Eksplorasi Kekuatan Kombinasi Dua Buah Teknik Cipher

Georgius Rinaldo Winata / 13509030
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
Georgius.rinaldo@students.itb.ac.id

***Abstrak*—Pada masa sekarang, sudah banyak sekali penggunaan teknik enkripsi untuk menjaga kerahasiaan dan keamanan. Proses enkripsi pun bermacam-macam dari teks, perangkat lunak, perangkat keras, dll. Tetapi seiring berjalannya waktu, untuk melakukan enkripsi sudah jarang sekali menggunakan algoritma yang klasik karena digantikan dengan cara yang lebih modern dan disesuaikan dengan spesifikasi objek yang akan dienkripsi. Pada makalah ini akan mencoba mengeksplorasi kembali kekuatan dari algoritma klasik. Akan tetapi, kekuatan algoritma klasik ini akan diperkuat dengan menggunakan kombinasi dari dua buah teknik enkripsi. Dengan menggunakan tolak ukur tertentu akan diketahui kombinasi teknik enkripsi yang mana yang terbaik.**

***Index Terms*—Cipher, Cipherteks, Plainteks, Kriptografi, Playfair Cipher, Vigenere Cipher, Hill Cipher, Affine Cipher**

I. PENDAHULUAN

Banyak teknik-teknik cipher yang tergolong dalam kriptografi klasik. Adapun tujuan dari kriptografi klasik ini adalah untuk menjaga kerahasiaan sebuah teks. Teknik-teknik ini banyak digunakan di zaman dahulu pada zaman perang. Namun teknik-teknik ini sudah menjadi *obsolete* karena telah terpecahkan juga pada masa itu. Seiring berkembangnya teknik-teknik enkripsi, teknik ini dianggap sederhana dan jarang digunakan karena tergantikan dengan kriptografi modern. Akan tetapi algorima kriptografi klasik masih bisa digunakan dan diterapkan pada teks sederhana yang masih belum mengggunakan media yang lebih modern seperti komputer atau alat elektronik lainnya.

Dalam teknik cipher terdiri dari sebuah plainteks, kunci, dan cipherteks sebagai hasilnya. Plainteks adalah sebuah teks atau tulisan yang ingin dijaga kerahasiaannya yang akan diolah nantinya dengan menggunakan kunci yang ditentukan sehingga menghasilkan sebuah cipherteks. Hasil cipherteks ini yang akan disebarkan dan telah dianggap terjaga kerahasiaannya. Kekuatan sebuah cipher bisa dilihat dengan tolak ukur tertentu misalnya dengan menghitung banyaknya perulangan kata, jarak kata yang berulang, dll. Pada makalah ini akan dieksplorasi kekuatan dari beberapa teknik cipher yang tergolong dalam algoritma kriptografi klasik dengan cara dikombinasikan terlebih dahulu dengan harapan menambah kekuatan dari teknik cipher hasil kombinasi itu.

II. TEORI DASAR

A. Vigenere Cipher

Vigenere Cipher termasuk dalam *cipher* abjad-majemuk (*polyalphabetic substitution cipher*). Vigenere Cipher dipublikasikan oleh diplomat (sekaligus seorang kriptologis) Perancis, Blaise de Vigenere pada abad 16 (tahun 1586). Tetapi sebenarnya teknik *cipher* ini pertama kali digambarkan oleh Giovan Batista Belaso dalam buku *La Cifra del Sig. Giovan Batista Belaso* pada tahun 1553. Algoritma ini baru dikenal luas 20 tahun kemudian setelah dikenalkan oleh Vigenere. Teknik *cipher* ini dipecahkan oleh Babage dan Kasiski pada pertengahan Abad 19.

Vigenere Cipher menggunakan bujursangkar vigenere untuk melakukan enkripsi dimana huruf pada plainteks dipotongkan posisinya dengan huruf dari kunci. Setiap baris di dalam bujursangkar menyatakan huruf-huruf cipherteks yang diperoleh dengan *Caesar Cipher*. Dalam pengisian tabel berukuran 26x26 yang dinotasikan dari a sampai z akan diisi pula dengan a sampai z dengan digeser satu per satu di tiap barisnya sehingga jika baris pertama dimulai dari a sampai z, maka baris kedua akan dimulai dari b sampai z dan kembali ke a, dst. Representasinya dari kolom dan baris masing-masing adalah untuk plainteks dan kunci yang dipotongkan untuk menghasilkan cipherteks.

