

## Implementasi Algoritma Enkripsi Playfair pada File Teks

**Rina Chandra Noer Santi**

Program Studi Teknik Informatika  
Fakultas Teknologi Informasi, Universitas Stikubank  
email : r\_candra\_ns@yahoo.com

### Abstrak

Kriptografi berasal dari kata *crypto* yang berarti rahasia dan *graphy* yang berarti tulisan. Jadi kriptografi dapat diartikan sebagai tulisan rahasia. Secara istilah dapat didefinisikan sebagai studi tentang teknik-teknik matematika yang berhubungan dengan keamanan informasi.

Playfair merupakan digraphs cipher, artinya setiap proses enkripsi dilakukan pada setiap dua huruf. Adapun tujuan yang akan di capai adalah membuat aplikasi untuk pengamanan pada file text dengan menggunakan metode Playfair yang dapat mendukung proses perlindungan data yang tidak mudah dicuri dan tidak mudah dipecahkan yang dapat digunakan sebagai keamanan pada data-data yang sangat penting.

**Kata Kunci** : Kriptografi, Enkripsi, *Playfair*, Delphi 6.0

### PENDAHULUAN

Jaringan komputer dan internet telah mengalami perkembangan yang sangat pesat. Teknologi ini mampu menghubungkan hampir semua komputer yang ada di dunia sehingga dapat saling berkomunikasi dan bertukar informasi berupa data teks seperti data keuangan, data user name dan password dari account suatu perusahaan, gambar bergerak, suara maupun email. Seiring dengan perkembangan tersebut, secara langsung ikut mempengaruhi cara berkomunikasi. Jika dahulu untuk berkomunikasi pesan atau surat dengan menggunakan pos, sekarang telah banyak layanan *e\_mail* di internet yang dapat mengirimkan pesan secara langsung kepenerimanya. Akan tetapi sebagai suatu jaringan publik, internet rawan sekali terhadap pencurian data. Maka dan salah satu cara untuk melindungi data dengan menggunakan seni Kriptografi.

#### Pengertian Kriptografi

Secara bahasa Kriptografi berasal dari kata *crypto* yang berarti rahasia dan *graphy* yang berarti tulisan. Jadi kriptografi dapat diartikan sebagai tulisan rahasia. Secara istilah dapat didefinisikan sebagai studi tentang teknik-teknik

matematika yang berhubungan dengan keamanan informasi.

Teknik kriptografi terdiri dari simetri dan asimetri. Teknik ini digunakan untuk mengamankan aplikasi (keamanan informasi) sehingga dapat menjaga kerahasiaan, integritas data, autentikasi data dan *non-repudiation*

Kriptografi diperlukan karena pada dasarnya informasi sangat penting bagi segala aspek, tuntutan keamanan informasi berubah dari waktu ke waktu. Perubahan tuntutan ini terjadi karena transformasi atau penggunaan perlengkapan kebutuhan utama untuk pertukaran informasi, dari mulai cara tradisional (fisik) yang membutuhkan mekansime pengarsipan atau administrasi secara fisik dan membutuhkan ruang yang lebih besar, menggunakan otomatisasi komputer personal, sampai transfer informasi melalui penggunaan jaringan komputer, baik intranet maupun internet yang sekarang menjadi tren dan kebutuhan.

Kriptografi secara umum merupakan ilmu dan seni untuk menjaga kerahasiaan berita (*Bruce Schneier–Applied Cryptography*). Kriptografi juga dapat diartikan sebagai ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi

seperti kerahasiaan data, keabsahan data, integritas data, serta Otentikasi data (A. Menezes, P. Van Oorschot and S. Vanstone—Handbook of Applied Cryptography). Namun, pada kriptografi tidak semua aspek keamanan informasi akan ditangani.

Ada empat tujuan mendasar dari kriptografi yang juga merupakan aspek keamanan informasi adalah :

1. Kerahasiaan (*Confidentiality*)

Adalah layanan yang digunakan untuk menjaga isi dari informasi dari siapapun kecuali yang memiliki kunci rahasia atau otoritas untuk membuka informasi yang telah disandikan.

