Implementasi Kompilasi Algoritma Kriptografi Transposisi Columnar Dan Rsa Untuk Pengamanan Pesan Rahasia

Yuza Reswan[1], Ujang Juhardi[2], Bobi Tri Yuliansyah[3]
[1,2,3]Teknik Informatika, Universitas Muhammadiyah Bengkulu
yuzareswan@umb.ac.id[1], ujangjuhardi@umb.ac.id[2], bobi3yuliansyah25@gmail.com[3]

Abstract ~ Data security is important in maintaining the confidentiality of certain data that can only be known by those who have rights. If the data transmission is done by using network, there is a big possibility the data to be known by unauthorized parties.
The columnar transposition algorithm is one simple transposition password. Columnar transposition is one of the classic cryptographic algorithms. Columnar transposition is one part of the transposition cipher with cryptographic method where the message is written in a row from a specified length, and then the column per column is read again with a reading sequence based on a keyword. Series length is determined by the length of the keyword. The order of column readings is based on column order.
RSA Algorithm is one of asymmetric cryptography, which is a type of cryptography that uses two different keys: public key and private key. Thus, there is one key, namely the public key, which can be sent through a free channel, without any particular security. In this case there are two keys arranged so that they have a relationship in modulo arithmetic equation. In this case, combining both columnar transposition algorithms and RSA algorithms allows security in the form of message data to be very effective for locking the data even better.

Abstrak ~ Keamanan data merupakan hal penting dalam menjaga kerahasiaan data-data tertentu yang hanya boleh diketahui oleh pihak yang memiliki hak saja. Apabila pengiriman data dilakukan melalui jaringan, maka kemungkinan data tersebut diketahui oleh pihak yang tidak berhak, menjadi besar.
Algoritma transposisi columnar adalah salah satu sandi transposisi sederhana. Transposisi columnar merupakan salah satu algoritma kriptografi klasik. Transposisi Columnar merupakan salah satu bagian dari *cipher* transposisi dengan metode kriptografi dimana pesan dituliskan berderet dari suatu panjang yang ditetapkan, lalu dibaca kembali kolom per kolom dengan urutan pembacaan berdasarkan suatu kata kunci. Panjang deret ditentukan oleh panjang kata kunci. Urutan pembacaan kolom berdasarkan urutan kolom.
Algoritma RSA merupakan salah satu kriptografi asimetri, yakni jenis kriptografi yang menggunakan dua kunci yang berbeda : kunci public (*public key*) dan kunci pribadi (*private key*). Dengan demikian, maka terdapat satu kunci, yakni kunci publik, yang dapat dikirimkan melalui saluran yang bebas, tanpa adanya suatu keamanan tertentu. Dalam hal ini ada dua kunci tersebut diatur sedemikian sehingga memiliki hubungan dalam suatu persamaan aritmatika modulo. Dalam hal ini, penggabungan kedua algoritma transposisi columnar dan algoritma RSA memungkinkan keamanan pada data yang berbentuk pesan sangat efektif untuk mengunci data menjadi lebih baik lagi.

Kata Kunci : Kriptografi, Transposisi Columnar, RSA, Java

I. PENDAHULUAN

Di era modern seperti sekarang, perkembangan teknologi kian hari semakin berkembang pesat khususnya tingkat keamanan data pada komputer. Keamanan data seperti pesan atau teks merupakan salah satu yang wajib dimiliki dikarenakan setiap pesan yang ada merupakan data yang berbentuk rahasia dan sebagainya.

Pada dunia IT integritas suatu pesan yang dikirim terkadang juga menjadi suatu pertanyaan, apakah data tersebut benar dikirim oleh yang bersangkutan atau tidak, dan apakah isi dari pesan tersebut benar otentik seperti sebelum dikirim ataupun tidak. Hal ini merupakan merupakan masalah yang tidak bisa dipandang sebelah mata saja, karena bisa saja sesorang mengirimkan pesan palsu.

Aktivitas penyimpanan pesan secata digital sangat mempunyai banyak resiko. Bisa dilihat apabila dalam aktivitas tersebut terdapat informasi yang penting dapat diakses oleh orang lain yang tidak berkepentingan (*unauthorized person*), misalnya informasi mengenai password atau PIN.

Kriptografi yang berasal dari kata yunani *"cryptos"* yang artinya rahasia dan *"graphein"* yang artinya tulisan, sehingga kriptografi adalah ilmu untuk

menjaga kerahasiaan pesan dengan cara menyandikan ke bentuk yang tidak dimengerti. Keungulan dari kriptografi adalah kemampuan penyandian pesan sehingga pesan terlihat seperti diacak. Kriptografi tidak sekedar berupa kerahasiaan data (*privacy*) saja, tapi juga bertujuan untuk menjaga integritas data (data integrity), keaslian data (*authentication*) dan anti penyangkalan (*non-repudiation*).

