9]$$
$$C_{20} = [4 \quad 11]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [48 \quad 41 \quad 15]$$
$$C_{21} = [7 \quad 9]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [43 \quad 41 \quad 16]$$
$$C_{22} = [6 \quad 15]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [66 \quad 57 \quad 21]$$
$$C_{23} = [5 \quad 6]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [29 \quad 28 \quad 11]$$
$$C_{24} = [5 \quad 11]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [49 \quad 43 \quad 16]$$

Dari perkalian matriks diatas, hasil dari perkalian matrik dikonversikan kedalam decimal. Dimana konversi dari elemen matriks bilangan decimal kedalam bilangan biner 8 bit dapat dilihat pada table dibawah ini.

Tabel 5. Proses konversi dari bilangan decimal menjadi bilangan biner

C_i	Bilangan Biner	C_i	Bilangan Biner	C_i	Bilangan Biner
10	00001010	13	00001101	62	00111110
15	00001111	16	00010000	54	00110110
7	00000111	7	00000111	60	00111100
47	00101111	25	00011001	33	00100001
44	00101100	25	00011001	31	00011111
17	00010001	10	00001010	12	00001100
59	00111011	9	00001001	21	00010101
53	00110101	13	00001101	22	00010110
20	00010100	6	00000110	9	00001001
48	00110000	66	01000010	48	00110000
41	00101001	57	00111001	41	00101001
15	00001111	21	00010101	15	00001111
43	00101011	21	00010101	43	00101011
41	00101001	22	00010110	41	00101001
16	00010000	9	00001001	16	00010000

50	00110010	48	00110000	66	01000010
45	00101101	41	00101001	57	00111001
17	00010001	15	00001111	21	00010101
5	00000101	63	00111111	29	00011101
10	00001010	56	00111000	28	00011100
5	00000101	21	00010101	11	00001011
74	01001010	50	00110010	49	00110001
73	01001001	45	00101101	43	00101011
29	00011101	17	00010001	16	00010000

Setelah mengkonversikan bilangan decimal ke biner maka gunakan fungsi logika XNOR untuk setiap 8 bit biner dengan menggunakan kunci k prosesnya dapat dilihat seperti table dibawah ini.

Tabel 6. Proses Enkripsi menggunakan fungsi XNOR

C_i	00001010	00001111	00001111	00101111	00101100	Ci	00111100	00100001	00011111	00001100	00010101
Kunci k	01110101	01110101	01110101	01110101	01110101	Kunci k	01110101	01110101	01110101	01110101	01110101
XNOR	10000000	10000101	10001101	10100101	10100110	XNOR	10110110	10101011	10010101	10000110	10011111
Ci	00010001	00111011	00110101	00010100	00110000	Ci	00010110	00001001	00110000	00101001	00001111
Kunci k	01110101	01110101	01110101	01110101	01110101	Kunci k	01110101	01110101	01110101	01110101	01110101
XNOR	10011011	10110001	10111111	10011110	10111010	XNOR	10011100	10000011	10111010	10100011	10000101
Ci	00101001	00001111	00101011	00101001	00010000	Ci	00101011	00101001	00010000	00100010	00111001
Kunci k	01110101	01110101	01110101	01110101	01110101	Kunci k	01110101	01110101	01110101	01110101	01110101
XNOR	10100011	10000101	10100001	10100011	10011010	XNOR	10111111	10100011	10011010	11001000	10110011