2. Integritas Data (*Data Integrity*)

Berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk dapat menjaga integritas data, suatu sistem harus memiliki kemampuan untuk mendeteksi manipulasi data yang dilakukan pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pendistribusian data lain ke dalam data yang asli.

3. Otentifikasi (*Authentication*)

Berhubungan dengan identifikasi, baik secara kesatuan sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan harus diOtentikasi keasliannya, isi datanya, waktu pengiriman dan lain sebagainya.

4. Non-repudiasi (*Non-repudiation*)

Merupakan usaha untuk mencegah terjadinya penyangkalan terhadap pengiriman atau terciptanya suatu informasi oleh yang mengirimkan atau membuat.

**Algoritma Kriptografi**

Berdasarkan kunci yang dipakai, algoritma kriptografi dapat dibedakan atas dua golongan, yaitu :

a. Algoritma Simetris (*Symmetric Algorithms*)

Dalam *Symmetric Algorithms* ini, kunci yang digunakan untuk proses enkripsi dan dekripsi pada prinsipnya identik  $K_1 = K_2 = K$ , tetapi satu buah kunci dapat pula diturunkan dari

kunci yang lainnya. Kunci-kunci ini harus dirahasiakan. Oleh karena itulah sistem ini sering disebut sebagai *secret-key ciphersystem*. Jumlah kunci yang dibutuhkan umumnya:

$$C = \frac{n \cdot (n-1)}{2}$$

dengan n = menyatakan banyaknya pengguna (user)

C = menyatakan banyaknya kunci.

Tingkat keamanan kriptosistem yang menggunakan algoritma ini sangat ditentukan oleh kerahasiaan kunci K yang digunakan . Jika seseorang hendak mengirimkan suatu pesan kepada orang lain , atau melakukan secure communication, orag tersebut harus terlebih dahulu memberikan kepada pihak yang dituju kunci K yang hendak digunakanya. Hal ini jelas membutuhkan saluran komunikasi yang benar-benr aan dan tidak dapat disadap (*secure channel*).

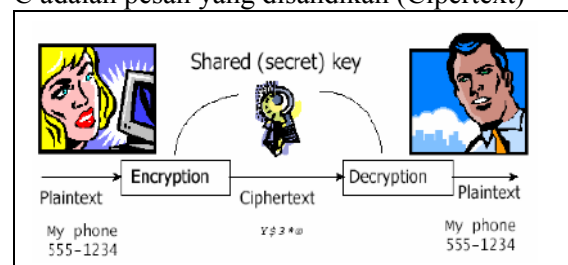
Secara matematis Algotima ini dapat ditulis :

$$E_k(M) = C \quad d_k(C) = M$$

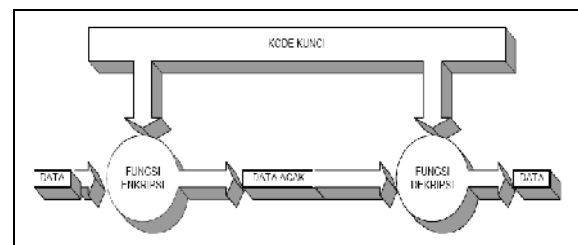
E adalah proses enkripsi dengan menggunakan kunci K

M adalah pesan asli (Plainetext)

C adalah pesan yang disandikan (Ciphertext)



Gambar 1. Algoritma Simetris (*Symmetric Algorithms*)



Gambar 2. Teknik Algoritma Simetris (*Symmetric Algorithms*)

Prinsip kerja dari kriptografi kunci simetrik adalah sebagai berikut :

1. Pengirim dan penerima data atau Informasi sepakat menggunakan system kriptografi tertentu
2. Pengirim dan penerima sepakat menggunakan satu kunci tertentu
3. Dilakukan enkripsi sebelum data dikirim dan dekripsi setelah data dikirim