Di dalam kriptografi terdapat 5 hal utama yaitu enkripsi, dekripsi, dan kunci (key), pengirim, dan penerima. Enkripsi merupakan proses penyandian plainteks (pesan awal) menjadi *cipherteks* (pesan yang tersandikan), sedangkan dekripsi merupakan kebalikan dari proses enkripsi. Baik proses enkripsi dan dekripsi, keduanya menggunakan kunci untuk menjaga kerahasiaan data. Penggunaan kriptografi mulai dari penggunaan kartu ATM, penggunaan password untuk file-file dokumen kantor, transaksi dengan kartu kredit, transaksi di bank, percakapan dengan handphone, dan akses internet telah membuktikan pentingnya kriptografi dalam pengamanan informasi.

Bermacam-macam algoritma telah diciptakan untuk mengamankan berbagai jenis data. Algoritma transposisi columnar adalah salah satu sandi transposisi sederhana. Transposisi columnar merupakan salah satu algoritma kriptografi klasik. Transposisi Columnar merupakan salah satu bagian dari *cipher* transposisi dengan metode kriptografi dimana pesan dituliskan berderet dari suatu panjang yang ditetapkan, lalu dibaca kembali kolom per kolom dengan urutan pembacaan berdasarkan suatu kata kunci. Panjang deret ditentukan oleh panjang kata kunci. Urutan pembacaan kolom berdasarkan urutan kolom.

Algoritma RSA merupakan salah satu kriptografi asimetri, yakni jenis kriptografi yang menggunakan dua kunci yang berbeda : kunci public (*public key*) dan kunci pribadi (*private key*). Dengan demikian, maka terdapat satu kunci, yakni kunci publik, yang dapat dikirimkan melalui saluran yang bebas, tanpa adanya suatu keamanan tertentu. Dalam hal ini ada dua kunci tersebut diatur sedemikian sehingga memiliki hubungan dalam suatu persamaan aritmatika modulo.

Dalam hal ini penggabungan kedua algoritma transposisi columnar dan algoritma RSA memungkinkan keamanan pada data yang berbentuk pesan sangat efektif untuk mengunci data menjadi lebih baik lagi.

II. LANDASAN TEORI

1. Algoritma Transposis Coulumnar

Algoritma transposisi columnar merupakan algoritma klasik yang penggunaanya cukup sederhana. Pada tahap enkripsi transposition cipher tidak mengganti huruf plaintext untuk menghasilkan ciphertext layaknya subtitution cipher. Hasil enkripsi transposition cipher didapatkan dari menyusun ulang karakter plaintext dengan posisi yang berbeda.

Plaintext akan ditulis dalam matrik dengan panjang kolom sesuai dengan panjang karakter kunci yang digunakan. Penulisan plaintext ditulis dari baris per baris dimulai dengan baris pertama. Ciphertext transposisi columnar cipher dihasilkan dari penyusunan ulang plaintext. Kolom yang disusun pertama adalah kolom yang berhubungan dengan karakter sesuai urutan abjad.

Contoh enkripsi menggunakan pesan "MEET ME AT NEXT MID NIGHT" dan kunci "FANCY". Penyusunan dimulai dari kolom yang berhubungan dengan karakter urutan pertama pada abjad yaitu "A", kemudian "C", "F", "N" dan "Y". Hasil dari enkripsi tersebut adalah "EATITNIHMEXNETMGMEDT". Model matematis proses enkripsi transposisi columnar cipher menggunakan persamaan:

$$Ct\ of\ P = \quad Yo \ldots\ldots\ldots\ldots Yl \begin{matrix} Xpo1 \\ \ldots\ldots\ldots\ldots \\ Xpo2 \\ \ldots\ldots\ldots\ldots \\ Xpl2 \\ (1) \\ Xpom \\ \ldots\ldots\ldots\ldots \\ Xplm \end{matrix}$$

Keterangan:

$Ct\ of\ P$ = *Columnar Transposition* dari pesan
Yo = Karakter pertama dari kunci
Yl = Karakter terakhir dari kunci
$X\ po$ = Karakter pertama dari pesan yang berelasi dengan 0
$X\ pl1$ = Karakter pertama dari pesan yang berelasi dengan $Y\ l$
$X\ pom$ = Karakter terakhir dari pesan yang berelasi dengan $Y0$
$X\ plm$ = Karakter terakhir dari pesan yang berelasi dengan Y e

