Volume 20 Nomor 1, Maret 2020 ISSN 1411-6669

MIMS

MAJALAH ILMIAH

Matematika dan Statistika

DITERBITKAN OLEH:
JURUSAN MATEMATIKA
FMIPA - UNIVERSITAS JEMBER

MAJALAH ILMIAH
Matematika dan Statistika

Alamat Redaksi:
Jurusan Matematika FMIPA – Universitas Jember
Jalan Kalimantan No 37 Kampus Tegalboto Jember 68121
Telp. : (0331) 334293
E-mail: mims.fmipa@unej.ac.id
Website: https://jurnal.unej.ac.id/index.php/MIMS/index

Diterbitkan oleh : Jurusan Matematika – FMIPA Universitas Jember.
Tahun pertama terbit : Oktober 2000
Jumlah terbit : Dua kali setahun pada bulan Maret dan September
Gambar cover depan : rancang bangun geometri, iterasi dan regresi

Majalah Ilmiah Matematika dan Statistika	Volume 20 Nomor 1	Halaman: 1 – 44	Jember, Maret 2020	ISSN 1411-6669

MAJALAH ILMIAH
Matematika dan Statistika

Volume 20 Nomor 1, Maret 2020 **ISSN 1411-6669**

Daftar Isi

Majalah Ilmiah Matematika dan Statistika Volume 20 Nomor 1, 2020, 15 – 26
https://jurnal.unej.ac.id/index.php/MIMS/index ISSN 1411-6669

VIGENERE CIPHER DENGAN MODIFIKASI *PLAINTEXT*

(*Vigenere Cipher Using Plaintext Modification*)

Dwi Rahmasari Kinasih Gusti, Kiswara Agung Santoso, Ahmad Kamsyakawuni

Jurusan Matematika, Fakultas MIPA, Universitas Jember
Jl. Kalimantan 37 Jember 68121, Indonesia
Email: dwi.rahmasarikg@gmail.com, {kiswara, kamsyakawuni}.fmipa@unej.ac.id

Abstract. Cryptography is knowledge of encoding messages by observe to security aspects. Cryptography uses two types of keys, namely symmetric keys and asymmetric keys. Vigenere cipher is a technique to encrypt messages by symmetric key. Vigenere cipher can be combined by several patterns and ASCII code. The pattern used can vary as long as the text can be returned to original message (can be decrypted). On this paper, we will modified plaintext before encrypt using vigenere cipher. The way to modified the plaintext are flip and shift rows of bit. The effect of the algorithm changes can be seen based on the renewal value obtained. If the correlation value gets smaller, it means the algorithm is better. The results of this study is the correlation value using vigenere cipher with modified plaintext is better compared to vigenere cipher with original plaintext.

Keywords: ASCII, Patterned, Bits, Vigenere Cipher
MSC 2010: 94A60

1. Pendahuluan

Perkembangan zaman yang semakin maju membuat teknologi informasi dan komunikasi semakin berkembang. Perkembangan tersebut perlu diimbangai dengan menjaga tingkat keamanannya agar informasi tetap aman. Pengkodean pesan merupakan salah satu cara untuk mengamankan informasi pesan. Teknik yang dapat digunakan yaitu dengan menggunakan teknik enkripsi dan dekripsi. Teknik tersebut berfungsi untuk membuat pesan tidak dapat dimengerti oleh pihak lain, selain pengirim dan penerima pesan. Bidang ilmu yang menerapkan penggunaan teknik tersebut adalah kriptografi. Kriptografi merupakan ilmu dan seni untuk menjaga kerahasiaan sebuah pesan dengan cara mengkodekannya ke dalam bentuk yang tidak dapat dipahami lagi maknanya [1]. Kriptografi menggunakan dua teknik yaitu teknik enkripsi dan dekripsi. Sebuah kunci diperlukan untuk dapat melakukan proses teknik tersebut. Kunci yang digunakan bisa berupa kunci simetris yaitu kunci yang penggunaannya untuk kedua proses sama ataupun kunci asimetris yang penggunaan kunci pada kedua proses berbeda. *Vigenere cipher* merupakan salah satu algoritma yang menggunakan kunci simetris dan dapat digunakan untuk mengkodekan pesan.

Vigenere cipher merupakan teknik menyandikan pesan dengan sandi Caesar menggunakan karakter pada kunci yang digunakan. Kunci yang digunakan pada algoritma ini berupa kunci simetris dan karakter pada kunci tersebut akan dipakai berulang apabila karakter pada pesan belum terproses semua. Karakter kunci yang berulang dapat membuat pesan mudah ditemukan. Oleh karena itu, tingkat keamanan *vigenere cipher* masih kurang optimal.