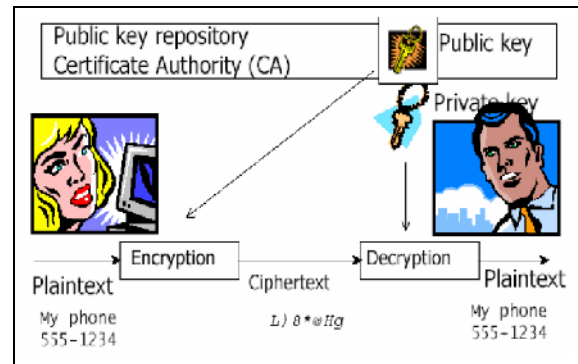
Tingkat keamanan dari kriptosistem yang menggunakan algoritma ini sangat ditentukan oleh kerahasiaan kunci K yang digunakan. Jika seserang hendak mengirimkan suatu pesan kepada orang lain atau melakukan *secure communication*, orang tersebut harus terlebih dahulu memberitahu kepada pihak yang dituju kunci K yang digunakannya. Hal jelas membutuhkan saluran komunikasi yang benar-benar aman tidak data disadap (*secure channel*). Faktor inilah yang menjadi kelemahan cara ini yaitu masalah keamanan kunci dan bagaimana mendistribusikan kunci K tersebut.

b. Algoritma Asimetri (*Asymmetric Algorithms*)

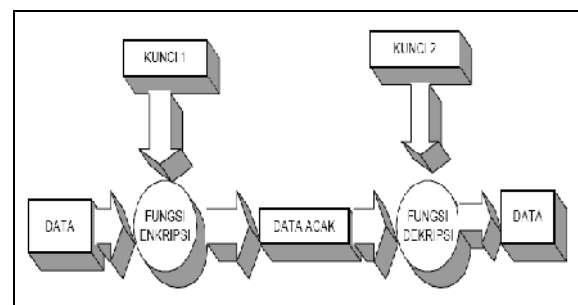
Dalam *Asymmetric Algorithms* ini digunakan dua buah kunci. Satu kunci yang disebut kunci publik (*public key*) dapat dipublikasikan, sedang kunci yang lain yang disebut kunci privat (*private key*) harus dirahasiakan. Proses menggunakan sistem ini dapat diterangkan secara sederhana sebagai berikut : bila A ingin mengirimkan pesan kepada B, A dapat menyandikan pesannya dengan menggunakan kunci publik B, dan bila B ingin membaca surat tersebut, ia perlu mendekripsikan surat itu dengan kunci privatnya. Dengan demikian kedua belah pihak dapat menjamin asal surat serta keaslian surat tersebut, karena adanya mekanisme ini. Contoh sistem ini antara lain RSA Scheme dan Merkle-Hellman Scheme.

Setiap *cryptosystem* yang baik harus memiliki karakteristik sebagai berikut :

1. Keamanan sistem terletak pada kerahasiaan kunci dan bukan pada kerahasiaan algoritma yang digunakan.
2. *Cryptosystem* yang baik memiliki ruang kunci (*keyspace*) yang besar.



Gambar 3. Algoritma Asimetris (*Asymmetric Algorithms*)



Gambar 4. Algoritma Simetris (*Symmetric Algorithms*)

3. *Cryptosystem* yang baik akan menghasilkan ciphertext yang terlihat acak dalam seluruh tes statistik yang dilakukan terhadapnya.
4. *Cryptosystem* yang baik mampu menahan seluruh serangan yang telah dikenal sebelumnya

Namun demikian perlu diperhatikan bahwa bila suatu *cryptosystem* berhasil memenuhi seluruh karakteristik di atas belum tentu ia merupakan sistem yang baik. Banyak *cryptosystem* lemah yang terlihat baik pada awalnya. Kadang kala untuk menunjukkan bahwa suatu *cryptosystem* kuat atau baik dapat dilakukan dengan menggunakan pembuktian matematika. Hingga saat ini masih banyak orang yang menggunakan *cryptosystem* yang relatif mudah dibuka, alasannya adalah mereka tidak mengetahui sistem lain yang lebih baik serta kadang kala terdapat motivasi yang kurang untuk menginvestasikan seluruh usaha yang diperlukan untuk membuka suatu system.