Ci	0011 0010	001 011 01	0001 0001	0000 0101	0000 1010	Ci	00010 101	00011 101	00011 100	00001 011	00110 001
Kunci k	0111 0101	011 101 01	0111 0101	0111 0101	0111 0101	Kunci k	01110 101	01110 101	01110 101	01110 101	01110 101
XNOR	1011 1000	101 001 11	1001 1011	1000 1111	1000 0000	XNOR	10011 111	10010 111	10010 110	10000 001	10111 011
Ci	0000 0101	010 010 10	0100 1001	0001 1101	0000 1101	Ci	00101 011	00010 000			
Kunci k	0111 0101	011 101 01	0111 0101	0111 0101	0111 0101	Kunci k	01110 101	01110 101			
XNOR	1000 1111	110 000 00	1100 0011	1001 0111	1000 0111	XNOR	10100 001	10011 010			
Ci	0001 0000	000 001 11	0001 1001	0001 1001	0000 1010						
Kunci k	0111 0101	011 101 01	0111 0101	0111 0101	0111 0101						
XNOR	1001 1010	100 011 01	1001 0011	1001 0011	1000 0000						
Ci	0000 1001	000 011 01	0000 0110	0100 0010	0011 1001						
Kunci k	0111 0101	011 101 01	0111 0101	0111 0101	0111 0101						
XNOR	1000 0011	100 001 11	1000 1100	1100 1000	1011 0011						
Ci	0001 0101	000 101 01	0001 0110	0000 1001	0011 0000						
Kunci k	0111 0101	011 101 01	0111 0101	0111 0101	0111 0101						
XNOR	1001 1111	100 111 11	1001 1100	1000 0011	1011 1010						
Ci	0010 1001	000 011 11	0011 1111	0011 1000	0001 0101						
Kunci k	0111 0101	011 101 01	0111 0101	0111 0101	0111 0101						
XNOR	1010 0011	100 001 01	1011 0101	1011 0010	1001 1111						
Ci	0011 0010	001 011 01	0001 0001	0011 1110	0011 0110						

Kunci k	0111 0101	011 101 01	0111 0101	0111 0101	0111 0101
XNOR	1011 1000	101 001 11	1001 1011	1011 0100	1011 1100

Table 7. Partisi bilangan biner hasil proses XNOR menjadi 4 bit, lalu konversikan kedalam bilangan decimal.

Biner	Desimal	Biner	Desimal	Biner	Desimal
1000	8	1000	8	1011	11
0000	0	0111	7	0100	4
1000	8	1001	9	1011	11
0101	5	1010	10	1100	12
1000	8	1000	8	1011	11
1101	13	1101	13	0110	6
1010	10	1001	9	1010	10
0101	5	0011	3	1011	11
1010	10	1000	8	1001	9
0110	6	0000	0	0101	5
1001	9	1000	8	1000	8
1011	11	0011	3	0110	6
1011	11	1000	8	1001	9
0001	1	0111	7	1111	15
1011	11	1000	8	1001	9
1111	15	1100	12	1100	12
1001	9	1100	12	1000	8
1110	14	1000	8	0011	3
1011	11	1011	11	1011	11
1010	10	0011	3	1010	10
1010	10	1001	9	1010	10
0011	3	1111	15	0011	3
1000	8	1001	9	1000	8
0101	5	1111	15	0101	5
0101	5	1001	9	0101	5
0001	1	1100	12	1111	15
1010	10	1000	8	1010	10
0011	3	0011	3	0011	3
1001	9	1011	11	1001	9
1010	10	1010	10	1010	10
1011	11	1010	10	1100	12
1000	8	0011	3	1000	8
1010	10	1000	8	1011	11
0111	7	0101	5	0011	3
1001	9	1011	11	1001	9
1011	11	0101	5	1111	15
1000	8	1011	11	1001	9
1111	15	0010	2	0111	7
1000	8	1011	11	1001	9
0000	0	0010	2	0110	6

1000	8	1001	9	1000	8
1111	15	1111	15	0001	1
1100	12	1011	11	1011	11
0000	0	1000	8	1011	11
1100	12	1010	10	1010	10
0011	3	0111	7	0001	1
1001	9	1001	9	1001	9
0111	7	1011	11	1010	10

Deret bilangan decimal yang diperoleh dari table diatas adalah [8, 0, 8, 5 ,8, 13, 10, 5, 10, 6, 9, 11, 11, 1, 11, 15, 9, 14, 11, 10, 10, 3, 8, 5, 5, 1, 10, 3, 9, 10, 11, 8, 10, 7, 9, 11, 8, 15, 8, 0, 8, 15, 12, 0, 12, 3, 9, 7, 8, 7, 9, 10, 8, 13, 9, 3, 8, 0, 8, 3, 8, 7, 318, 12, 12, 8, 11, 3, 9, 15, 9, 15, 9, 12, 8, 3, 11, 10, 10, 3, 8, 5, 11, 5, 11, 2, 11, 2, 9, 15, 11, 8, 10, 7, 9, 11, 11, 4, 11, 12, 11, 6, 10, 11, 9, 5, 8, 6, 9, 15, 9, 12, 8, 3, 11, 10, 10, 3, 8, 5, 5, 15, 10, 3, 9, 10, 12, 8, 11, 3, 9, 15, 9, 7, 9, 6, 8, 1, 11, 11, 10, 1, 9, 10].