Karakter Cipherteks: $c_1(p) = (p + k_i) \bmod 26$
c: cipherteks
p: plainteks
k: kunci

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Gambar 1. Tabel Vigenere Cipher

B. Playfair Cipher

Playfair Cipher termasuk dalam *polygram cipher*. Teknik ini ditemukan oleh Sir Charles Wheatstone namun dipromosikan oleh Baron Lyon Playfair pada tahun 1854. Hampir sama dengan Vigenere Cipher, teknik ini juga menggunakan tabel.

Teknik *cipher* ini mengenkripsi pasangan huruf (digram atau digraph), bukan huruf tunggal seperti pada *cipher* klasik lainnya. Tujuannya adalah untuk membuat analisis frekeunsi menjadi sangat sulit sebab frekuensi kemunculan huruf-huruf di dalam cipherteks menjadi datar (flat). Dengan menggunakan tabel yang dibentuk dari kunci, hasil cipherteks menjadi sulit dipecahkan.

Proses pembentukan cipherteks dari *playfair cipher* terdiri dari beberapa tahap. Pertama adalah dengan pemisahan plainteks menjadi digraf dan mengganti huruf j menjadi i pada plainteks. Jika pada plainteks ada karakter yang sama, maka sisipkan z di tengahnya. Penambahan alfabet z juga berlaku apabila plainteks yang telah terpisah ternyata ganjil. Kedua adalah pembentukan kunci. Pembentukan kunci dilakukan dengan menuliskan kunci secara berurut pada tabel berukuran 5x5 yang nantinya akan diperluas menjadi 6x6. Kunci ditulis tanpa ada huruf yang berulang diteruskan dengan semua huruf dalam alphabet sampai tabel penuh. Kemudian kolom dan baris terakhir yaitu keenam diisi dengan masing-masing kolom dan baris pertama.

Algoritma enkripsi dari playfair cipher adalah sebagai berikut:

1. Jika dua huruf terdapat pada baris kunci yang sama maka tiap huruf diganti dengan huruf di kanannya.
2. Jika dua huruf terdapat pada kolom kunci yang sama maka tiap huruf diganti dengan huruf di bawahnya.
3. Jika dua huruf tidak pada baris yang sama atau kolom yang sama, maka huruf pertama diganti dengan huruf pada perpotongan baris huruf pertama dengan kolom huruf kedua. Huruf kedua diganti dengan huruf pada titik sudut keempat dari persegi panjang yang dibentuk dari 3 huruf yang digunakan sampai sejauh ini.

Contoh: Kunci (yang sudah diperluas) ditulis kembali sebagai berikut:

S	T	A	N	D	S
E	R	C	H	B	E
K	F	G	I	L	K
M	O	P	Q	U	M
V	W	X	Y	Z	V
S	T	A	N	D	

Plainteks (dalam pasangan huruf):

GO OD BR OZ OM SZ SW EZ EP CL EA NZ

Cipherteks:

FP UT EC UW PO DV TV BV CM BG CS DY

Playfair cipher memiliki kelemahannya sendiri. Salah satu kelemahannya adalah *polygram* playfair cipher tidaklah cukup besar sehingga kurang aman karena hanya memakai dua huruf. Kelemahan lainnya adalah walaupun playfair cipher sulit dipecahkan dengan menggunakan analisis frekuensi relatif huruf-huruf, namun ia dapat dipecahkan dengan menggunakan analisis yang serupa yaitu dengan pasangan huruf.

C. Hill Cipher

Hill Cipher adalah teknik *cipher* yang menggunakan m buah persamaan linier. Teknik ini dikembangkan oleh Lester Hill pada tahun 1929. Dengan menggunakan kunci berbentuk matriks dengan ukuran MxM, plainteks sepanjang M juga akan dienkripsi secara terus menerus.