Modifikasi pada *vigenere cipher* dapat dilakukan untuk membuat tingkat keamanannya meningkat [3]. Tingkat keamanan yang semakin baik dapat dilihat berdasarkan korelasi antara pesan asli dan pesan yang telah bersandi semakin tidak linier. Penelitian oleh [5] yaitu "Penggabungan *Vigenere Cipher* dengan *Hill Cipher* pada Pengkodean *Plaintext* dengan Kunci Bertahap". Penelitian tersebut membahas tentang *vigenere cipher* yang digabung dengan *hill cipher* menggunakan kunci bertahap untuk proses pengkodean pesan teks. [4] melakukan penelitian "Efektivitas Pesan Teks dengan *Cipher* Substitusi, *Vigenere Cipher*, dan *Cipher* Transposisi". Penelitian tersebut melakukan proses enkripsi dan dekripsi secara bergantian dengan menggunakan ketiga algoritma tersebut untuk mendapatkan hasil yang paling baik. [2] melakukan penelitian berupa "Kombinasi *Vigenere Cipher* dan *Polyalphabet Cipher* pada Pengamanan *File Text*". Penelitian tersebut menggabungkan *vigenere cipher* dengan *polyalphabet cipher* dengan menggunakan tiga kunci pada prosesnya.

Pada penelitian ini, penulis memodifikasi pengkodean *vigenere cipher* dengan menggunakan perubahan cara baca biner berpola pada *plaintext* atau pesan. *Plaintext* akan dibuat per blok berisi 5 karakter dan setiap karakter *plaintext* tersebut diubah ke dalam bentuk biner ASCII. Setiap karakter pada *plaintext* yang telah berubah menjadi biner tersebut akan ditulis per baris tiap karakternya. Selanjutnya, *plaintext* tersebut akan dibaca binernya sesuai pola yang telah dibuat dan menghasilkan *plaintext* baru. *Plaintext* yang baru tersebut kemudian diproses menggunakan *vigenere cipher*. Penggunaan cara baca berpola dapat memberikan beberapa hasil *plaintext* yang berbeda sehingga pola akan sulit didapatkan. Selain itu, jika dilakukan kriptanalisis untuk memperoleh *plaintext*, maka *plaintext* yang didapatkan masih sampai pada tahap proses pergantian *plaintext*, sehingga *plaintext* yang asli masih aman.

2. Metodologi

Data yang digunakan dalam penelitian ini adalah pesan teks. Pesan teks yang dapat digunakan berupa abjad, angka, maupun simbol pada ASCII *printable characters*. Pesan akan diubah menjadi bentuk biner ASCII terlebih dahulu sebelum diproses. Hal tersebut untuk mengubah hasil *plaintext*-nya yang akan digunakan pada proses enkripsi. metode pada penelitian ini adalah sebagai berikut.

a. Pemeriksaan *Plaintext*

Pada tahap ini, dilakukan pembagian *plaintext* ke dalam blok yang telah ditentukan, yaitu tiap blok berisi 5 karakter dengan tiap karakter disusun per barisnya. Pembagian blok berisi 5 karakter tersebut karena disesuaikan dengan pola yang akan digunakan yaitu *shift row*. Apabila *plaintext* kurang dari blok maka *plaintext* akan ditambah dengan spasi hingga jumlah *plaintext* sesuai dengan blok.

b. Pergantian *Plaintext*

Plaintext akan dikonversi menjadi biner 7 bit sesuai dengan kode ASCII. Setelah itu, *plaintext* dalam bentuk biner kode ASCII tersebut akan dibaca dengan dua macam pola. Pola pertama yaitu membaca biner *plaintext* dengan membaca 2 digit pertama biner tetap dan 5 digit belakang dibaca secara flip. Pola kedua membaca *plaintext* dengan 2 digit pertama biner tetap dan 5 digit belakang dibaca secara *shift row*, setelah itu pola tersebut akan dibaca lagi dengan membaca baris mulai dari baris terbawah. Kedua pola tersebut akan digunakan secara berulang sampai *plaintext* terakhir dengan urutan menggunakan pola pertama kemudian pola kedua, lalu kembali ke pola pertama lagi apabila *plaintext* belum berakhir. Kedua pola tersebut harus menuliskan 2 digit pertamanya selalu tetap karena untuk menghindari nilai biner yang kurang dari 32.