Partisi bilangan decimal kedalam blok-blok P_1, P_2 , …, P_{72} dengan jumlah masing-masing blok 2 elemen yang sesuai dengan Rank(A). Hitunglah nilai C_i dengan menggunakan rumus $C_i = P_i$. A

$$C_1 = [8 \quad 0]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [8 \quad 16 \quad 8]$$
$$C_2 = [8 \quad 5]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [28 \quad 31 \quad 13]$$
$$C_3 = [8 \quad 13]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [60 \quad 55 \quad 21]$$
$$\vdots$$
$$C_{72} = [9 \quad 10]\begin{bmatrix}1 & 2 & 1\\ 4 & 3 & 1\end{bmatrix} mod\ 95 = [49 \quad 48 \quad 19]$$

Gabungan C_1 , C_2 , … , C_{72} Untuk mendapatkan Ciphertext dalam bentuk decimal, sehingga didapatkan [8, 16, 8, 28, 31, 13, 60, 55, 21, 30, 35, 15, 34 38, 16, 53, 51, 20, 15, 25, 12, 71, 67, 26, 65, 60, 23, 51, 52, 21, 22, 29, 13, 28, 31, 13, 9, 13, 6, 22, 29, 13, 49, 48, 19, 43, 46, 19, 38, 41, 17, 53, 51, 20, 68, 61, 23, 8, 16, 8, 68, 61, 23, 12, 24, 12, 24, 33, 15, 37, 39, 16, 36, 37, 15, 49, 48, 19, 60, 55, 21, 21, 27, 12, 8, 16, 8, 20, 25, 11, 36, 37, 15, 56, 52, 20, 44, 48, 20, 23,31, 14, 69, 63, 24, 69, 63, 24, 57, 54, 21, 20, 25, 11, 51, 52, 21, 22, 29, 13, 28, 31, 13, 31, 37, 16, 19, 28, 13, 19, 28, 13, 69, 63, 24, 43, 46, 19, 38, 41, 17, 53, 51, 20, 27, 34, 15, 59, 58, 23, 35, 40, 17, 54, 53, 21, 29, 33, 14, 32, 34, 14, 69, 63, 24, 57, 54, 21, 20, 25, 11, 51, 52, 21, 22, 29, 13, 28, 31, 13, 65, 55, 20, 22, 29, 13, 49, 48, 19, 44, 48, 20, 23, 31, 14, 69, 63, 24, 37, 39, 16, 33, 36, 15, 12, 19, 9, 55, 55, 22, 14, 23, 11, 49, 48, 19]

Ciphertext yang berupa angka dikorespondesikan kedalam karakter didapatkan sebagai berikut :

IQIcfN83VejPimQ1zzUPZM'/a.8Xz0VWdNcfNJNGWdNxwTruTmpR1zU?9XIQI ?9XMYMYMYhPlnQklPxwT83VVbMIQIUZLklP40UswUswUXfO;,Y;,Y52VUZ LzVWdNcfNflQTcNTcN;,YruTmpR1zUbiP76XjoR21VdhOgiO;,Y52VUZLyzVW dNcfN.3UWdNxwTswUXfO;,YlnQhkPMTJ33WOXLxwT

3. Proses Algoritma Deskripsi

Proses deskripsi dilakukan menggunakan persamaan $P_i = C_i$. A jika m < n matrik kunci full row rank. Partisi matriks C menjadi beberapa blok matriks yang masing-masing terdiri dari 3 elemen sehingga didapatkan (Baris pada matriks $A^{\#}$ 3 baris).

$C_1 = [8 \quad 16 \quad 8]$

$C_2 = [28 \quad 31 \quad 13]$

$C_3 = [60 \quad 55 \quad 21]$

⋮

$C_{72} = [49 \quad 48 \quad 19]$

$$P_1 = [8 \quad 16 \quad 8] \cdot \begin{bmatrix} -\frac{18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & -\frac{1}{7} \end{bmatrix} mod\ 95 = [8 \quad 0]$$

$$P_2 = [28 \quad 31 \quad 13] \cdot \begin{bmatrix} -\frac{18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & -\frac{1}{7} \end{bmatrix} mod\ 95 = [8 \quad 5]$$

$$P_3 = [60 \quad 55 \quad 21] \cdot \begin{bmatrix} -\frac{18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & -\frac{1}{7} \end{bmatrix} mod\ 95 = [8 \quad 13]$$