**Sandi Playfair**

Sandi Playfair digunakan oleh Tentara Inggris pada saat Perang Boer II dan Perang

Dunia I. Ditemukan pertama kali oleh Sir Charles Wheatstone dan Baron Lyon Playfair pada tanggal 26 Maret 1854. Playfair merupakan digraphs cipher, artinya setiap proses enkripsi dilakukan pada setiap dua huruf. Misalkan plainteksnya “KRIPTOLOGI”, maka menjadi “KRIPTOLOGI”. Playfair menggunakan tabel 5x5. Semua alfabet kecuali J diletakkan ke dalam tabel. Huruf J dianggap sama dengan huruf I, sebab huruf J mempunyai frekuensi kemunculan yang paling kecil. Kunci yang digunakan berupa kata dan tidak ada huruf sama yang berulang. Apabila kuncinya “MATAHARI”, maka kunci yang digunakan adalah “MATHRI”. Selanjutnya, kunci dimasukkan ke dalam tabel 5x5, isian pertama adalah kunci, selanjutnya tulis huruf-huruf berikutnya secara urut dari baris pertama dahulu, bila huruf telah muncul, maka tidak dituliskan kembali.

Tabel 1. Kunci Matahari

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

Berikut ini aturan-aturan proses enkripsi pada Playfair yaitu

1. Jika kedua huruf tidak terletak pada baris dan kolom yang sama, maka huruf pertama menjadi huruf yang sebaris dengan huruf pertama dan sekolom dengan huruf kedua. Huruf kedua menjadi huruf yang sebaris dengan huruf kedua dan yang sekolom dengan huruf pertama. Contohnya, SA menjadi PH, BU menjadi EP.
2. Jika kedua huruf terletak pada baris yang sama maka huruf pertama menjadi huruf setelahnya dalam baris yang sama, demikian juga dengan huruf kedua. Jika terletak pada baris kelima, maka menjadi baris pertama, dan sebaliknya. Arahnya tergantung dari posisi huruf pertama dan kedua, pergeserannya ke arah huruf kedua. Contohnya, AH menjadi TR, LK menjadi KG, BE menjadi CI.

3. Jika kedua huruf terletak pada kolom yang sama maka huruf pertama menjadi huruf setelahnya dalam kolom yang sama, demikian juga dengan huruf kedua. Jika terletak pada kolom kelima, maka menjadi kolom pertama, dan sebaliknya. Arahnya tergantung dari posisi huruf pertama dan kedua, pergeserannya ke arah huruf kedua. Contohnya, DS menjadi LY, PA menjadi GW, DH menjadi HY.
4. Jika kedua huruf sama, maka letakkan sebuah huruf di tengahnya (sesuai kesepakatan).
5. Jika jumlah huruf plainteks ganjil, maka tambahkan satu huruf pada akhirnya, seperti pada aturan ke-4.

Sedangkan proses dekripsinya adalah kebalikan dari proses enkripsi. Contohnya, HR didekrip menjadi HT, BS didekrip menjadi DP, ZU didekrip menjadi RZ.