Proses dekrispi dari Hill Cipher perlu melakukan proses awal terlebih dahulu. Proses awal itu adalah dengan menghitung invers dari matriks kunci yang ada. Dengan mendapatkan matriks kunci, proses yang dilakukan sama seperti proses enkripsi hanya saja dengan menggunakan matriks kunci yang telah diinvers.
Contoh:

$$\mathbf{K} = \begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix}$$ sebagai kunci

Plainteks: PAYMOREMONEY
Enkripsi tiga huruf pertama:
PAY = (15, 0 ,24)
Cipherteks: C =

$$\begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix} \begin{pmatrix} 15 \\ 0 \\ 24 \end{pmatrix} = \begin{pmatrix} 375 \\ 819 \\ 486 \end{pmatrix} \bmod 26 = \begin{pmatrix} 11 \\ 13 \\ 18 \end{pmatrix}$$

= **LNS**

Proses ini dilakukan sampai semua plainteks terenkripsi dengan kunci yang disediakan.

D. Affine Cipher

Affine cipher adalah teknik *cipher* yang merupakan perluasan dari caesar cipher. Dengan menggunakan faktor pengali dan substitusi, didapatkan cipherteks hasil enkripsi. Caesar cipher adalah kasus khusus dari affine cipher dengan faktor pengali satu sehingga menghasilkan cipherteks yang hanya berupa penggeseran.

Enkripsi: $C = mP + b \pmod n$
Dekripsi: $P = m^{-1} (C-b) \pmod n$
Kunci: m dan b

Keterangan:
n: ukuran alfabet
m: bilangan bulat yang relatif prima dengan n
b: jumlah pergeseran
m^{-1}: inversi dari m (mod n)

Affine cipher tidak aman karena kunci mudah ditemukan. Salah satu caranya adalah dengan menggunakan exhaustive search. Hal ini disebabkan ada 25 pilihan untuk b dan 12 buah nilai m yang relatif prima dengan 26 (alfabet saja). Selain dengan *exhaustive search*, affine cipher juga mudah diserang menggunakan *known-plaintext attack.* Misalkan kriptanalis mempunyai dua buah plainteks P_1 dan P_2 yang berkoresponden dengan cipherteks C_1 dan C_2., maka m dan b mudah dihitung dari buah kekongruenan simultan berikut ini:

$C_1 = mP_1 + b \pmod n$
$C_2 = mP_2 + b \pmod n$

III. APLIKASI KOMBINASI CIPHER

Pada makalah ini, teknik *cipher* yang akan dikombinasi akan dibatasi pada tiga teknik saja, yaitu vigenere cipher, playfair cipher, dan hill cipher. Adapun alasan dipilihnya teknik ini adalah karena memiliki kelebihan dan kekurangannya tersendiri. Dengan kombinasi yang dibuat sedemikan rupa akan dieksplorasi tingkat kekuatannya. Kombinasi yang dilakukan juga diharapkan menghasilkan sesuatu yang menutupi kekurangannya dengan melakukan perimbangan-pertimbangan dari kelemahan yang ada terlebih dahulu. Kombinasi yang dilakukan diprioritaskan pada teknik *cipher* yang lebih lemah, bukan yang kuat.

Adapun tolak ukur yang akan digunakan untuk pengukuran dari kekuatan kombinasi cipher yang akan dibuat adalah sebagai berikut:

1. Jarak perulangan kata
2. Banyak digraf yang berulang
   - Keterangan: semua kombinasi akan diseimbangkan pada saat pengujian kekuatan, misalnya: semua dalam bentuk digraph
3. Rumus yang menghindari exhaustive search atau teknik penyerangan serupa

Sebenarnya dengan melakukan dua kali proses cipher pada plainteks yang akan dienkripsi sudahlah cukup menjaga kerahasiaan plainteks. Hal ini dikarenakan maksud yang ada dari cipherteks ketika berhasil didekripsi pertama kali masih kabur dalam arti tidak bermakna. Oleh karena itu menggunakan teknik dua kali cipher secara simultan sebenarnya cukup. Tetapi pada makalah ini dicoba mengkombinasikan cara ciphernya, bukan menggunakan dua kali cipher secara simultan.