…

$$P_{72} = [49 \quad 48 \quad 19] \cdot \begin{bmatrix} -\frac{18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & -\frac{1}{7} \end{bmatrix} mod\ 95 = [9 \quad 10]$$

Gabungkan P_1, $P_{2,...}$, P_3 sehingga didapatkannya deret bilangan decimal sebagai beriku : [8, 0, 8, 5 ,8, 13, 10, 5, 10, 6, 9, 11, 11, 1, 11, 15, 9, 14, 11, 10, 10, 3, 8, 5, 5, 1, 10, 3, 9, 10, 11, 8, 10, 7, 9, 11, 8, 15, 8, 0, 8, 15, 12, 0, 12, 3, 9, 7, 8, 7, 9, 10, 8, 13, 9, 3, 8, 0, 8, 3, 8, 7, 318, 12, 12, 8, 11, 3, 9, 15, 9, 15, 9, 12, 8, 3, 11, 10, 10, 3, 8, 5, 11, 5, 11, 2, 11, 2, 9, 15, 11, 8, 10, 7, 9, 11, 11, 4, 11, 12, 11, 6, 10, 11, 9, 5, 8, 6, 9, 15, 9, 12, 8, 3, 11, 10, 10, 3, 8, 5, 5, 15, 10, 3, 9, 10, 12, 8, 11, 3, 9, 15, 9, 7, 9, 6, 8, 1, 11, 11, 10, 1, 9, 10].

Proses deskripsi diatas memperoleh 144 elemen. Selanjutnya konversi bilangan decimal tersebut kebilangan biner 4 bit.

Tabel 8. Konversi bilangan decimal tersebut kebingan biner 4 bit.

Desimal	Biner	Desimal	Biner	Desimal	Biner	Desimal	Biner	Desimal	Biner	Desimal	Biner
8	1000	8	1000	11	1011	15	1111	2	10	7	111
0	**0**	7	**111**	4	**100**	8	1000	11	1011	9	1001
8	1000	9	1001	11	1011	0	0	2	10	6	110
5	101	10	1010	12	1100	8	1000	9	1001	8	1000
8	1000	8	1000	11	1011	15	1111	15	1111	1	1
13	1101	13	1101	6	110	12	1100	11	1011	11	1011
10	1010	9	1001	10	1010	0	0	8	1000	11	1011
5	101	3	11	11	1011	12	1100	10	1010	10	1010
10	1010	8	1000	9	1001	3	11	7	111	1	1
6	110	0	0	5	101	9	1001	9	1001	9	1001
9	1001	8	1000	8	1000	7	111	11	1011	10	1010
11	1011	3	11	6	110						
11	1011	8	1000	9	1001						
1	1	7	111	15	1111						
11	1011	8	1000	9	1001						
15	1111	12	1100	12	1100						
9	1001	1	1100	8	1000						

		2			
14	1110	8	1000	3	11
11	1011	11	1011	11	1011
10	1010	3	11	10	1010
10	1010	9	1001	10	1010
3	11	15	1111	3	11
8	1000	9	1001	8	1000
5	101	15	1111	5	101
5	101	9	1001	5	101
1	1	12	1100	15	1111
10	1010	8	1000	10	1010
3	11	3	11	3	11
9	1001	11	1011	9	1001
10	1010	10	1010	10	1010
11	1011	10	1010	12	1100
8	1000	3	11	8	1000
10	1010	8	1000	11	1011
7	111	5	101	3	11
9	1001	11	1011	9	1001
11	1011	5	101	15	1111
8	1000	11	1011	9	1001

Gabungkan dua blok deret menjadi satu blok deret biner, kemudian operasikan fungsi logika XNOR antara ciphertext dengan biner kunci k.

