APLIKASI ENKRIPSI DATA PADA FILE TEKS DENGAN ALGORITMA RSA (*RIVEST SHAMIR ADLEMAN*)

Hendri Syaputra[1], Hendrik Fery Herdiyatmoko[2]
[1,2]*Teknik Informatika, Sekolah Tinggi Teknik Musi, Palembang 30113*
E-mail : hendri_121@yahoo.com ,hendrik_023@yahoo.co.id

ABSTRAK

Aplikasi ini ditujukan untuk membantu melindungi isi pesan atau data serta informasi yang rahasia. Algoritma RSA (*RIVEST SHAMIR ADLEMAN*) adalah salah satu algoritma kriptografi yang cukup baik, karena sulitnya memfaktorkan bilangan yang besar menjadi factor-faktor prima. Aplikasi RSA dikembangkan sesuai dengan kebutuhan zaman sekarang yang simple dan efisien. Aplikasi ini dalam pengembangannya melibatkan *platform* VB 6.0. Analisis dan kebutuhan aplikasi bagi pengguna nantinya dikembangkan dengan metode *Water Fall* dengan bahasa permodelan UML (*Unified Modelling Language*) sehingga aplikasi ini dapat menyediakan fitur enkripsi, dan dekripsi. Hasil penelitian ini adalah melindungi pesan dengan mengubah pesan asli menjadi sebuah kode rahasia yang berupa angka-angka dan dapat mengubah kembali menjadi pesan aslinya.

Kata Kunci : Kriptografi, Enkripsi, Dekripsi, Algoritma RSA, teks

1. PENDAHULUAN

Masalah keamanan dan kerahasiaan merupakan salah satu aspek penting dari suatu sistem, pesan, data atau informasi. Masalah keamanan seringkali kurang mendapat perhatian dari para perancang dan pengelola sitem informasi. Seringkali masalah keamanan berada di urutan setelah tampilan, atau bahkan di urutan terakhir dalam daftar hal-hal yang dianggap penting. Dalam hal ini sangat terkait dengan betapa pentingnya pesan, data, atau informasi tersebut oleh pihak atau orang yang berkepentingan, apakah pesan, data, atau informasi masih authentifikasi. Pesan, data, atau informasi akan tidak menjadi rahasia lagi apabila di tengah jalan informasi itu di akses oleh orang yang tidak berhak atau berkepentingan.

Kriptografi merupakan seni dan ilmu untuk menjaga keamanan data. Dalam menjaga keamanan data, kriptografi mentransformasikan data jelas (*plaintext*) ke dalam bentuk data sandi (*ciphertext*) yang tidak dapat dikenali. *Ciphertext* inilah yang kemudian dikirimkan oleh pengirim (*sender*) kepada penerima (*receiver*). Setelah sampai di penerima, *ciphertext* tersebut ditranformasikan kembali ke dalam bentuk *plaintext* agar dapat dikenali [1].

Algoritma kriptografi dibagi menjadi tiga bagian berdasarkan dari kunci yang dipakainya. Algoritma Simetri, menggunakan satu kunci untuk enkripsi dan deskripsinya. Algoritma Asimetri, mengunakan kunci yang berbeda untuk enkripsi dan dekripsinya. Algortima terakhir adalah *Hash Function*. Algoritma Asimetri mempunyai beberapa beberapa macam algoritma antara lain RSA, DSA, Diffie-Helman [2].

RSA adalah salah satu model dan metode enkripsi. Keamanan enkripsi dari RSA cukup baik, hal ini terjadi karena sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima [3]. RSA banyak diaplikasikan untuk mengenkripsi teks. Teks merupakan data yang penting dan paling sering digunakan, apalagi jika teks tersebut berisi rahasia penting suatu perusahaan maupun rahasia pribadi seseorang. Data teks sering menjadi sasaran kejahatan. Sebagai contoh, teks yang berisi privasi seseorang diambil kemudian di-publishkan ke internet oleh orang yang tidak berhak. Maka untuk melindungi teks tersebut dari orang-orang yang tidak berhak, akan dibuat sebuah aplikasi kriptografi file teks menggunakan algoritma RSA.