A. Kombinasi untuk Mengurangi atau Mengaburkan Pengulangan pada Cipherteks

Pada playfair cipher dan Vigenere cipher memiliki kelemahan yang cukup serupa. Walaupun vigenere cipher memiliki tabel yang lebih besar, tapi kasusnya sama seperti playfair cipher. Kelemahannya yaitu terdapat frekuensi kemunculan dimana pada kasus vigenere adalah huruf dan kasus playfair adalah digraph. Oleh karena itu, untuk mengurangi terjadinya hal itu, maka akan digunakan kombinasi dengan teknik *cipher* lain dengan tujuan saling mengurangi pengulangan huruf atau digraf pada teknik cipher ini (playfair cipher dan vigenere cipher).

1. Kombinasi Vigenere Cipher dengan Affine Cipher

Tujuan utama dari kombinasi dua teknik cipher ini adalah untuk mengurangi dan mengaburkan jumlah frekuensi kemunculan huruf yang paling sering sehingga kuncinya sulit dicari dan plainteks terjaga kerahasiaannya. Pada kasus ini, teknik yang akan diperkuat berfokus pada vigenere cipher. Perubahan rumus yang akan diajukan cukup sederhana karena hanya untuk mengurangi dan mengaburkan jumlah frekuensi yang muncul. Dengan menggunakan kombinasi kedua teknik ini, maka walau misalnya pada plainteks terdapat banyak huruf "e" yang muncul akan tersamarkan dengan adanya bilangan random sebagai faktor pengali yang akan diajukan sebagai rumus di bawah ini.

Rumus yang diajukan:

$$C_1(P) = m(P + K_i) \bmod 26$$

Keterangan:
C = cipherteks
P = plainteks
K = kunci
m = suatu angka random yang terus berubah supaya kunci hasil cipherteks sulit diprediksi.

- Dalam kasus ini misalnya akan dpakai bilangan fibonacci saja.

Contoh:
Plainteks: **AKU CINTA KRIPTOGRAFI**
Kunci: **GANTENG**
Cipherteksnya:
$C = C_1 + C_2 + \ldots + C_n$
Dimana C adalah cipherteks penuh dan C_1 sampai C_n adalah karakternya

➔ Karakter dimulai dari 0

$C_1 = 1(0+6) \bmod 26 = 6 = g$
$C_2 = 1(11+0) \bmod 26 = 10 = i$
$C_3 = 2(20+13) \bmod 26 = 14 = m$
dst

2. Kombinasi Playfair Cipher dengan Hill Cipher

Pada kombinasi kali ini, akan digunakan playfair cipher sebagai dasarnya. Seperti yang diketahui, playfair cipher memiliki kelemahan karena adanya perulangan kata dan tabelnya yang kecil. Dengan hal ini mengakibatkan playfair cipher cenderung mudah untuk didekripsi. Untuk rumus yang diajukan lebih bertujuan untuk mengaburkan

pengulangan huruf sehingga kriptanalis akan bingung atau "tertipu" karena sebenarnya hasil dekripsinya belum selesai atau ternyata bukanlah upa-kata yang dituju.
Modifikasi yang diajukan adalah sebagai berikut:

a. Memecah plainteks menjadi digraf sama seperti playfair cipher
b. Membentuk tabel playfair cipher
c. Proses enkripsi dengan penambahan perkalian hill cipher ketika didapatkan cipherteks hasil enkripsi dari playfair cipher biasa

Adapun syarat dari perkalian hill cipher pada kombinasi ini adalah sebagai berikut:

- Pembentukan matriks berdasarkan tabel playfair cipher dengan ukuran 2x2 disesuaikan dengan digraf

Kunci Hill: $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$

- Prioritas perkalian yang diutamakan adalah huruf pertama yang ditemukan dari hasil enkripsi yang menghasilkan digraf sebagai matriks kunci pada posisi a pada matriks
- Kemudian b diambil dari hasil kedua pada digraf sedangkan c dan d adalah huruf pembentuk antara plainteks dan kunci

Contoh:

Plainteks: **AKU CINTA KRIPTOGRAFI**
Plainteks dalam digraf:
AK UC IN TA KR IP TO GR AF IZ
Kunci: **GANTENG**