Tabel 9. Operasikan fungsi logika XNOR antara ciphertext dengan biner kunci k

Ciphertext	10000000	10000101	10001101	10100101	10100110	Ciphertext	10011100	10000011	10111010	10100011	10000101
Kunci k	01110101	01110101	01110101	01110101	01110101	Kunci k	01110101	01110101	01110101	01110101	01110101

XNOR	**0000 1010**	**000 011 11**	**000 001 11**	**0010 1111**	**0010 1100**	**XNOR**	**0001 0110**	**0000 1001**	**0011 0000**	**0010 1001**	**0000 1111**
Ciphertext	1001 1011	101 100 01	101 111 11	1001 1110	1011 1010	Ciphertext	1011 111	1010 0011	1001 1010	1100 1000	1011 0011
Kunci k	0111 0101	011 101 01	011 101 01	0111 0101	0111 0101	Kunci k	0111 0101	0111 0101	0111 0101	0111 0101	0111 0101
XNOR	**0001 0001**	**001 110 11**	**001 101 01**	**0001 0100**	**0011 0000**	**XNOR**	**0010 1011**	**0010 1001**	**0001 0000**	**0010 0001 0**	**0011 1001**
Ciphertext	1010 0011	100 001 01	101 000 01	1010 0011	1001 1010	Ciphertext	1001 1111	1001 0111	1001 0110	1000 0001	1011 1011
Kunci k	0111 0101	011 101 01	011 101 01	0111 0101	0111 0101	Kunci k	0111 0101	0111 0101	0111 0101	0111 0101	0111 0101
XNOR	**0010 1001**	**000 011 11**	**001 010 11**	**0010 1001**	**0001 0000**	**XNOR**	**0001 0101**	**0001 1101**	**0001 1100**	**0000 1011**	**0011 0001**
Ciphertext	1011 1000	101 001 11	100 110 11	1000 1111	1000 0000	Ciphertext	1010 0001	1001 1010			
Kunci k	0111 0101	011 101 01	011 101 01	0111 0101	0111 0101	Kunci k	0111 0101	0111 0101			
XNOR	**0011 0010**	**001 011 01**	**000 100 01**	**0000 0101**	**0000 1010**	**XNOR**	**0010 1011**	**0001 0000**			
Ciphertext	1000 1111	110 000 00	110 000 11	1001 0111	1000 0111						
Kunci k	0111 0101	011 101 01	011 101 01	0111 0101	0111 0101						
XNOR	**0000 0101**	**010 010 10**	**010 010 01**	**0001 1101**	**0000 1101**						
Ciphertext	1001 1010	100 011 01	100 100 11	1001 0011	1000 0000						
Kunci k	0111 0101	011 101 01	011 101 01	0111 0101	0111 0101						
XNOR	**0001 0000**	**000 001 11**	**000 110 01**	**0001 1001**	**0000 1010**						
Ciphertext	1000 0011	100 001 11	100 011 00	1100 1000	1011 0011						

Kunci k	0111 0101	011 10101	011 10101	0111 0101	0111 0101
XNOR	**0000 1001**	**000 01101**	**000 00110**	**0100 0010**	**0011 1001**
Ciphertext	1001 1111	100 11111	100 11100	1000 0011	1011 1010
Kunci k	0111 0101	011 10101	011 10101	0111 0101	0111 0101
XNOR	**0001 0101**	**000 10101**	**000 10110**	**0000 1001**	**0011 0000**
Ciphertext	1010 0011	100 00101	101 10101	1011 0010	1001 1111
Kunci k	0111 0101	011 10101	011 10101	0111 0101	0111 0101
XNOR	**0010 1001**	**000 01111**	**001 11111**	**0011 1000**	**0001 0101**
Ciphertext	1011 1000	101 00111	100 11011	1011 0100	1011 1100
Kunci k	0111 0101	011 10101	011 10101	0111 0101	0111 0101
XNOR	**0011 0010**	**001 01101**	**000 10001**	**0011 1110**	**0011 0110**
Ciphertext	1011 0110	101 01011	100 10101	1000 0110	1001 1111
Kunci k	0111 0101	011 10101	011 10101	0111 0101	0111 0101
XNOR	**0011 1100**	**001 00001**	**000 11111**	**0000 1100**	**0001 0101**

Tabel 10. Konversi bilangan deret diatas ke bilangan decimal. Maka didapatkan hasilnya sebagai berikut.