## HASIL IMPLEMENTASI

### Form Enkripsi



Gambar 5. Form Enkripsi

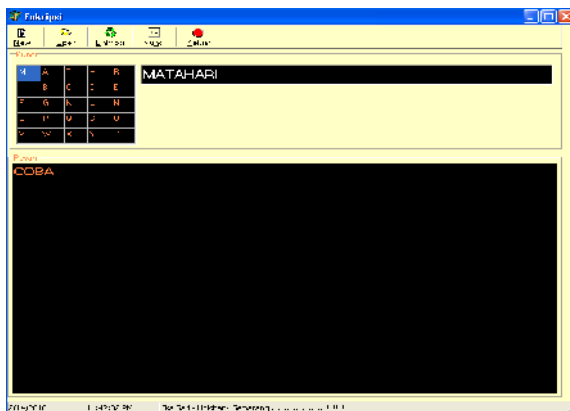
Pada form enkripsi terdapat 5 tombol yang dapat digunakan yaitu tombol **New** yang digunakan untuk membuat file baru yang akan di enkripsi, tombol **Open** yang digunakan untuk membuka file yang akan dienkripsi, tombol **Enkripsi** yang digunakan untuk melakukan enkripsi file dengan menggunakan metode playfair, tombol **Kunci** yang digunakan untuk membuat kunci dengan bentuk tabel 5 x 5 secara otomatis dan tombol **Keluar** yang digunakan untuk keluar dari program enkripsi.

Pada proses enkripsi, program akan membuat kunci kriptografi dengan sandi (“MATAHARI”) dengan ukuran tabel 5 x 5. Pada metode playfair kunci yang dimasukkan tidak boleh ada kata yang berulang dan semua huruf kecuali J tidak dimasukkan ke dalam tabel sehingga jika menggunakan kunci “MATAHARI” maka akan menjadi “MATHRI”. Selanjutnya, kunci dimasukkan ke dalam tabel 5x5 dimana isian pertama adalah kunci, selanjutnya tulis huruf-huruf berikutnya secara urut dari baris pertama dahulu, bila huruf telah muncul, maka tidak dituliskan kembali sampai tabel terisi semua.

Tabel 2. Kunci “MATAHARI”

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

Setelah itu, program akan mengelompokkan pesan (“coba”) yang akan dienkripsi yang terdapat pada komponen richedit masing-masing menjadi 2 huruf dan membuat semua karakter menjadi huruf besar (*upper case*) dan menyimpan ke dalam array tmp1 dengan urutan sebagai berikut:



Gambar 6. Pesan Enkripsi “coba”

$\text{Tmp1}[0] : = \text{CO}$   
 $\text{Tmp1}[2] : = \text{BA}$

Setelah itu program akan melakukan pengecekan sesuai dengan metode enkripsi playfair dimana :

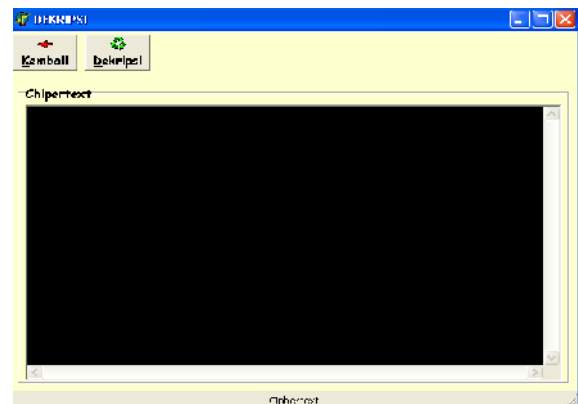
1. Jika kedua huruf tidak terletak pada baris dan kolom yang sama, maka huruf pertama menjadi huruf yang sebaris dengan huruf pertama dan sekolom dengan huruf kedua. Huruf kedua menjadi huruf yang sebaris dengan huruf
2. Jika kedua huruf terletak pada baris yang sama maka huruf pertama menjadi huruf setelahnya dalam baris yang sama, demikian juga dengan huruf kedua. Jika terletak pada baris kelima, maka menjadi baris pertama, dan sebaliknya. Arahnya tergantung dari posisi huruf pertama dan kedua, pergeserannya ke arah huruf kedua.
3. Jika kedua huruf terletak pada kolom yang sama maka huruf pertama menjadi huruf setelahnya dalam kolom yang sama, demikian juga dengan huruf kedua. Jika terletak pada kolom kelima, maka menjadi kolom pertama, dan sebaliknya. Arahnya tergantung dari posisi huruf pertama dan kedua, pergeserannya ke arah huruf kedua.