Jika pada persamaan (1) $Ct\ of\ P$ didefinisikan sebagai $CtPi$ dengan i adalah kolom pada persamaan (1). Maka cipher text dari proses enkripsi tersebut dapatdimodelkan sebagai :

$$Cp = \{CtP1 + Ct\ P2 + Ct\ P3 + \ldots + CtPm\} \quad (2)$$

Keterangan:
Cp : Hasil enkripsi (Cipher teks)
m : Kolom terakhir dari persamaan (1)

Contoh :

Pada plainteks, misalnya terdapat 18 karakter yang dimasukkan seperti :

Plainteks : FAKULTAS TEKNIK
Kunci : TES

Cara penyelesaiannya adalah dengan membagi setiap karakter dengan jumlah kolom, jumlah kolom ditentukan oleh kunci yang dapat dilihat pada tabel berikut ini.

3	1	2
F	A	K
U	L	T
A	S	
T	E	K
N	I	K

Table 1. Proses Enkripsi

Pembacaan Proses Enkripsi ini akan dibaca dengan tabel secara vertikal sehingga didapatkan hasil seperti "ALSEIKT KKFUATN".

2. Algoritma RSA

Sandi RSA merupakan algoritma kriptografi kunci public (asimetris). Ditemukan pertama kali pada tahun 1977 oleh Ron Rivest, Adi Shamir, dan Len Adleman. Nama RSA sendiri diambil dari ketiga penemunya tersebut. Sebagai algoritma kunci publik, RSA mempunyai dua kunci, yaitu kunci publik dan kunci rahasia. RSA mendasarkan proses enkripsi dan dekripsinya pada konsep bilangan prima dan aritmetika modulo. Baik kunci enkripsi maupun dekripsi keduanya merupakan bilangan bulat. Kunci enkripsi tidak dirahasiakan dan diberikan kepada umum (sehingga disebut dengan kunci publik), namun kunci untuk dekripsi bersifat rahasia (kunci privat). Untuk menemukan kunci dekripsi, dilakukan dengan memfaktorkan suatu bilangan bulat menjadi faktor-faktor primanya. Kenyataannya, memfaktorkan bilangan bulat menjadi faktor primanya bukanlah pekerjaan yang mudah.

Karena belum ditemukan algoritma yang efisien untuk melakukan pemfaktoran. Cara yang bisa digunakan dalam pemfaktoran adalah dengan menggunakan pohon faktor. Jika semakin besar bilangan yang akan difaktorkan, maka semakin lama waktu yang dibutuhkan. Jadi semakin besar bilangan yang difaktorkan, semakin sulit pemfaktorannya, semakin kuat pula algoritma RSA.

Besaran-besaran yang digunakan pada algoritma RSA:

1. p dan q bilangan prima (rahasia)
2. $n = p \cdot q$ (tidak rahasia)
3. $\phi(n) = (p - 1)(q - 1)$ (rahasia)
4. e (kunci enkripsi) (tidak rahasia)
5. d (kunci dekripsi) (rahasia)
6. m (plainteks) (rahasia)
7. c (cipherteks) (tidak rahasia)

2.1 *ASCII System*

Plainteks yang akan dienkripsi dengan RSA *Coding* merupakan angka-angka, sedangkan pesan yang dikirimkan bisaanya berbentuk teks atau tulisan. Sehingga dibutuhkan suatu kode yang sifatnya universal untuk mengubah pesan teks menjadi plain teks dalam bentuk bilangan. ASCII (*American Standard Code for Information Interchange*) atau Kode Standar Amerika untuk pertukaran informasi merupakan suatu standar internasional dalam kode huruf dan symbol seperti *Hex* dan *Unicode* tetapi ASCII lebih bersifat universal, contohnya 124 adalah untuk karakter "|". ASCII selalu digunakan oleh komputer dan alat komunikasi lain untuk menunjukkan teks.

2.2 Aritmatika Modulo

Dalam penerapan *Teorema Euler* pada perumusan algoritma RSA *Coding* sangat dibutuhkan pemahaman tentang modulo. Modulo sendiri berarti sisa hasil bagi. Misalkan a adalah bilangan bulat dan m adalah bilangan bulat dimana a dan m lebih besar dari 0. Maka operasi $a \bmod m$ (dibaca "a modulo m") memberikan sisa jika a dibagi dengan m. Bilangan m disebut modulus atau modulo, dan hasil modulo m terletak di dalam himpunan $\{0, 1, 2, \ldots, m-1\}$

Contoh :

Diambil a = 20 dan m = 6 . Karena 20 dibagi 6 adalah 3 bersisa 2, maka diperoleh a mod m ≡ 20 mod 6 ≡ 2.