Tabel Playfair

g	a	n	t	e	g
b	c	d	f	h	b
i	k	l	m	o	i
p	q	r	s	u	q
v	w	x	y	z	v
g	a	n	t	e	

- Proses enkripsi digraf kedua yaitu UC

g	a	n	t	e	g
b	c	d	f	**h**	b
i	k	l	m	o	i
p	**q**	r	s	u	q
v	w	x	y	Z	v
g	a	n	t	E	

Didapatkan huruf Q dan H sehingga hasil ciphernya yaitu QH dan disimpan terlebih dahulu

- Mengambil matriks kunci hill dengan representasi karakter dimulai dari 0 (a = 0)

Kunci hill: $\begin{pmatrix} q & h \\ u & c \end{pmatrix} = \begin{pmatrix} 16 & 7 \\ 20 & 2 \end{pmatrix}$

- Mengalikan hasil cipher pertama dengan matriks kunci yang didapat sehingga:

$$C = \begin{pmatrix} q & h \\ u & c \end{pmatrix}\begin{pmatrix} q \\ h \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 16 & 7 \\ 20 & 2 \end{pmatrix}\begin{pmatrix} 16 \\ 7 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 256+49 \\ 320+14 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 305 \\ 334 \end{pmatrix} \bmod 26 = \begin{pmatrix} 19 \\ 22 \end{pmatrix} = \begin{pmatrix} t \\ w \end{pmatrix}$$

Sehingga hasil akhirnya adalah **TW,** dst.

B. Kombinasi untuk Mengaburkan Exhaustive Search Attack dan Known-Plaintext Attack

Pada teknik cipher tertentu, dapat dipecahkan dan dicari kunci beserta plainteksnya dengan menggunakan metode *exhaustive search*. Metode ini dilakukan pada affine yang tanpa mengalami modifikasi. Pada makalah ini akan dicoba modifikasi dari affine cipher untuk mencegah terjadinya kebocoran dengan metode *exhaustive search* atau dengan *known-plaintext* attack sehingga keamanan dari plainteks lebih terjamin.

Pada algoritma kriptografi klasik, untuk memperbesar faktor kerja dari *exhaustive search* adalah dengan mengelompokkan plainteks menjadi beberapa blok huruf. Sedangkan untuk mencegah *known-plaintext* attack, harus dihindari nilai faktor pengali yang sama sehingga tidak terjadi kekongruenan simultan dari teks. Akan tetapi, dalam makalah ini tidak akan menggunakan modifikasi tersebut. Modifikasi yang akan dilakukan adalah dengan penggabungan teknik cipher yang lain dengan affine cipher sebagai dasarnya.

Modifikasi yang diajukan adalah dengan menggunakan tabel vigenere untuk pembangkit nilai dari faktor pengali. Tabel vigenere dipilih dengan alasan dapat mencegah terjadinya perulangan huruf dengan jarak yang cukup jauh jika dimodifikasi, terutama jika kunci cukup panjang. Dengan modifikasi ini tentunya akan terbangkitkan nilai yang cukup acak sebagai faktor pengali.
Contoh:
Plainteks: **AKU CINTA KRIPTOGRAFI**
Kunci: **GANTENG**

Rumus : C = mP +K (mod n)
dengan m didapat dari hasil perkalian digraf yang didapat dari tabel vigenere

Rumus akhir:
C = (P+K (mod n))P + K (mod n)

Contoh enkripsi karakter kedua:

- Mendapatkan nilai m dengan melihat tabel vigenere
  - Perpotongan K dan A adalah K
- Hitung C

  C = mP + K (mod n)
  = 10*10 + 0 (mod 26)
  = 100 (mod 26) = 22 = w

  Didaptkan hasil enkripsinya adalah huruf **W**
- Lakukan hal yang sama untuk karakter lainnya sehingga didapatkan cipherteksnya

IV. PERBANDINGAN KEAMANAN HASIL KOMBINASI

Pada makalah ini menghasilkan tiga buah kombinasi teknik *cipher* yang baru yaitu:

1. Kombinasi vigenere cipher dan affine cipher
2. Kombinasi playfair cipher dan hill cipher
3. Kombinasi affine cipher dengan tabel vigenere-playfair

Supaya setara, perbandingan keamanan akan dilakukan dalam bigraf dan dihitung berdasar tolak ukur berikut:

1. Jumlah perulangan bigraf terbanyak
2. Jarak kemunculan huruf terdekat

Plainteks yang akan digunakan adalahs sebagai berikut dengan hanya memerhitungkan alphabet dan mengabaikan tanda baca. Ada terdapat beberapa kata yang berulang pada teks ini seperti “sudah”, “kapan”, dll yang nanti akan bisa dilihat hasil ciphernya apakah plainteks tersamarkan dengan baik atau tidak. Dengan tolak ukur yang ada nanti dapat terlihat hasilnya.