c. Penyesuaian Kunci

Kunci yang digunakan pada pengkodean ini seperti kunci *vigenere cipher* pada umumnya. Apabila kunci kurang dari *plaintext* maka kunci akan diulang hingga seluruh *plaintext* tersebut terpenuhi. Apabila kunci melebihi *plaintext*, maka kunci akan ditulis sesuai dengan panjang *plaintext* yang diperlukan.

d. Proses Enkripsi

Proses enkripsi *plaintext* yang digunakan yaitu *plaintext* yang telah diubah karakternya. *Plaintext* akan dienkripsi dengan kunci sesuai dengan Persamaan (1). Rumus yang digunakan mengalami modifikasi berupa adanya pengurangan dan penambahan 32. Hal tersebut dilakukan untuk membuat karakter yang digunakan berada pada rentang nilai 32 – 126 sesuai pada ASCII *printable characters*. Selain itu, pengurangan 32 harus dilakukan agar *ciphertext* dapat didekripsi.

$$C_i = ((P_i^* - 32 + K_i) \bmod 95) + 32 \quad (1)$$

dengan P_i^* merupakan *plaintext* baru hasil perubahan pola biner.

e. Proses Dekripsi

Ciphertext akan diproses dengan kunci sesuai *vigenere cipher* dengan menggunakan Persamaan (2). Rumus dekripsi pada algoritma juga mengalami modifikasi seperti pada proses enkripsi yaitu berupa pengurangan dan penambahan 32.

$$P_i = ((C_i - 32 - K_i) \bmod 95) + 32 \quad (2)$$

f. Pergantian *Plaintext*

Proses dekripsi akan mengkasilkan *plaintext* namun belum menjadi *plaintext* yang sesungguhnya. Proses pergantian *plaintext* yaitu dengan mengubahnya menjadi biner kode ASCII dan dituliskan per baris untuk tiap karakternya. Setelah itu, membaca

biner *plaintext* dengan menggunakan dua pola seperti pada proses pergantian *plaintext* dienkripsi hingga mendapatkan *plaintext* sesungguhnya.

g. Koefisien Korelasi

Koefisien korelasi dari *plaintext* dan *ciphertext* digunakan untuk mengetahui hubungan linier dari keduanya. Hal tersebut untuk mengetahui tingkat keamanan terhadap serangan statistik, sehingga *plaintext* dan *ciphertext* harus memiliki tingkat perbedaan yang tinggi. Rumus untuk menghitung koefisien korelasi dapat dilihat pada Persamaan (3).

$$KK(p,c) = \frac{\sum_{i=1}^{n}(p_i-\mu(p))(c_i-\mu(c))}{\sigma(p)\sigma(c)} \quad (3)$$

Perhitungan rata-rata pada *plaintext* dan *ciphertext* dapat dilihat pada Persamaan (4) serta perhitungan untuk standar deviasi terdapat pada Persamaan (5).

$$\mu(p) = \frac{1}{n}\sum_{i=1}^{n} p_i \text{ dan } \mu(c) = \frac{1}{n}\sum_{i=1}^{n} c_i \quad (4)$$

$$\sigma(p) = \sqrt{\sum_{i=1}^{n}(p_i-\mu(p))^2} \text{ dan } \sigma(c) = \sqrt{\sum_{i=1}^{n}(c_i-\mu(c))^2} \quad (5)$$

3. Hasil dan Pembahasan

Plaintext Password SISTER EN5*gj5c dapat dilakukan proses enkripsi dan dekripsi seperti berikut.

A. Enkripsi

a. Pemeriksaan *Plaintext*

Plaintext pada tahap ini akan dikelompokkan per blok berisi 5 karakter. Apabila karakter *plaintext* pada blok terakhir kurang, maka akan ditambah karakter spasi pada blok tersebut. *Plaintext Password* SISTER EN5*gj5c akan terbagi menjadi beberapa blok berisi 5 karakter seperti berikut ini:

Blok 1 : Passw
Blok 2 : ord*(SP)*S
Blok 3 : ISTER
Blok 4 : *(SP)*EN5*
Blok 5 : gj5c*(SP)*

Blok 5 hanya berisi 4 karakter, sehingga perlu ditambah karakter spasi untuk melengkapi blok tersebut. *Plaintext* yang akan digunakan menjadi *Password* SISTER EN5*gj5c*(SP)*.

b. Pergantian *Plaintext*

Pada tahap ini, *plaintext* akan dikonversi menjadi bentuk biner ASCII 7 bit. Setelah mengubah menjadi bentuk biner, *plaintext* akan diubah sesuai pola yang telah ditentukan, yaitu menggunakan pola 1 berupa flip dan pola 2 berupa *shift rows* yang kemudian dibaca mulai baris terbawah. Pola-pola tersebut dapat dilihat seperti pada Gambar 1 – 5.