2.3 Pembangkitan Pasangan Kunci

Sebagai algoritma Asimetris Kriptografi, pengkodean RSA membutuhkan dua kunci yang berbeda untuk enkripsi dan dekripsi. Bilangan yang dipilih sebagai kunci adalah bilangan prima yang besar, dengan alasan pemfaktoran sebuah bilangan hasil perkalian dari dua bilangan prima yang besar menjadi dua bilangan prima yang sesuai akan sangat sulit. Sehingga keamanan dari RSA *Coding* dapat terjamin.

Berikut langkah-langkah proses pembangkitan pasangan kunci pada RSA:

1. Pilih dua buah bilangan prima sembarang, p dan q.
2. Hitung $n = p \cdot q$. Sebaiknya $p \neq q$, sebab jika $p = q$ maka $n = p_2$ sehingga p dapat diperoleh dengan menarik akar pangkat dua dari n.
3. Hitung $\phi(n) = (p - 1)(q - 1)$.
4. Pilih kunci publik, e , yang relatif prima terhadap $\phi(n)$.
5. Bangkitkan kunci rahasia dengan menggunakan $d \cdot e \equiv 1 \pmod{\phi(n)}$.

Perhatikan bahwa $d \,.\, e \equiv 1 \pmod{\phi(n)}$ ekivalen dengan $d \,.\, e = 1 + m\phi(n)$, sehingga *SK* dapat dihitung dengan

$$d = \frac{1 + m\phi(n)}{e}$$

2.4 Proses Enkripsi

Langkah-langkah pada proses enkripsi adalah sebagai berikut :

1. *Plaintext* diubah ke dalam bentuk bilangan. Untuk mengubah *plaintext* yang berupa huruf menjadi bilangan dapat digunakan kode *ASCII* dalam sistem bilangan desimal.
2. *Plaintext* m dinyatakan menjadi blok-blok $m_1,m_2,m_3,....$, sedemikian sehingga setiap blok merepresentasikan nilai di dalam selang [0, n-1] , sehingga transformasinya menjadi satu ke satu.
3. Setiap blok m_i dienkripsi menjadi blok c_i dengan rumus $m_i = c_i\, e \bmod n$

2.5 Proses Dekripsi

Langkah-langkah pada proses dekripsi adalah sebagai berikut :

1. Setiap blok *ciphertext* c_i didekripsi kembali menjadi blok m_i dengan rumus $m_i = c_i{}^d \bmod n$
2. Kemudian blok-blok $m_1,m_2,m_3,....,$ diubah kembali ke bentuk huruf dengan melihat kode *ASCII* hasil dekripsi.

Keamanan algoritma RSA terletak pada tingkat kesulitan dalam memfaktorkan bilangan non prima menjadi faktor primanya, yang dalam hal ini $n = p \times q$. Sekali r berhasil difaktorkan menjadi p dan q, maka $\phi(n) = (p - 1)\,(q - 1)$ dapat dihitung. Selanjutnya, karena kunci enkrispi e diumumkan (tidak rahasia), maka kunci dekripsi d dapat dihitung dari persamaan $e \,.\, d \equiv 1 \pmod{\phi(n)}$.

Penemu algoritma RSA menyarankan nilai p dan q panjangnya lebih dari 100 digit. Dengan demikian hasil kali $n = p \times q$ akan berukuran lebih dari 200 digit. Menurut Rivest dan kawan-kawan, uasaha untuk mencari faktor bilangan 200 digit membutuhkan waktu komputasi selama 4 milyar tahun! (dengan asumsi bahwa algoritma pemfaktoran yang digunakan adalah algoritma yang tercepat saat ini dan komputer yang dipakai mempunyai kecepatan 1 milidetik).

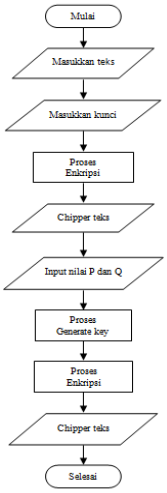
Untunglah algoritma yang paling mangkus untuk memfaktorkan bilangan yang besar belum ditemukan. Inilah yang membuat algoritma RSA tetap dipakai hingga saat ini. Selagi belum ditemukan algoritma yang mangkus untuk memfaktorkan bilangan bulat menjadi faktor primanya, maka algoritma RSA tetap direkomendasikan untuk menyandikan pesan.