Jawa Timur Bakal Tenggelam

Semburan lumpur panas di desa Porong, Sidoarjo, Jawa Timur belum juga berakhir. Sudah beberapa desa tenggelam. Entah sudah berapa rumah, bangunan, pabrik, dan sawah yang tenggelam.

Sampai kapan semburan lumpur berhenti, tiada yang tahu. Teknologi manusia tidak berhasil menutupi lubang semburan. Jika semburan lumpur tidak berhenti juga, mungkin Jawa Timur akan tenggelam

Kunci: GANTENG

A. Hasil Cipher Kombinasi 1

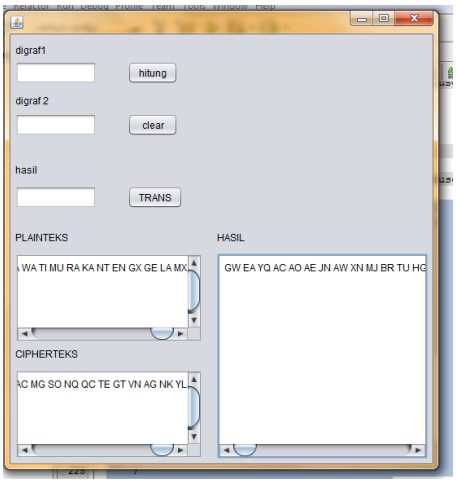
Kode Java

```
// megambil plaintext
String plaintext = jTextArea1.getText();
// mengambil key
 String key = jTextField1.getText();
// panjang dari plaintext
int ptext = plaintext.length();
// panjang dari key
int pkey = key.length();
int i = 0;  // iterasi plaintext
int j = 0;  // iterasi kunci
plaintext = plaintext.toLowerCase();
key = key.toLowerCase();
int Fibonacci = 1;
while (i < ptext){
        char dapet = plaintext.charAt(i++);
        char keydapet = key.charAt(j++);
        char jadi;

        if (j > pkey-1)
        {
           j = 0;
        }
        // encipher normal
        if ((int)dapet<'A' || ((int)dapet>'Z' &&
(int)dapet<'a') || (int)dapet>'z') {
                jadi = dapet;
                if(j>0)
                {j--;}
        }
        else {
                jadi = (char)(fibonacci*((int)keydapet - 'a' +
(int)dapet - 'a')%26 + 'a');
        }
        hasil = hasil+jadi; // cipher normal

}
```

Hasil cipher dalam digraf dan tanpa tanda baca

pa jt xv sa ro to nr ze az kr rg mf xq oa xa ae yz va rc tr ny ji qx wn vo eh rt yi qh ee pu jn pe go su eu iy as jh ze ok xa xa me yu qt lo kh ee tt nj ef tx rt mg re ez kn gt lf aj au ui eg va en qn nh aa zy ag tp nu vv qj aa le jg ny ng kg kt gt xp ns ya zi ev qa ct rf ks bh ke ar uz iy eh ee ai az ot vt hn ea az xn na tr dr br ug vf ea ay in mm qg qb rk ln yo lz xr hz ap ve yo gt gf xq oa xa ac mx gy ez uy eg tl hf th xz iq to ok xh rg xv pu tt qh tm kv gn nc gt vf ye gk ng xr tm gr ee zz

B. Hasil Cipher Kombinasi 2



Gambar 2. Program untuk penghitungan digraf dan enkripsi kombinasi cipher ke-dua pada makalah

Kode Java untuk modifikasi hasil playfair cipher ke dalam kombinasi

```
String s1 = jTextArea1.getText().toLowerCase();
String s2 = jTextArea2.getText().toLowerCase();int
length = s1.length()-1;
   String hasil = new String();
   for (int i=0;i<=length;i++)
   {
      if(i%3==0)
      {
         int c = (int)s1.charAt(i) - 97;
         int d = (int)s1.charAt(i+1) - 97;
         int a = (int)s2.charAt(i) - 97;
         int b = (int)s2.charAt(i+1) - 97;

         int h1 = ((a*a + b*b) % 26)+97;
         int h2 = ((c*a + d*b) % 26)+97;
         hasil += (char)h1;
         hasil += (char)h2;
         hasil += " ";
      }
   }
   jTextArea3.setText(hasil.toUpperCase());
```