Biner	Desimal	Biner	Desimal	Biner	Desimal
00001010	10	00001101	13	00111110	62
00001111	15	00010000	16	00110110	54
00000111	7	00000111	7	00111100	60
00101111	47	00011001	25	00100001	33

00101100	44	00011001	25	00011111	31
00010001	17	00001010	10	00001100	12
00111011	59	00001001	9	00010101	21
00110101	53	00001101	13	00010110	22
00010100	20	00000110	6	00001001	9
00110000	48	01000010	66	00110000	48
00101001	41	00111001	57	00101001	41
00001111	15	00010101	21	00001111	15
00101011	43	00010101	21	00101011	43
00101001	41	00010110	22	00101001	41
00010000	16	00001001	9	00010000	16
00110010	50	00110000	48	01000010	66
00101101	45	00101001	41	00111001	57
00010001	17	00001111	15	00010101	21
00000101	5	00111111	63	00011101	29
00001010	10	00111000	56	00011100	28
00000101	5	00010101	21	00001011	11
01001010	74	00110010	50	00110001	49
01001001	73	00101101	45	00101011	43
00011101	29	00010001	17	00010000	16

Partisi bilangan desimal diatas kedalam blok-blok masing-masing terdiri 3 elemen sehingga terbentuk 18 blok.

$$C_1 = [10 \quad 15 \quad 7]$$

$$C_2 = [47 \quad 44 \quad 17]$$

$$C_3 = [59 \quad 53 \quad 20]$$

...

$$C_{18} = [49 \quad 43 \quad 16]$$

Cari nilai dari P_1 sampai P_{18}

$$P_1 = [10 \quad 15 \quad 7] \cdot \begin{bmatrix} \frac{-18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & \frac{-1}{7} \end{bmatrix} mod\ 95 = [6 \quad 1]$$

$$P_2 = [47 \quad 44 \quad 17].\begin{bmatrix} \frac{-18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & \frac{-1}{7} \end{bmatrix} mod\ 95 = [7 \quad 10]$$

$$P_3 = [59 \quad 53 \quad 20].\begin{bmatrix} \frac{-18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & \frac{-1}{7} \end{bmatrix} mod\ 95 = [7 \quad 13]$$

...

$$P_{18} = [49 \quad 43 \quad 16].\begin{bmatrix} \frac{-18}{35} & \frac{13}{35} \\ \frac{19}{35} & -\frac{4}{35} \\ \frac{3}{7} & \frac{-1}{7} \end{bmatrix} mod\ 95 = [5 \quad 11]$$

Dari perkalian diatas didapatkan deret bilangan decimal yaitu

P = [6 1 7 10 7 13 4 11 7 9 6 11 5 0 14 15 5 2 5 5 5 1 6 15 5 4 4 11 7 14 6 11 6 14 5 7 5 4 4 11 7 9 6 15 5 6 5 11].

Table 11. Konversikan bilangan decimal tersebut kedalam bilangan biner 4 bit.

Desimal	Biner	Desimal	Biner	Desimal	Biner
6	0110	5	0101	6	0110
1	0001	2	0010	14	1110
7	0111	5	0101	5	0101
10	1010	5	0101	7	0111
7	0111	5	0101	5	0101
13	1101	1	0001	4	0100
4	0100	6	0110	4	0100
11	1011	15	1111	11	1011
7	0111	5	0101	7	0111
9	1001	4	0100	9	1001
6	0110	4	0100	6	0110
11	1011	11	1011	15	1111
5	0101	7	0111	5	0101
0	0000	14	1110	6	0110
14	1110	6	0110	5	0101
15	1111	11	1011	11	1011

Gabungkan 2 blok deret biner menjadi 1 blok sehingga terbentuk menjadi 8

bit biner, kemudian opersikan fungsi logika XOR dengan bilangan biner kunci k. Hasil operasi konversi biner ke decimal dapat dilihat dalam table berikut:

Tabel 12. Opersikan fungsi logika XOR dengan bilangan biner kunci k

Plainte xt	01100001	01111010	01111101	01001011	01111001
Kunci K	01110101	01110101	01110101	01110101	01110101
X-OR	**00010100**	**00001111**	**00001000**	**00111110**	**00001100**
Desimal	**20**	**15**	**8**	**62**	**12**
Plainte xt	01101011	01010000	11101111	01010010	01010101
Kunci K	01110101	01110101	01110101	01110101	01110101
X-OR	**00011110**	**00100101**	**00011010**	**00100111**	**00100000**
Desimal	**30**	**37**	**26**	**39**	**32**
Plainte xt	01001001	01101111	01010100	01001011	01111110
Kunci K	01110101	01110101	01110101	01110101	01110101
X-OR	**00100100**	**00011010**	**00100001**	**00111110**	**00001011**
Desimal	**36**	**26**	**33**	**62**	**11**
Plainte xt	01101011	01101110	01010111	01010100	01001011
Kunci K	01110101	01110101	01110101	01110101	01110101
X-OR	**00011110**	**00011011**	**00100010**	**00100001**	**00111110**
Desimal	**30**	**27**	**34**	**33**	**62**
Plainte xt	01111001	01101111	01010110	01011011	
Kunci K	01110101	01110101	01110101	01110101	
X-OR	**00001100**	**00011010**	**00100011**	**00101110**	
Desimal	**12**	**26**	**35**	**46**	