Kar	Bit ke-						
	1	2	3	4	5	6	7
P	**1**	**0**	1	0	0	0	0
a	**1**	**1**	0	0	0	0	1
s	**1**	**1**	1	0	0	1	1
s	**1**	**1**	1	0	0	1	1
w	**1**	**1**	1	0	1	1	1

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
A	**1**	**0**	0	0	0	0	1
P	**1**	**1**	1	0	0	0	0
Y	**1**	**1**	1	1	0	0	1
Y	**1**	**1**	1	1	0	0	1
}	**1**	**1**	1	1	1	0	1

Gambar 1. Pergantian *plaintext* blok 1 enkripsi

Kar	Bit ke-						
	1	2	3	4	5	6	7
o	**1**	**1**	0	1	1	1	1
r	**1**	**1**	1	0	0	1	0
d	**1**	**1**	0	0	1	0	0
(SP)	**0**	**1**	0	0	0	0	0
S	**1**	**0**	1	0	0	1	1

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
O	**1**	**1**	0	1	1	1	1
E	**1**	**1**	0	0	1	0	1
P	**1**	**1**	1	0	0	0	0
(SP)	**0**	**1**	0	0	0	0	0
Y	**1**	**0**	1	1	0	0	1

↓

Kar	Bit ke-						
	1	2	3	4	5	6	7
Y	1	0	1	1	0	0	1
(SP)	0	1	0	0	0	0	0
P	1	1	1	0	0	0	0
E	1	1	0	0	1	0	1
O	1	1	0	1	1	1	1

Gambar 2. Pergantian *plaintext* blok 2 enkripsi

Kar	Bit ke-						
	1	2	3	4	5	6	7
I	**1**	**0**	0	1	0	0	1
S	**1**	**0**	1	0	0	1	1
T	**1**	**0**	1	0	1	0	0
E	**1**	**0**	0	0	1	0	1
R	**1**	**0**	1	0	0	1	0

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
R	**1**	**0**	1	0	0	1	0
Y	**1**	**0**	1	1	0	0	1
E	**1**	**0**	0	0	1	0	1
T	**1**	**0**	1	0	1	0	0
I	**1**	**0**	0	1	0	0	1

Gambar 3. Pergantian *plaintext* blok 3 enkripsi

Kar	Bit ke-						
	1	2	3	4	5	6	7
(SP)	**0**	**1**	0	0	0	0	0
E	**1**	**0**	0	0	1	0	1
N	**1**	**0**	0	1	1	1	0
5	**0**	**1**	1	0	1	0	1
*	**0**	**1**	0	1	0	1	0

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
(SP)	**0**	**1**	0	0	0	0	0
J	**1**	**0**	0	1	0	1	0
Y	**1**	**0**	1	1	0	0	1
-	**0**	**1**	0	1	1	0	1
%	**0**	**1**	0	0	1	0	1

↓

Kar	Bit ke-						
	1	2	3	4	5	6	7
%	0	1	0	0	1	0	1
-	0	1	0	1	1	0	1
Y	1	0	1	1	0	0	1
J	1	0	0	1	0	1	0
(SP)	0	1	0	0	0	0	0

Gambar 4. Pergantian *plaintext* blok 4 enkripsi

Kar	Bit ke-						
	1	2	3	4	5	6	7
g	**1**	**1**	0	0	1	1	1
j	**1**	**1**	0	1	0	1	0
5	**0**	**1**	1	0	1	0	1
c	**1**	**1**	0	0	0	1	1
(SP)	**0**	**1**	0	0	0	0	0

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
|	**1**	**1**	1	1	1	0	0
J	**1**	**1**	0	1	0	1	0
5	**0**	**1**	1	0	1	0	1
X	**1**	**1**	1	1	0	0	0
(SP)	**0**	**1**	0	0	0	0	0