Digraf yang terbentuk

IA WA TI MU RB AK AL TE NG XG EL AM SE MB UR AN LU MP UR PA NA SD ID ES AP OR ON GS ID OA RI OI AW AT IM UR BE LU MI UG AB ER AK HI RS UD AH BE BE RA PA DE SA TE NG XG EL AM EN TA HS UD AH BE RA PA RU MA HB AN GU NA NP AB RI KD AN SA WA HY AN GT EN GX GE LA MS AM PA IK AP AN SE MB UR AN LU MP UR BE RH EN TI TI AD AY AN GT AH UT EK NO LO GI MA NU SI AT ID AK BE RH AS IL ME NU TU PI LU BA NG SE MB UR AN IX IK AS EM BU RA NL UM PU RT ID AK BE RH EN TI XI UG AM UN GK IN IA WA TI MU RA KA NT EN GX GE LA MX

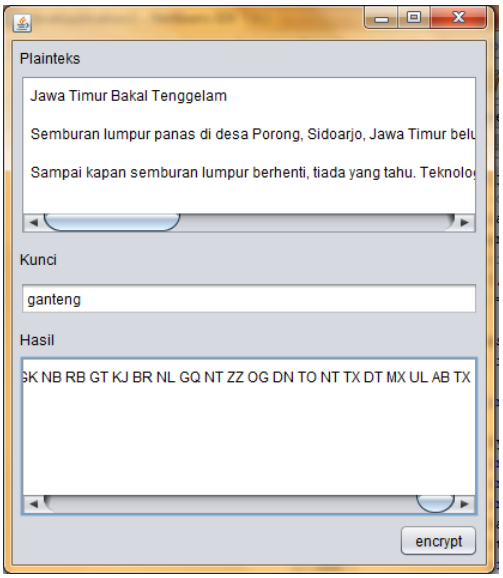
Hasil cipher playfair biasa

GK AC MG SO DP CQ KN EG TA NV ON KT TU FI PS NT RO SI PS GQ TN FR BL UT QG UL EL PT BL EK LP IK CA NE KO PS GH RO OK EP CG UN CQ OB SU HR CE GH GH NQ GQ NH TQ EG TA NV ON KT GT EN UF HR CE GH NQ GQ SP TK BC NT PE TN RG CG LP CL NT TQ AC ZF NT AE GT VN AG NK SY KT GQ KL QG NT TU FI PS NT RO SI PS GH DU GT MG MG CN WT NT AE CE ES OA LE MI BP TK RE MP NE BL CQ GH DU QT KM TO RE SE VP RO GC TA TU FI PS NT VL KL QT OT PH NQ DR OS QP NS BL CQ GH DU GT MG LV EP KT ER IA GL GK AC MG SO NQ QC TE GT VN AG NK YL

Hasil setelah modifikasi

GW EA YQ AC AO AE JN AW XN MJ BR TU HG LQ DI KN RZ YY DI GM KN CL SP HG GM BZ HR OQ SP ME IV IK EA DY KO DI HI RZ KO HO OG XP AE PC WQ AJ UC HI HI JN GM KP TE AW XN MJ BR TU HL DY JW AJ UC HI JN GM DI TU FJ KN HO KN NZ OG IV VB KN TE EA AJ KN QY HL MJ KY JN QY TU GM NI GM KN HG LQ DI KN RZ YY DI HI TJ HL YQ YQ RN NO KN QY UC CG OE HR AK SW TU TP FY DY SP AE HI TJ TE KE LY TP CG QT RZ OG XN HG LQ DI KN QF NI TE LY OZ JN MS AC NU ZR SP AE HI TJ HL YQ QF HO TU TP MW BJ GW EA YQ AC JN AE NL HL MJ KY JN VV