Dari tabel diatas didapatkan deret bilangan decimal 20 15 8 62 12 30 37 26 39 32 36 26 33 62 11 30 27 34 33 62 12 26 35 46]. Korespondesikan bilangan decimal dengan karakter sehingga didapat *plaintext* sebagai berikut : **UPI Melangkah Lebih Maju**. Ini merupakan kalimat deskripsi yang dapat kita mengerti setelah melalui ketiga tahapan

proses algoritma diatas.

Seringkali Kripnatalis matrik kunci dalam kriptografi bisa diketahui dengan mudah dengan menggunakan persamaan linier. Pada kesempatan ini penulis menggunakan matriks kunci persegi panjang dengan Plaintext yang terdiri dari 24 karakter. Kriptografi dengan menggunakan metode *Hill Chiper* yang dimodifikasi dengan bilangan biner dan disertai table untuk mempermudahkan agar tahapannya jelas. Bilangan biner disini menggunakan logika *X-OR* dan *X-NOR* dalam proses *Enskripsinya*. Tujuannya disini supaya Kripnatalis sulit menemukan persamaan linier untuk *Plaintext* dan matriks kuncinya. Operasi biner memungkinkan menghasilkan *chipertext* dengan karakter yang panjang. Dengan *ciphertext* yang lebih panjang diharapkan pesan yang disampaikan lebih tersamarkan.

SIMPULAN

Penelitian ini memakai kunci ordo persegi panjang dengan ordo 2 x 3 dimodifikasi dengan metode kriptograpi hill cipher. Metode ini lebih rumit dan Panjang. Metode ini memakai operasi biner X-OR dan XNOR.

DAFTAR RUJUKAN

Alawiyah, T. (2016). Modifikasi Kriptografi *Hill Cipher* Kunci Matriks Persegi Panjang menggunakan Fungsi Xor dan Fungsi Xnor. *Indonesia Journal on Computer and Information Technology*, *1*(1), 68–82.

Aribowo, D., Desmira, D., Ekawati, R., & Rahmah, N. (2021). SISTEM PERANCANGAN CONVEYOR MENGGUNAKAN SENSOR PROXIMITY PR18-8DN PADA WOOD SANDING MACHINE. *EDSUAINTEK: Jurnal Pendidikan, Sains Dan Teknologi*, *8*(1), 67–81. https://doi.org/10.47668/edusaintek.v8i1.146

Arif, M. H., & Fanani, A. Z. (2016). Kriptografi *Hill Cipher* dan Least Significant Bit untuk Keamanan Pesan pada Citra *Hill Cipher* and Least Significant Bit for Image Messaging Security. In *60 CSRID Journal* (Vol. 8, Issue 1).

Danny Wowor, A. (2013). MODIFIKASI KRIPTOGRAFI *HILL CIPHER* MENGGUNAKAN CONVERT BETWEEN BASE. In *Seminar Nasional Sistem Informasi Indonesia*.

Dwitiyanti, N., & Satria Setiawan, H. (2021). *APLIKASI OPERASI MATRIKS PADA PERANCANGAN SIMULASI METODE HILL CIPHER MENGGUNAKAN MICROSOFT EXCEL*.

Erdriani, D., Devita, D., Ilmu Komputer, F., Putra Indonesia, U., Raya Lubuk Begalung Nan, J. X., Lubuk Begalung, K., Padang, K., & Barat, S. (2021). APLIKASI MATRIK PADA ILMU KRIPTOGRAFI DENGAN MENGGUNAKAN MATLAB. *Jurnal Komtekinfo*, *8*(2), 154–162. https://doi.org/10.35134/komtekinfo.v7i4