Gambar 5. Pergantian *plaintext* blok 5 enkripsi

Plaintext yang telah diproses dengan pola menghasilkan *plaintext* baru. *Plaintext* baru tersebut yaitu Apyy}Y*(SP)*peoRYETI%-YJ*(SP)*|j5x*(SP)* dan akan digunakan pada proses enkripsi *vigenere cipher.*

c. Penyesuaian Kunci

Kunci yang digunakan pada tahap ini dapat menggunakan berbagai karakter pada ASCII *printable characes*, sehingga tidak terbatas pada abjad saja. Apabila kunci yang digunakan memiliki panjang karakter yang kurang dari *plaintext* maka karakter pada kunci tersebut akan diulang sampai karakter *plaintext* terpenuhi semua. Kunci Kripto yang digunakan hanya memiliki panjang 6 karakter saja, sedangkan *plaintext* Apyy}y*(SP)*peoRYETI%-YJ*(SP)*|j5x*(SP)* memiliki panjang 25 karakter, sehingga kunci Kripto akan diulang-ulang penggunaannya menyesuaikan panjang dari *plaintext*.

d. Proses Enkripsi

Plaintext Apyy}Y*(SP)*peoRYETI%-YJ*(SP)*|j5x*(SP)* dan kunci Kripto akan

dienkripsi menggunakan *vigenere cipher* dengan modulo 95, karena menyesuaikan dengan jumlah karakter pada ASCII *printable charactes*. Berikut merupakan proses enkripsinya:

(A – 32 + K mod 95) + 32 = 45 = –
(p – 32 + r mod 95) + 32 = 36 = $
⋮
(x – 32 + o mod 95) + 32 = 41 =)
(*(SP)* – 32 + K mod 95) + 32 = 107 = k

Setelah proses enkripsi selesai, didapatkan *ciphertext* -$$+3ik$o!gi1gS6Bi63'{J)k.

B. Dekripsi

a. Proses Dekripsi

Proses dekripsi dilakukan dengan menggunakan *ciphertext* dan kunci yang diberikan, kemudian diproses menggunakan algoritma yang sesuai. *Ciphertext* -$$+3ik$o!gi1gS6Bi63'{J)k dan kunci Kripto akan didekripsi menggunakan *vigenere cipher*. Berikut merupakan proses dekripsinya:

(– - 32 – K mod 95) + 32 = 65 = A
($ – 32 – r mod 95) + 32 = 112 = p
⋮
() – 32 – o mod 95) + 32 = 120 = x
(k – 32 – K mod 95) + 32 = 32 = *(SP)*

Proses dekripsi menghasilkan *plaintext* Apyy}Y*(SP)*peoRYETI%-YJ*(SP)*|j5x*(SP)*. *Plaintext* tersebut akan digunakan untuk mencari *plaintext* yang sesungguhnya menggunakan pola yang telah ditetapkan.

b. Pergantian *Plaintext*

Plaintext Apyy}Y*(SP)*peoRYETI%-YJ*(SP)*|j5x*(SP)* akan diproses menggunakan pola 1 dan 2 agar didapatkan *plaintext* asli. Berikut proses perubahan *plaintext* menggunakan pola dapat dilihat pada Gambar 6 – 10.

Kar	Bit ke-						
	1	2	3	4	5	6	7
P	**1**	**0**	1	0	0	0	0
a	**1**	**1**	0	0	0	0	1
s	**1**	**1**	1	0	0	1	1
s	**1**	**1**	1	0	0	1	1
w	**1**	**1**	1	0	1	1	1

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
P	**1**	**0**	1	0	0	0	0
a	**1**	**1**	0	0	0	0	1
s	**1**	**1**	1	0	0	1	1
s	**1**	**1**	1	0	0	1	1
w	**1**	**1**	1	0	1	1	1

Gambar 6. Pergantian *plaintext* blok 1 dekripsi

Kar	Bit ke-						
	1	2	3	4	5	6	7
Y	**1**	**0**	1	1	0	0	1
(SP)	**0**	**1**	0	0	0	0	0
P	**1**	**1**	1	0	0	0	0
E	**1**	**1**	0	0	1	0	1
O	**1**	**1**	0	1	1	1	1

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
S	**1**	**0**	1	0	0	1	1
(SP)	**0**	**1**	0	0	0	0	0
d	**1**	**1**	0	0	1	0	0
r	**1**	**1**	1	0	0	1	0
o	**1**	**1**	0	1	1	1	1

↓

Kar	Bit ke-						
	1	2	3	4	5	6	7
o	1	1	0	1	1	1	1
r	1	1	1	0	0	1	0
d	1	1	0	0	1	0	0
(SP)	0	1	0	0	0	0	0
S	1	0	1	0	0	1	1

Gambar 7. Pergantian *plaintext* blok 2 dekripsi

Kar	Bit ke-						
	1	2	3	4	5	6	7
R	**1**	**0**	1	0	0	1	0
Y	**1**	**0**	1	1	0	0	1
E	**1**	**0**	0	0	1	0	1
T	**1**	**0**	1	0	1	0	0
I	**1**	**0**	0	1	0	0	1