pesan ciphertext sebagai berikut :



1. Pesan “CO” tidak terletak pada baris dan kolom yang sama sehingga masuk ke aturan playfair enkripsi no 1, sehingga pesan “CO” menjadi C = “IQ”
2. Pesan “BA” terletak pada kolom yang sama sehingga masuk ke aturan playfair enkripsi no 3, sehingga pesan “BA” menjadi C = “GB”

Sehingga program akan mengenkripsi pesan “coba” dengan ciphertext ”IQGB”

**Proses Dekripsi**



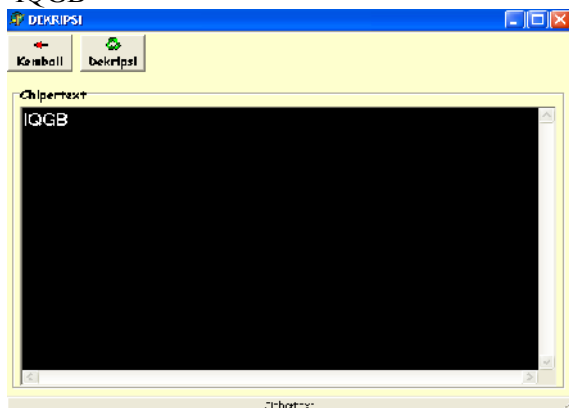
Gambar 7. Form Dekripsi

Pada proses dekripsi terdapat 2 tombol yang dapat digunakan yaitu tombol  yang digunakan yang digunakan untuk kembali ke form enkripsi dan menutup form dekripsi, tombol  yang digunakan untuk melakukan proses dekripsi file yang telah dienkripsi.

Pada proses dekripsi kebalikan dengan proses enkripsi dengan ketentuan dekripsi playfair sebagai berikut :

1. Jika kedua huruf tidak terletak pada baris dan kolom yang sama, maka huruf pertama menjadi huruf yang sebaris dengan huruf pertama dan sekolom dengan huruf kedua. Huruf kedua menjadi huruf yang sebaris dengan huruf kedua dan sekolom dengan huruf pertama.
2. Jika kedua huruf terletak pada baris yang sama maka huruf pertama menjadi huruf sebelumnya dalam baris yang sama, demikian juga dengan huruf kedua. Jika terletak pada baris kesatu, maka menjadi baris kelima, dan sebaliknya.
3. Jika kedua huruf terletak pada kolom yang sama maka huruf pertama menjadi huruf sebelumnya dalam kolom yang sama, demikian juga dengan huruf kedua. Jika terletak pada kolom pertama, maka menjadi kolom kelima, dan sebaliknya.

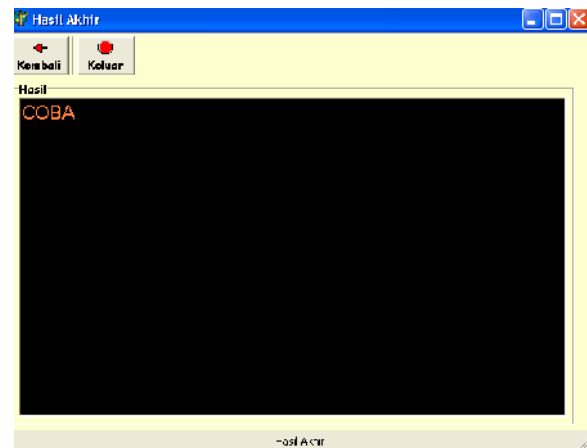
Pada proses dekripsi program akan melakukan pencarian pesan dan mengelompokkan menjadi 2 huruf yang dienkripsi dan menyimpan ke dalam array tmp1. Pesan "coba" dienkripsi menjadi chipertext "IQGB"