III. HASIL DAN PENGUJIAN

Program aplikasi ini dibuat untuk memberikan sistem baru yang lebih aman untuk mengamankan pesan yang penting secara lebih aman.

Dalam menjalankannya ada beberapa tahapan seperti menenkripsikan pesan supaya isi pesan yang akan disampaikan menjadi samar dan selanjutnya mendeskripsikannya supaya isi pesan bisa dikatahui menggunakan kunci yang telah dibuat pada proses deskripsi.

3.1 Proses Enkripsi dan Dekripsi



Gambar. 1 Flowchart Sistem (Enkripsi dan Deskrisi)

Adapun proses enkripsi pesan rahasia diantaranya sebagai berikut:

a. User harus terlebih dahulu mengisi pesan atau plainteks yang akan dikirim pada aplikasi keamanan pesan yang telah dibuat
b. Setelah itu user harus memasukkan kunci rahasia yang akan di gunakan untuk proses enkripsi
c. User harus menekan tombol enkripsi pada aplikasi keamanan pesan agar pesan terenkripsi.
d. Setelah pesan terenkripsi, maka user menekan tombol generatekey untuk mendapatkan nilai n, kunci publik, dan kunci privat secara otomatis.
e. User menekan tombol enkripsi, pesan yang teah terenkripsi disimpan dengan format .txt, barulah pesan kemudian dikirmkan ke penerima.

Setelah pesan diterima maka pesan harus didekripsikan oleh penerima pesan agar pesan yang diterima dapat dibaca. Adapun prosesnya sebagai berikut:

a. User harus mencari file yang telah di terima dengan format .txt, maka memasukkan nilai n, kunci publik dan kunci privat.
b. User harus menekan tombol dekripsi agar pesan kembali ke pesan awal sebelum dienkripsi.

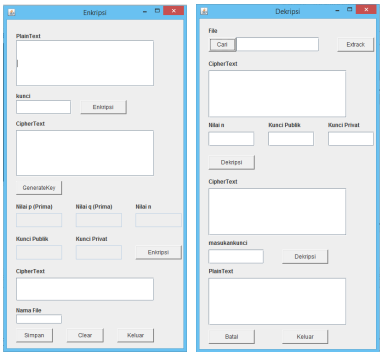
c. Masukkan kunci rahasia yang dibuat, lalu tekan tombol dekripsi untuk melihat pesan yang telah diterima tersebut.

Keterangan : Semua proses pembentukan generatekey secara otomatis.

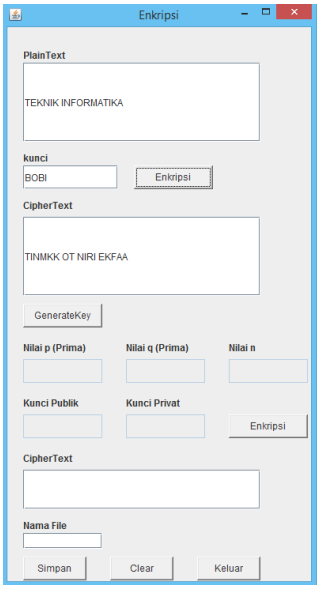
3.2 Pengujian Aplikasi

Adapun hasil pengujian yang telah dilakukan terlihat bahwa, aplikasi dapat menjalankan fitur-fitur yang ada. Hal ini ditunjukkan dengan berhasilnya penggunaan uji coba fitur-fitur yang dibuat. Hasil pengujian dapat dilihat pada gambar di bawah ini.