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
I	**1**	**0**	0	1	0	0	1
S	**1**	**0**	1	0	0	1	1
T	**1**	**0**	1	0	1	0	0
E	**1**	**0**	0	0	1	0	1
R	**1**	**0**	1	0	0	1	0

Gambar 8. Pergantian *plaintext* blok 3 dekripsi

Kar	Bit ke-						
	1	2	3	4	5	6	7
%	**0**	**1**	0	0	1	0	1
-	**0**	**1**	0	1	1	0	1
Y	**1**	**0**	1	1	0	0	1
J	**1**	**0**	0	1	0	1	0
(SP)	**0**	**1**	0	0	0	0	0

→

Kar	Bit ke-						
	1	2	3	4	5	6	7
*	**0**	**1**	0	1	0	1	0
5	**0**	**1**	1	0	1	0	1
N	**1**	**0**	0	1	1	1	0
E	**1**	**0**	0	0	1	0	1
(SP)	**0**	**1**	0	0	0	0	0

↓

Kar	Bit ke-						
	1	2	3	4	5	6	7
(SP)	0	1	0	0	0	0	0
E	1	0	0	0	1	0	1
N	1	0	0	1	1	1	0
5	0	1	1	0	1	0	1
*	0	1	0	1	0	1	0

Gambar 9. Pergantian *plaintext* blok 4 dekripsi

Kar	Bit ke-								Kar	Bit ke-						
	1	2	3	4	5	6	7			1	2	3	4	5	6	7
|	**1**	**1**	1	1	1	0	0		G	**1**	**1**	0	0	1	1	1
J	**1**	**1**	0	1	0	1	0	→	J	**1**	**1**	0	1	0	1	0
5	**0**	**1**	1	0	1	0	1		5	**0**	**1**	1	0	1	0	1
X	**1**	**1**	1	1	0	0	0		C	**1**	**1**	0	0	0	1	1
(SP)	**0**	**1**	0	0	0	0	0		*(SP)*	**0**	**1**	0	0	0	0	0

Gambar 10. Pergantian *plaintext* blok 5 dekripsi

Proses pergantian pola tersebut menghasilkan plaintext yang sesungguhnya yaitu *Password* SISTER EN5*gj5c**.**

C. Koefisien Korelasi

Koefisien korelasi digunakan untuk membandingkan tingkat keamanan dari algoritma yang digunakan. Uji ini dilakukan dengan menghitung nilai rata-rata dan standar deviasi pada *plaintext* dan *ciphertext*. Karakter pada *plaintext* dan *ciphertext* akan diubah menjadi bentuk desimal terlebih dahulu. Setelah karakter dikonversi ke desimal, maka langkah selanjutnya adalah menghitung koefisien korelasi. Berikut merupakan perhitungan korelasi *vigenere cipher* modifikasi *plaintext*.

$$\mu(p) = \frac{1}{n}\sum_{i=1}^{n} p_i$$
$$= \frac{80+97+115+\cdots+32}{25}$$
$$= 80{,}96$$

$$\mu(c) = \frac{1}{n}\sum_{i=1}^{n} c_i$$
$$= \frac{45+36+36+\cdots+107}{25}$$
$$= 70{,}4$$

$$\sigma(p) = \sqrt{\sum_{i=1}^{n}(p_i - \mu(p))^2}$$
$$= \sqrt{(80 - 80{,}96)^2 + \cdots + (32 - 80{,}96)^2}$$
$$= 135{,}893$$

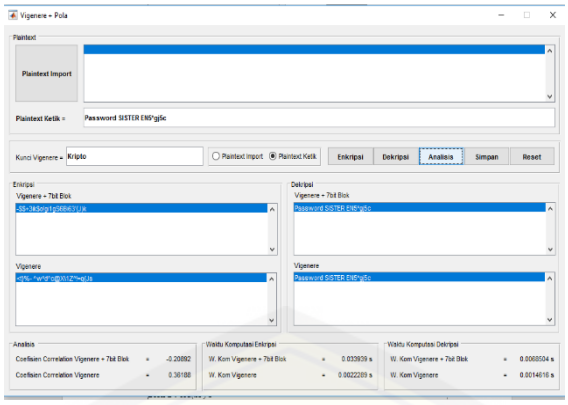
$$\sigma(c) = \sqrt{\sum_{i=1}^{n}(c_i - \mu(c))^2}$$
$$= \sqrt{(45 - 70{,}4)^2 + \cdots + (107 - 70{,}4)^2}$$
$$= 151{,}479$$

$$KK(p,c) = \frac{\sum_{i=1}^{n}(p_i - \mu(p))(c_i - \mu(c))}{\sigma(p)\sigma(c)}$$
$$= \frac{(80 - 80{,}96)(45 - 70{,}4) + \cdots + (32 - 80{,}96)(107 - 70{,}4)}{(135{,}893)(151{,}479)}$$
$$= \frac{24{,}384 - 551{,}776 - \cdots - 1791{,}936}{20584{,}935}$$
$$= \frac{-4300{,}6}{20584{,}935}$$
$$= -0{,}20892$$