2. TINJAUAN PUSTAKA

2.1 Kriptografi

Kriptografi (*cryptography*) berasal dari bahasa Yunani yaitu "*cryptos*" artinya "*secret*" (rahasia), sedangkan "*graphein*" artinya "*writing*" (tulisan). Jadi, kriptografi berarti "*secret writing*" (tulisan rahasia). Ada beberapa definisi kriptografi yang telah dikemukakan di dalam berbagai literatur. Definisi yang dipakai di dalam buku-buku lama (sebelum tahun 1980-an) menyatakan bahwa kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya. Definisi ini mungkin cocok pada masa lalu dimana kriptografi digunakan untuk keamanan komunikasi penting seperti komunikasi di kalangan militer, diplomat, dan

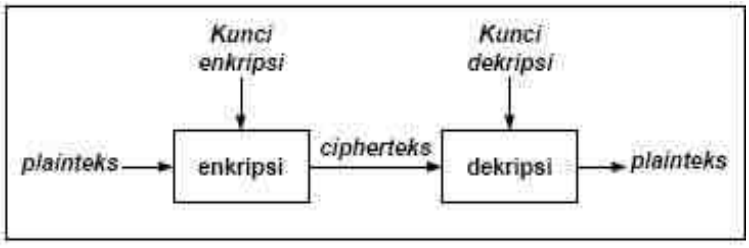
mata-mata. Namun saat ini kriptografi lebih dari sekedar *privacy*, tetapi juga untuk tujuan *data integrity*, *authentication*, dan *non-repudiation* [3].

Kriptografi berasal dari bahasa yunani, menurut bahasa dibagi menjadi kripto dan graphia, kripto berarti *secret* (rahasia) dan graphia berarti writing (tulisan). Menurut terminologinya kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ketempat yang lain [2].

Di dalam kriptografi terdapat berbagai istilah atau terminologi. Beberapa istilah yang penting untuk diketahui antara lain:

a. Pesan, Plainteks, dan chiperteks
   Pesan (*message*) adalah data atau informasi yang dapat dibaca atau dimengerti maknanya. Nama lain untuk pesan adalah plainteks (*plaintext*) atau teks-jelas (*cleartext*). Pesan dapat berupa data atau informasi yang dikirim (melalui kurir, saluran telekomunikasi, dsb) atau yang disimpan di dalam media perekaman (kertas, *storage*, dsb). Pesan yang disimpan tidak hanya berupa teks, tetapi juga dapat berbentuk citra (*image*), suara/bunyi (*voice*), dan video.
   Agar pesan tidak dapat dimengerti maknanya oleh pihak lain, maka pesan perlu disandikan ke bentuk lain yang tidak dapat dipahami. Bentuk pesan yang tersandi disebut cipherteks (*ciphertext*) atau kriptogram (*cryptogram*). Cipherteks harus dapat ditransformasikan kembali menjadi plainteks semula agar pesan yang diterima bisa dibaca.
b. Pengirim dan penerima
   Komunikasi data melibatkan pertukaran pesan antara dua entitas. Pengirim (sender) adalah entitas yang mengirim pesan kepada entitas lainnya. Penerima (receiver) adalah entitas yang menerima pesan. Entitas disini dapat berupa orang, kertu kredit, dan sebagainya. Pengirim tentu menginginkan pesan dapat dikirim secara aman, yaitu ia yakin bahwa pihak lain tidak dapat membaca isi pesan yang ia kirim. Solusinya adalah dengan cara menyandikan pesan menjadi cipherteks.
c. Enkripsi dan dekripsi
   Proses menyandikan plainteks menjadi cipherteks disebut enkripsi (encryption). Sedangkan proses mengembalikan cipherteks menjadi plainteks dinamakan dekripsi (decryption). Enkripsi dan dekripsi dapat diterapkan baik pada pesan yang dikirim maupun pesan yang tersimpan.
d. Cipher dan kunci
   Algoritma kriptografi disebut juga cipher yaitu aturan untuk enkripsi dan dekripsi, atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Beberapa cipher memerlukan algoritma yang berbeda untuk enkripsi dan dekripsi.

   Konsep matematis yang mendasari algoritma kriptografi adalah relasi antara dua buah himpunan yang berisi elemen-elemen plainteks dan himpunan yang berisi cipherteks. Enkripsi dan dekripsi merupakan fungsi yang memetakan elemen-elemen antara kedua himpunan tersebut. Misalkan P menyatakan plainteks dan C menyatakan chiperteks, maka fungsi enkripsi E memetakan P ke C,E(P) = C. Dan fungsi dekripsi D memetakan C ke P, D(C) = P
   Karena proses enkripsi kemudian dekripsi mengembalikan pesan ke pesan asal, maka kesamaan berikut harus benar,
   D(E(P)) = P
   Kunci (key) adalah parameter yang digunakan untuk transformasi enkripsi dan dekripsi. Kunci biasanya berupa string atau deretan bilangan. Dengan menggunakan kunci K, maka fungsi enkripsi dan dekripsi dapat ditulis sebagai Ek(P) = C dan Dk(C) = P
   Dan kedua fungsi ini memenuhi
   Dk(Ek)(P)) = P