Gambar 8. Pesan Enkripsi "COBA"

1. Pesan "IQ" tidak terletak pada baris dan kolom yang sama sehingga masuk ke aturan playfair dekripsi no 1, sehingga pesan "IQ" menjadi M = "CO"
2. Pesan "GB" terletak pada kolom yang sama sehingga masuk ke aturan playfair dekripsi no 3, sehingga pesan "GB" menjadi M = "BA"

Sehingga akan didapatkan pesan "COBA" dan pesan setelah didekripsi sama pada saat pesan sebelum dienkripsi



Gambar 9. Hasil Dekripsi

Pada form hasil akhir terdapat 2 tombol yang dapat digunakan yaitu tombol  yang digunakan untuk kembali ke form dekripsi dan tombol  untuk menutup form hasil dan kembali ke form utama.

" COBA LAGI DEH "  
Dengan kunci " MATAHARI"

	1	2	3	4	5
1	M	A	T	H	R
2	I	B	C	D	E
3	F	G	K	L	N
4	O	P	Q	S	U
5	V	W	X	Y	Z

Plaintext	CO	BA	LA	GI	DU	LU
-----------	----	----	----	----	----	----

Baris	24	21	31	32	24	34
-------	----	----	----	----	----	----

Kolom	31	22	42	21	45	45
-------	----	----	----	----	----	----

**Di Enkripsi**

Chypertext	IQ	GB	GH	FB	ES	NS
------------	----	----	----	----	----	----

Baris	24	32	31	32	24	34
-------	----	----	----	----	----	----

Kolom	13	22	24	12	54	54
-------	----	----	----	----	----	----

**PENUTUP**

Kesimpulan

- a. Program aplikasi kriptografi ini akan membatasi orang yang tidak berhak atas informasi atau data yang dimiliki oleh si pengirim untuk dibaca karena pesan sudah dienkripsi dan dapat menjaga kerahasiaan pesan atau informasi file-file yang ada dalam sebuah komputer.
- b. Pembuatan teknik kriptografi enkripsi dekripsi dengan menggunakan metode playfair dapat melindungi data dimana program akan melakukan proses enkripsi hanya berupa huruf dengan menggunakan tabel 5 X 5.

**DAFTAR PUSTAKA**

Elka ,Lab, 2001, *Pelatihan Delphi*, www.planck.fi.itb.ac.id

Jogiyanto.HM, 2002, *Analisis dan Desain Sistem Informasi*, Andi Offset, Yogyakarta.

Kadir,Abdul, 2001, *Dasar Pemrograman Delphi 6.0*, Andi Offset, Yogyakarta.

Mahyusir,Tavri D. 2002, *Pengantar Analisis dan Perancangan Perangkat Lunak*, Andi Offset, Yogyakarta

Mahyusir,Tavri D 1991, *Pengantar Analisis dan Perancangan Perangkat Lunak*, Andi Offset,Yogyakarta.

Munir,Rinaldi, 2006, *Diktat Kuliah IF5054 Kriptografi*, Institut Teknologi Bandung, Bandung.

Musalini,Uus 2004, *Membangun Aplikasi Super Cantik Dan Full Animasi Dengan Delphi*, PT. Elex media Computindo, Jakarta

Pressman ,Roger S, 2002, *Rekayasa Perangkat Lunak*, Andi Offset,Yogyakarta

Riyanto,M Zaki, 2008, *Kriptografi Pada Perang Dunia I : Sandi Playfair*, <http://zaki.math.we.id>, Yogyakarta.

Sisyboy, 2004, *Flowchart dan Source Code RSA*, <http://sisyboy.files.wordpress.com/2008/01/flowchart.doc>

Whitten ,Jeffery L., 2004, *Metode Desain dan Analisa Sistem*, Andi Offset, Yogyakarta

46goenk, 2008, *RSA Algorithm* , <http://agcrypt.wordpress.com/2008/02/25/rsa-algorithm/>