Nilai koefisien korelasinya didapat -0,20892 yang nantinya akan dibandingkan dengan nilai korelasi algoritma yang lain.

D. Aplikasi Program

Program dibuat menggunakan Matlab 2015b dengan desain program menggunakan GUI. Gambar 11 merupakan tampilan GUI untuk pengkodean pesan menggunakan *vigenere cipher* modifikasi *plaintext* dan *vigenere cipher*.



Gambar 11. Aplikasi program algoritma

Analisis keamanan pada penelitian ini menggunakan analisis koefisien korelasi dan perhitungan waktu komputasi enkripsi dan dekripsi. Ketiga analisis tersebut akan dijadikan sebagai tolak ukur keamanan algoritma pada penelitian ini. Percobaan penelitian akan menggunakan *plaintext* yang sama dengan beberapa jenis kunci yang berbeda. *Plaintext* yang digunakan yaitu Sister234*. Percobaan dilakukan sebanyak 4 kali dan dapat dilihat pada Tabel 1 – 4.

Tabel 1. Percobaan pertama (Kunci : la<0)

Perbandingan	*Vigenere Cipher* Modifikasi *Plaintext*	*Vigenere Cipher*
Ciphertext	fTV6"fa^2T	`KPErTncAk
Koefisien Korelasi	-0,14454	-0,39937
Waktu Enkripsi	$0{,}28263 \times 10^{-1}$ detik	$0{,}1263 \times 10^{-2}$ detik
Waktu Dekripsi	$0{,}6133 \times 10^{-1}$ detik	$0{,}10937 \times 10^{-1}$ detik

Tabel 2. Percobaan kedua (Kunci : jA;2b)

Perbandingan	*Vigenere Cipher* Modifikasi *Plaintext*	*Vigenere Cipher*
Ciphertext	dTU8w0fiWu	^KOGh}snf-
Koefisien Korelasi	-0,61969	0,0056176
Waktu Enkripsi	$0{,}2104 \times 10^{-2}$ detik	$0{,}92369 \times 10^{-3}$ detik
Waktu Dekripsi	$0{,}22911 \times 10^{-2}$ detik	$0{,}1531 \times 10^{-2}$ detik

Tabel 3. Percobaan ketiga (Kunci: fE@47_Qh)

Perbandingan	*Vigenere Cipher* Modifikasi *Plaintext*	*Vigenere Cipher*
Ciphertext	`XZ:L%v7,X	ZOTI=r$<;o
Koefisien Korelasi	-0,18221	0,30349
Waktu Enkripsi	$0{,}13259 \times 10^{-1}$ detik	$0{,}83495 \times 10^{-3}$ detik
Waktu Dekripsi	$0{,}63801 \times 10^{-2}$ detik	$0{,}65142 \times 10^{-3}$ detik

Tabel 4. Percobaan keempat (Kunci : kR%0tO.0fI)

Perbandingan	*Vigenere Cipher* Modifikasi *Plaintext*	*Vigenere Cipher*
Ciphertext	ee?6*tS^,\	_\9Ezb`c;s
Koefisien Korelasi	-0,05893	-0,23344
Waktu Enkripsi	$0{,}43818 \times 10^{-2}$ detik	$0{,}66349 \times 10^{-3}$ detik
Waktu Dekripsi	0,12949 detik	$0{,}42411 \times 10^{-2}$ detik