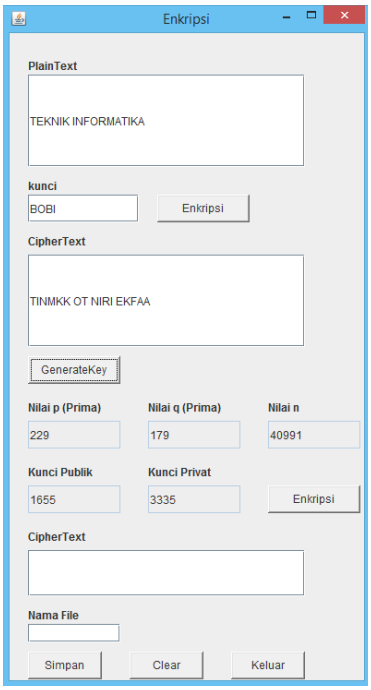
Gambar 1. Halaman Awal Aplikasi



Gambar 2. Tampilan Enkripsi dan Dekripsi

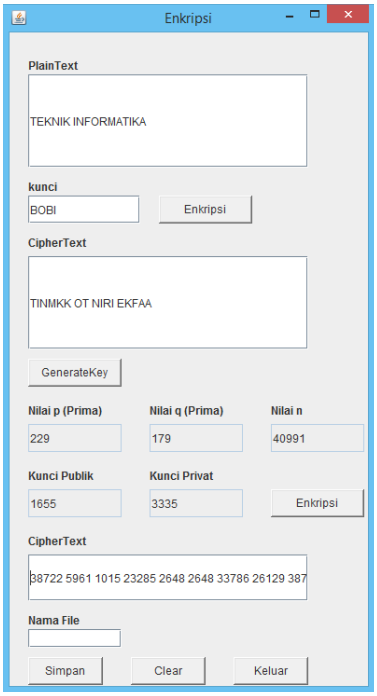


Gambar 3. Memasukkan Plainteks dan Kunci

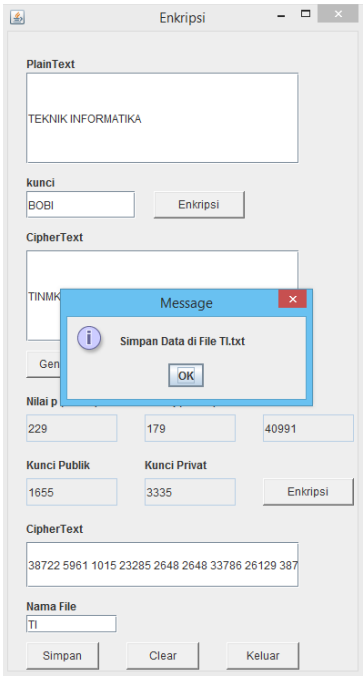


Gambar 4. Membuat Generatekey

Pada proses ini bisa dilihat, bahwa nilai generatekey akan otomatis terbentuk dan selanjutnya akan dienkripsikan sehingga tebentuknya chipertext. Bisa dilihat pada gambar berikut.



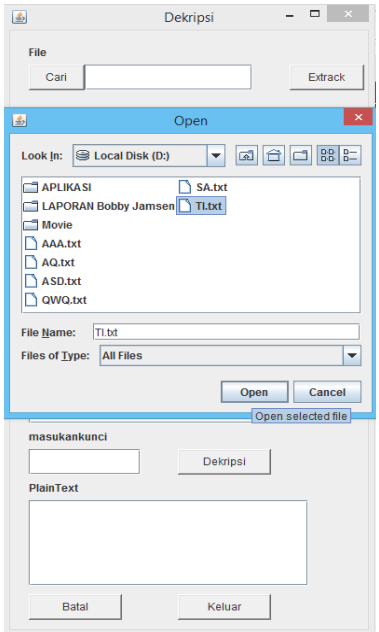
Gambar 5. Hasil Proses Enkripsi



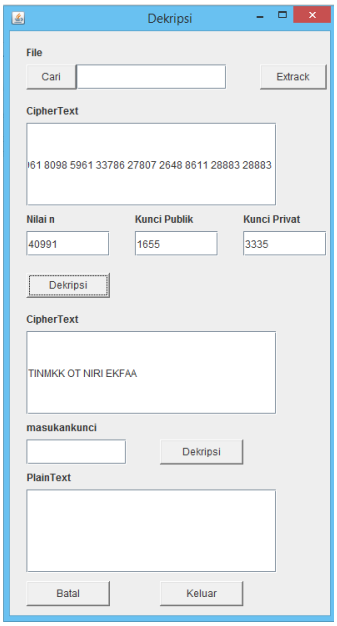
Gambar 6. Menyimpan Hasil Proses Enkripsi

Dari proses diatas telah diketahui bahwa proses Enkripsi telah selesai dijalankan dan selanjutnya file .txt yang telah disimpan akan dikirim kepada penerima yang telah dituju.

Adapun proses Dekripsi dari file yang telah diterima dari pengirim pesan dapat dilihat pada gambar di bawah ini.