Gunawan, I., Satria Tambunan, H., Irawan, E., Okta Kirana, I., Tunas Banga

Pematangsiantar, S., Jend Sudirman Blok, J. A., Pematangsiantar, K., & Utara, S. (2018). FUNGSI ALGORITMA KRIPTOGRAFI *HILL CIPHER* UNTUK PENGAMANAN FILE GAMBAR DAN PESAN TEKS. *TECHSI*, 118–128. https://doi.org/10.29103/techsi.v10i1.605

Gusti Awang Aritonang, O., Anwar, B., Taufik, F., Studi Mahasiswa, P., Triguna Dharma, S., & Studi Dosen Pembimbing, P. (2019). *Implementasi Kriptografi Menggunakan Metode HILL CIPHER Untuk Keamanan Data Gaji Karyawan Kasir Di PT. Matahari Department Store Plaza Medan Fair*.

Harefa, F. A., Syahril, M., Kusnasari, S., Stmik, S. I., & Dharma, T. (2020). Implementasi Security System Data Customer Pegadaian Dengan Penerapan Metode *Hill Cipher* Dan Xor. *Jurnal CyberTech*, *3*(3), 565–577. https://ojs.trigunadharma.ac.id/

Hasugian, A. H. (2013). IMPLEMENTASI ALGORITMA *HILL CIPHER* DALAM PENYANDIAN DATA. *Pelita Informatika Budi Darma*, 115–122. http://www.stmik-budidarma.ac.id//

Informasi, J., & Logika, K. (2021). Algoritma *Hill Cipher* Untuk Kebenaran Informasi pada Gambar dalam Media Sosial. *Jurnal Informasi Komputer Logika*, *2*(2).

Juliana Pangaribuan, L. (2018). *KRIPTOGRAFI HYBRIDA AGLORITMA HILL CIPHER DAN RIVEST SHAMIR ADLEMAN (RSA) SEBAGAI PENGEMBANGAN KRIPTOGRAFI KUNCI SIMETRIS (STUDI KASUS : NILAI MAHASISWA AMIK MBP)* (Vol. 7, Issue 1).

Makhomah, R., Santoso, K. A., & Kamsyakawuni, A. (2021). Pengkodean Teks Menggunakan Kombinasi *Hill Cipher* dan Operasi XOR. PRISMA, Prosiding Seminar Nasional Matematika. *PRISMA*, *4*, 548–552. https://journal.unnes.ac.id/sju/index.php/prisma/

Manaor, A., Pardede, H., Manurung, H., & Filina, D. (2017a). ALGORITMA VIGENERE CIPHER DAN *HILL CIPHER* DALAM APLIKASI KEAMANAN DATA PADA FILE DOKUMEN. *Jurnal Teknik Informatika Kaputama (JTIK)*, *1*(1), 26–33.

Manaor, A., Pardede, H., Manurung, H., & Filina, D. (2017b). ALGORITMA VIGENERE CIPHER DAN *HILL CIPHER* DALAM APLIKASI KEAMANAN DATA PADA FILE DOKUMEN. *Jurnal Teknik Informatika Kaputama (JTIK)*, *1*(1).

Puji Widodo, A., Adi Sarwoko, E., Suharto, E., & Freddy Orlando Siahaan, J. (2016). PENGAMANAN DATA FOTO PADA PERANGKAT OS ANDROID MENGGUNAKAN TEKNIK KRIPTOGRAFI HILL CIPHER. In *JISKa* (Vol. 1, Issue 2).

Puspita, N. Prima. B. N. (2010). *Kriptografi Hill Cipher Menggunakan Matriks.*

Putera, A., & Siahaan, U. (2016). *ALGORITMA GENETIKA UNTUK PEMBENTUKAN KUNCI MATRIKS 3 X 3 PADA KRIPTOGRAFI HILL CIPHER*.

Warni Hasibuan, Y., & Budhiati Veronica, R. (2022). *Perancangan dan Implementasi Aplikasi Kriptografi Algoritma Hill Cipher dalam Dekripsi Enkripsi Data Keuangan Nasabah Bank Sampoerna Menggunakan Kode ASCII*. http://journal.unnes.ac.id/sju/index.php/ujm

Yasmin, W., Teknik, A., Stmik, I., Karawang, K., Teknik, S., & Teknik, Y. Y. (2017). KRIPTANALISIS *HILL CIPHER* TERHADAP KNOWN PLAINTEXT ATTACK MENGGUNAKAN METODE DETERMINAN MATRIKS BERBASIS ANDROID. *Jurnal SIMETRIS*, *8*.