Percobaan dilakukan menggunakan panjang karakter kunci yang berbeda-beda. Panjang karakter kunci yang digunakan yaitu panjang karakter kunci setengah dari panjang karakter *plaintext*, panjang karakter kunci lebih sedikit dibanding panjang karakter *plaintext* serta panjang karakter kunci melebihi panjang karakter *plaintext*. Sebanyak 40 percobaan dengan masing-masing percobaan dilakukan 10 kali pada panjang karakter kunci 4, 5, 8, dan 11 menghasilkan nilai koefisien korelasi yang beragam. Nilai korelasi tersebut ada yang bernilai lebih baik maupun lebih buruk menggunakan *vigenere cipher* modifikasi *plaintext* pada semua kategori panjang kunci yang digunakan. Percobaan masing-masing 10 kali menggunakan panjang karakter kunci 4, 8, dan 11 menghasilkan masing-masing 7 kali *vigenere cipher* modifikasi *plaintext* lebih baik korelasinya, sedangkan percobaan 10 kali menggunakan panjang karakter kunci 5 menghasilkan 6 kali *vigenere cipher* modifikasi *plaintext* lebih baik korelasinya. 40 percobaan tersebut secara keseluruhan mendapatkan 27 percobaan dengan hasil korelasi lebih baik menggunakan *vigenere cipher* modifikasi *plaintext*. Nilai korelasi yang lebih mendekati 0 dikatakan lebih baik karena menunjukkan bahwa tingkat linieritasnya lebih rendah. Semakin panjang kunci yang digunakan bukan hanya penentu bahwa algoritma menjadi semakin baik. *Plaintext* dan kunci dapat saling mempengaruhi hasil dari koefisien korelasi karena perubahan satu karakter saja dapat membuat hasil korelasi menjadi berubah lebih baik ataupun lebih buruk. Waktu komputasi enkripsi dan dekripsi pada *vigenere cipher* modifikasi *plaintext* membutuhkan waktu yang lebih lama dibandingkan dengan *vigenere cipher*, karena semakin kompleks algoritma membutuhkan waktu proses yang lebih lama, namun perbedaan waktu kedua algoritma tersebut tidak sampai satu detik. *Vigenere cipher* modifikais *plaintext* masih memerlukan pergantian *plaintext* terlebih dahulu untuk dapat diproses sedangkan *vigenere cipher* langsung dapat diproses. *Ciphertext* yang dihasilkan algoritma menjadi lebih bervariasi karakternya meskipun menggunakan *plaintext* berupa abjad saja. Hal itu karena karakter yang digunakan pada algoritma menggunakan ASCII *printable charactes* yang membuat karakter menjadi semakin beragam. Secara representatif dengan menggunakan panjang karakter kunci yang berbeda-beda didapatkan bahwa *vigenere cipher* modifikasi *plaintext* lebih baik dibandingkan dengan *vigenere cipher*.

4. Kesimpulan

Percobaan yang telah dilakukan 40 kali dengan menggunakan perwakilan 4 jenis panjang kunci yang berbeda (4, 5, 8, dan 11) pada satu *plaintext* menghasilkan 27 percobaan memiliki tingkat korelasi yang lebih baik pada *vigenere cipher* modifikasi *plaintext* dibandingkan dengan *vigenere cipher* yang asli. Tingkat korelasi yang lebih baik ditunjukkan dengan hasil yang semakin mendekati nilai 0 yang berarti tingkat linieritas antara *plaintext* dan *ciphertext* rendah. *Plaintext* dan kunci saling mempengaruhi hasil dari nilai koefisien korelasi karena dengan melakukan perubahan satu karakter dan perubahan panjang kunci dapat membuat nilai koefisien korelasi berubah.

Daftar Pustaka

[1] Harini, R. T. dan Utami, E. (2012). Aplikasi Enkripsi SMS dengan Modifikasi *Vigenere Cipher* pada Ponsel Android. *Jurnal Data Manajemen dan Teknologi Informasi*. 13(2): 65-70.

[2] Mahmudah, M. (2018). Implementasi Kriptografi pada Teks Menggunakan Algoritma Tanam Padi dan Bajak Sawah. *Skripsi*. Surabaya: Fakultas Sains dan Teknologi Universitas Islam Negeri Sunan Ampel Surabaya.

[3] Maricar, M. A. dan Sastra, N. P. (2018). Efektivitas Pesan Teks dengan *Cipher* Substitusi, *Vigenere Cipher*, dan *Cipher* Transposisi. *Majalah Ilmiah Teknologi Elektro*. 17(1): 59-65.

[4] Rahmawati, R. (2017). Penggabungan *Vigenere Cipher* dengan *Hill Cipher* pada Pengkodean *Plaintext* dengan Kunci Bertahap. *Skripsi*. Jember: Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

[5] Tarigan, E., Maha, D. H. S. dan Ketaren, E. (2018). Kombinasi *Vigenere Cipher* dan *Polyalphabet Cipher* pada Pengamanan *File Text*. *Journal Article*. 3(1): 71-76.