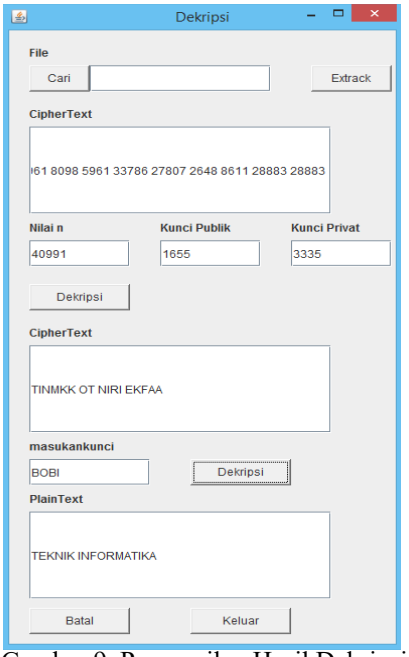


Gambar 7. Mencari file .txt yang telah diterima



Gambar 8. Memasukkan Generatekey dan mendekripsikannya

Proses ini bisa dilihat, bahwa generatekey yang telah dimasukkan akan menghasilkan chipertext seperti pada gambar 8. Untuk selanjutnya penerima pesan akan memasukkan kata kunci, untuk mengetahui isi dari pesan tersebut.



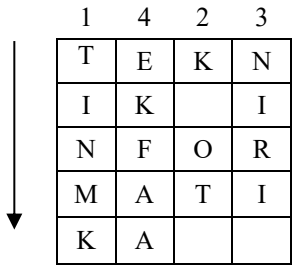
Gambar 9. Penampilan Hasil Dekripsi

3.3 Pembuktian Enkripsi dan Dekripsi

a. Pengujian Transposisi columnar

Pada gambar diatas, telah diketahui :

Plainteks : **TEKNIK INFORMATIKA**
Kunci : **BOBI**
Jawab :



1	4	2	3
T	E	K	N
I	K		I
N	F	O	R
M	A	T	I
K	A		

Dari tabel diatas, penentuan chiperteks dihitung berdasarkan kunci. Maka didapatkan chiperteks **"TINMKK OT NIRI EKFAA"**

b. Pembuktian RSA

Diketahuai dari plainteks didapatkan chiperteks "**TINMKK OT NIRI EKFAA**". Dengan nilai ***p* = 229**, ***q* = 179**, ***n* = *40991***, ***e* = *1655*, *d* = *3335***, yang telah diketahui generatekey didapat secara otomatis pada aplikasi.

Untuk mendapatkan chiperteks 2, maka akan dibuktikan dari perhitungan berikut :

Jawab :

p* = 229** ***q* = 179** ***n* = *40.991
e* = *1.655 ***d* = *3.335***

$n = p \,.\, q$
$= 229 \,.\, 179$
$= 40.991$

$\phi(n) = (p - 1)(q - 1)$
$= (229 - 1)(179 - 1)$
$= (228)(178)$
$= 40.584$

Sebelum pembuktian nilai *e* dan *d*, maka kita harus mencari nilai *m* terlebih dahulu.

$$d = \frac{1 + m\phi(n)}{e}$$
$$3.335 = \frac{1 + m\ 40.584}{1.655}$$
$$(3.335)(1.655) = 1 + 40.584\ m$$
$$5.519.425 - 1 = 40.584\ m$$
$$5.519.424 = 40.584\ m$$
$$40.584\ m = 5.519.424$$
$$m = \frac{5.519.424}{40.584}$$
$$m = 136$$

Setelah nilai *m* didapatkan, selanjutnya pembuktian darimanakah nilai *d* diperoleh.

$$d = \frac{1 + m\phi(n)}{e}$$
$$= \frac{1 + (136)(40.584)}{1.655}$$
$$= \frac{1 + 5.519.524}{1.655}$$
$$= \frac{5.519.524}{1.655}$$
$$d = 3.335$$

Selanjutnya pembuktian diperolehnya nilai *e*.

$$d = \frac{1 + m\phi(n)}{e}$$
$$3.335 = \frac{1 + (136)(40.584)}{e}$$
$$3.335 = \frac{1 + 5.519.424}{e}$$
$$3.335 = \frac{5.519.425}{e}$$
$$3.335e = 5.519.425$$
$$e = \frac{5.519.425}{3.335}$$
$$e = 1.655$$

Dari keseluruhan semua nilai telah dibuktikan cara mendapatkannya. Selanjutnya, pembuktian cara mendapatkan nilai chiperteks yang telah diekripsikan :

"38722 5961 1015 23285 2648 2648 33786 26129 38722 33786 1015 5961 8098 5961 33786 27807 2648 8611 28883 28883".

Jawab :

Plaintext diubah ke dalam bentuk bilangan *ASCII* dalam sistem bilangan desimal.