C. Hasil Cipher Kombinasi 3



Gambar 3. Program hasil modifikasi affine cipher

Kode Java

```
String plainteks = jTextArea1.getText().toLowerCase();
   String kunci = jTextField1.getText().toLowerCase();
   // iterasi kunci
   int j = 0;
   int iter = 0;

   String hasil = new String();
   for (int i=0;i<=plainteks.length()-1;i++)
   {
      if((int)plainteks.charAt(i) >= 'a' || (int)plainteks.charAt(i) <= 'a')
      {
         System.out.println("i: "+i);
         int get = (int)plainteks.charAt(i) - 'a';
         int getk = (int)kunci.charAt(j++) - 'a';
         int add = ((((get+getk)%26)*get) + getk) % 26 + 'a';
         hasil+= (char)add;
         iter++;
         if(j > kunci.length()-1)
         {
            j=0;
         }
         if(iter>0 && iter%2 == 0)
         {
            hasil+=" ";
         }
      }
   }
   jTextArea2.setText(hasil.toUpperCase());
```

Hasil cipher dalam digraf dan tanpa tanda baca

LA DT RX OO KD TJ NK GR NN KN AA QR TO NH HM DB JX HG NN LQ BJ GD NJ EN GW NJ BR JU WA NJ WD AT KB TK ZH AA DL WB TL AD TR XO OK DT JD LG ON LQ XG TB DH EJ TO DB TK XH GX NN KB UH AR TR JU WA NN KN AA QR TO BT UN XT DN WG JN TR BU HA RT RD GO AX BR BG

TK XT EN BT RN NX ZK BN JT RN WG QN TR RG
TK NN KN AA QR TO BT HD ZT OR GO NJ TD NT
TM DB JX HG NN LQ BJ GD NN KD TU NX BB NN
OA JT RR GT KN NE XG NN XH ON AL OX BR BG
TK ZB EN NO JN XR BU HX NJ WR TO QN TZ XJ
ON RT JN TA NZ HO BG HA NN RD OK AN JK BN
GD NT RR GO RX HR XO HA JT JD HT QN NW NL
GK NB RB GT KJ BR NL GQ NT ZZ OG DN TO NT
TX DT MX UL AB TX

Tabel Perbandingan kekuatan cipher dimana K adalah hasil kombinasi dari dua teknik cipher

Cipher	Bigraf Berulang	Jarak TKB	KKT
K1	6 (AA,SA)	12	32 (A)
K2	10 (KN)	2	27 (N)
K3	14 (NN)	3	64 (N)

Penghitungan data yang dilakukan pada tabel dilakukan menggunakan aplikasi CryptoHelper dari Gary Watson ©

Keterangan:
Jarak TKB: (dengan satuan karakter)
jarak terdekat dari bigraf terbanyak yang berulang pada cipherteks
KKT: Kemunculan Karakter Terbanyak

V. KESIMPULAN

Berdasarkan dari data yang ada, dapat dilihat bahwa kombinasi teknik cipher pertama adalah yang terkuat. Hal ini dikarenakan hasil kombinasi penghitungan bigraf dan penghitungan frekuensi menghasilkan hasil yang optimal, masing-masing paling sedikit dan paling jauh untuk bigraf berulang dan jarak karakter terjauh. Tetapi hal itu hanya dilihat dari kasat mata analisis frekuensi. Karena sebenarnya dalam algoritma pengenkripsiiannya sudah dimanipulasi supaya cukup membingungkan kriptanalisis dengan analisis frekuensi yang ada dimana bigraf yang sama dapat menghasilkan karakter yang berbeda.

REFERENCES

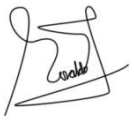
[1] Munir, Rinaldi, Ir.,M.T. 2005. Diktat Kuliah IF-5054 Kriptografi. Bandung : Informatika ITB
[2] http://en.wikipedia.org/wiki/Affine_cipher
[3] http://en.wikipedia.org/wiki/Hill_cipher
[4] http://en.wikipedia.org/wiki/Playfair_cipher
[5] http://en.wikipedia.org/wiki/Vigen%C3%A8re_cipher
[6] http://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi
[7] http://www.simonsingh.net/The_Black_Chamber/playfair_cipher.html

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 19 Maret 2012

ttd



Georgius Rinaldo Winata / 13509030