$T \quad m_1 = 84$
$= 84^{1655} mod\ 40991$
$= 38722$

$I \quad m_2 = 7$
$= 73^{1655} mod\ 40991$
$= 5961$

$N \quad m_3 = 78$
$= 78^{1655} mod\ 40991$
$= 1015$

$M \quad m_4 = 77$
$= 77^{1655} mod\ 40991$
$= 23285$

$K \quad m_5 = 75$
$= 75^{1655} mod\ 40991$
$= 2648$

$K \quad m_6 = 75$
$= 75^{1655} mod\ 40991$
$= 2648$

$SP \quad m_7 = 32$
$= 32^{1655} mod\ 40991$
$= 33786$

$O \quad m_8 = 79$
$= 79^{1655} mod\ 40991$
$= 26129$

$T \quad m_9 = 84$
$= 84^{1655} mod\ 40991$
$= 38722$

$SP \quad m_{10} = 32$
$= 32^{1655} mod\ 40991$
$= 33786$

$N \quad m_{11} = 78$
$= 78^{1655} mod\ 40991$
$= 1015$

$I \quad m_{12} = 73$
$= 73^{1655} mod\ 40991$
$= 5961$

$R \quad m_{13} = 82$
$= 82^{1655} mod\ 40991$
$= 8098$

$I \quad m_{14} = 73$
$= 73^{1655} mod\ 40991$
$= 5961$

$SP \quad m_{15} = 32$
$= 32^{1655} mod\ 40991$
$= 33786$

$E \quad m_{16} = 69$
$= 69^{1655} mod\ 40991$
$= 27807$

$K \quad m_{17} = 75$
$= 75^{1655} mod\ 40991$
$= 2648$

$F \quad m_{18} = 70$
$= 70^{1655} mod\ 40991$
$= 8611$

$A \quad m_{19} = 65$
$= 65^{1655} mod\ 40991$
$= 28883$

$A \quad m_{20} = 65$
$= 65^{1655} mod\ 40991$
$= 28883$

IV. KESIMPULAN

Dari hasil penelitian dapat disimpulkan bahwa, dalam pembentukan pesan rahasia dari penggabungan kriptografi transposisi columnar dan RSA sangat efektif dalam mengamankan suatu pesan rahasia. Ada beberapa proses pembentukan pesan rahasia seperti :

1. Dalam proses pembentukan plainteks (isi pesan) akan diacak dengan posisi matrik yang berbeda.
2. Proses pembentukan generatekey dibuat secara otomatis untuk mempermudah penggunaannya, pembentukan kunci untuk memperoleh pasangan kunci publik dan kunci rahasia kemudian proses enkripsi dan proses dekripsi terhadap pesan/ informasi yang akan ditransmisikan
3. Tanpa kunci pesan tersebut tidak akan mudah untuk dibuka oleh yang tidak berkepentingan pada pesan tersebut.

Dengan kata lain, penggabungan kedua algoritma untuk mengamankan pesan sangatlah efektif, sehingga dapat dipastikan pesan yang ada tidak akan diperoleh dan diketahui dengan mudah oleh orang-orang yang tidak mempunyai kepentingan untuk itu.

Daftar Pustaka

[1] Kester. 2013. *A Hybrid Cryptosystem Based on Vigenere Cipher and Columnar Transposition Cipher. International Journal of Advanced Technology & Engineering Research (IJATER)* Vol. 3(1): 141-147.

[2] Weiman, Donald. 2012. *ASCII Conversion Chart.doc Copyright © 2008, 22 March 2012*. http://creativecommons.org/licenses/by-sa/3.0/

[3] Gaines, H.F. 1956. *Cryptanalysis: A Study of Cipher and Their Solution*. Dover Publications Inc. New York.

[4] Sinkov,A. 1966. *Elementary Cryptanalysis: A Mathematical Approach*. International and Pan-American Copyright Conventions. Wahington D.C.

[5] Nathasia, N. D. , & Wicaksono, A. E. (2011). *Penggunaan Teknik Kriptografi Stream Cipher untuk Pengamanan Basis Data*. Jurnal Basis Data, ICT Research Center UNAS, 6(1), 1-22.

[6] Sukrisno, & Utami, E. (2007). *Implementasi Steganografi Teknik EOF Dengan Gabungan Enkripsi Rijndael, Shift Cipher Dan Fungsi Hash MD5*. Seminar Nasional Teknologi 2007 (SNT 2007), (November), 1-16.

[7] Ginting, Albert, R. Rizal Isnanto & Ike Pertiwi Windasari. *Implementasi Algoritma Kriptografi RSA untuk Enkripsi dan Dekripsi Email. Jurnal Teknologi dan Sistem Komputer*, Vol.3(2): 254-255, , April 2015 (e-ISSN: 2338-0403).