Gambar 1 Skema Enkripsi dan Dekripsi

e. Sistem kriptografi
   Kriptografi membentuk sebuah sistem yang dinamakan sistem kriptografi. Sistem kriptografi (*cryptosystem*) adalah kumpulan yang terdiri dari algoritma kriptografi, semua plainteks dan cipherteks yang mungkin, dan kunci. Di dalam sistem kriptografi, *cipher* hanyalah salah satu komponen saja.

f. Penyadap
Penyadap (eavesdropper) adalah orang yang mencoba menangkap pesan selama ditransmisikan. Tujuan penyadap adalah untuk mendapatkan informasi sebanyak-banyaknya mengenai sistem kriptografi yang digunakan untuk berkomunikasi dengan maksud untuk memecahkan cipherteks. Nama lain penyadap: enemy, adversary, intruder, interceptor, bad guy. Ron Rivest, seorang pakar kriptografi, menyatakan bahwa cryptography is about communication in the presence of adversaries (kriptografi adalah perihal berkomunikasi dengan keberadaan pihak musuh).

g. Kriptanalisis dan kriptologi
Kriptografi berkembang sedemikian rupa sehingga melahirkan bidang yang berlawanan yaitu kriptanalisis. Kriptanalisis (cryptanalysis) adalah ilmu dan seni untuk memecahkan chiperteks menjadi plainteks tanpa mengetahui kunci yang digunakan. Pelakunya disebut kriptanalis. Jika seorang kriptografer (cryptographer) mentransformasikan plainteks menjadi cipherteks dengan suatu algoritma dan kunci maka sebaliknya seorang kriptanalis berusaha untuk memecahkan cipherteks tersebut untuk menemukan plainteks atau kunci. Kriptologi (cryptology) adalah studi mengenai kriptografi dan kriptanalisis. Baik kriptografi maupun kriptanalisis keduanya saling berkaitan [3].

2.2 Algoritma RSA

Algoritma RSA diuat oleh 3 orang peneliti dari MIT (*Massachussets Instittute of Technology*) pada tahun 1976, yaitu Ron (R)ivest, Adi (S)hamir, dan Leonard (A)dleman. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi factor-faktor prima. Pemfaktoran dilakukan unutk memperoleh kunci privat. Selama pemfaktoran bilangan besar menjadi factor-faktor prima belum ditemukan algoritma yang mangkus, maka selama itu pula keamanan algoritma RSA tetap terjamin [4].

Algortima RSA memilki besaran-besaran sebagai berikut [5] :

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 (chiperteks) (tidak rahasia)

Algoritma RSA berdasarkan pada teorema Euler yang menyatakan bahwa

$$a^{\phi(n)} \equiv 1 \pmod{n} \quad \ldots\ldots (1)$$

dengan syarat :

1. a harus relative prima terhadap n
2. $\phi(n) = n(1 - 1/p1)(1 - 1/p2) \ldots (1 - 1/pr)$, yang dalam hal ini p1, p2, …, pr adalah faktor prima dari n. $\phi(n)$ adalah fungsi yang menentukan berapa banyak dari bilangan – bilangan 1, 2, 3,…, n yang relatf prima terhadap n.

Berdasarkan sifat $a^k \equiv b^k \pmod{n}$ untuk k bilangan bulat ≥ 1, maka persamaan (2.1) dapat ditulis menjadi

$$a^{k\phi(n)} \equiv 1^k \pmod{n} \quad \ldots\ldots (2)$$

atau

$$a^{k\phi(n)} \equiv 1 \pmod{n}$$

Bila a diganti dengan m , maka persamaan (2.2) dapat di tulis menjadi

$$m^{k\phi(n)} \equiv 1^k \pmod{n} \quad \ldots\ldots (3)$$

Berdasarkan sifat $ac \equiv bc \pmod{n}$, maka bila persamaan (2.3) dikali dengan m menjadi

$$m^{k\phi(n)+1} \equiv m \pmod{n} \quad \ldots\ldots (4)$$

Yang dalam hal ini m reratif prima terhadap n.
Misalkan e dan d dipilih sedemikian sehingga

$$e \cdot d \equiv 1 \pmod{\phi(n)} \quad \ldots\ldots (5)$$

atau

$$e \cdot d \equiv k\phi(n) + 1 \quad \ldots\ldots (6)$$

Subtitusikan (2.6) ke dalam persamaan (2.4) menjadi

$$m^{e \cdot d} \equiv m \pmod{n} \quad \ldots\ldots (7)$$

Persamaan (2.7) dapat ditulis kembali menjadi

$$(m^e)^d \equiv m \pmod{n} \quad \ldots\ldots (8)$$

Yang artinya, perpangkatan m dengan e diikuti dengan perpangkatan dengan d menghasilkan kembali m semula. Berdasarkan persamaan (2.8), maka enkripsi dan sekripsi dirumuskan sebagai berikut:

$E_e(m) = c \equiv m^e \bmod n$ ..... (9)

$D_d(m) = m \equiv c^d \bmod n$ ..... (10)

Algortima membangkitkan pasangan kunci

1. Pilih dua buah bilangan prima sembarang, p dan q.
2. Hitung n = p . q (sebaiknya $p \neq q$, sebab jika p = q maka $n= p^2$ sehingga p dapat diperoleh dengan numeric akar pangkat dua dari n).
3. Hitung $\phi(n) = (p - 1)(q – 1)$.
4. Pilih kunci public, e , yang relative prima terhadap $\phi(n)$.
5. Bangkitkan kunci pruvat dengan mengunakan persamaan $e . d \equiv 1 \pmod{\phi(n)}$. Perhatikan bahwa $e . d \equiv 1 \pmod{\phi(n)}$ ekivalen dengan $e . d = 1 + k\phi(n)$, sehingga secara sederhana d dapat dihitung dengan

$d = (1 + k\phi(n)) / e$..... (11)

2.3 Aritmetika Modulo

Misalkan a dalah bilanagn bulat dan m adalah bilangan bulat > 0. Operasi a mod m (dibaca “a modulo m”) memebrikan sisa jika a dibagi dengan m. Bilangan m disebut **modulus** atau **modulo**, dan hasil aritmetika modulo m terletak di dalam himpunan {0, 1, 2, ..., m – 1} [3].

Notasi: a mod m = r sedemikian sehingga a = mq + r, dengan $0 \leq r < m$. Beberapa contoh operasi dengan operator modulo :

(i) 23 mod 5 = 3 (23 = 5 . 4 + 3)
(ii) 27 mod 3 = 0 (27 = 3 . 9 + 0)

2.4 Fungsi Totient Ueler ϕ

Fungsi Totient Euler ϕ mendefinisikan $\phi(n)$ untuk $n \geq 1$ yang menyatakan jumlah bilangan bulat positif < n yang relatif prima dari n.

Contoh :

ϕ (20) = 8; Perhitungannya adalah sbb: bilangan bulat positif yang lebih kecil dari 20 adalah 1 sampai 19. Di antara bilangan-bilangan tersebut, terdapat ϕ (20) = 8 buah yang relative prima dengan 20 yaitu 1, 3, 7, 9, 11, 13, 17, 19.

Untuk n = 1, 2, ..., 10, fungsi Euler adalah

$\phi(1) = 0$ ϕ (6) = 2
$\phi(2) = 1$ ϕ (7) = 6
$\phi(3) = 2$ ϕ (8) = 4
$\phi(4) = 2$ ϕ (9) = 6
$\phi(5) = 4$ ϕ (10) = 4

Jika n prima, maka setiap bilangan bulat yang lebih kecil dari n relatif prima terhadapa n. Dengan kata lain, ϕ (n) = n – 1 hanya jika n prima.

Contoh :

ϕ (3) = 2, ϕ (5) = 4, ϕ (7) = 6, ϕ (11) = 10, ϕ (13) = 12, dst.

Jika n = pq adalah bilangan komposit dengan p dan q prima, maka ϕ (n) = ϕ (p) ϕ (q) = (p – 1)(q – 1).

Contoh :

Tetukan ϕ (21).

Penyelesaian:

Karena 21 = 7 . 3, ϕ (21) = ϕ (7) ϕ (3) = 6 . 2 = 12 buah bilangan bulat yang relatif prima terhadap 21, yaitu 1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20.

Jika p bilangan prima dan k > 0, maka ϕ $(p^k) = p^k – p^{k-1} = p^{k-1}(p – 1)$.

Contoh :

Tentukan ϕ (16).

Penyelesaian:

Karena ϕ (16) = ϕ (2^4) = $2^4 - 2^3 = 16 - 8 = 8$, maka ada delapan buah bilanagan bulat yang relatif prima terhadap 16, yaitu 1, 3, 5, 7, 9, 11, 13, 15.

(*Euler's generalization of Fermat theorem*). Jika PBB (a, n) = 1, maka $a^{\phi(n)} \bmod n = 1$ atau $a^{\phi(n)} \equiv 1 \pmod n$.

2.3 ASCII

ASCII (*American Standard Code for Information Interchange*) merupakan standar internasional dalam kode huruf dan simbol seperti *hex* dan *unicode* tetapi ASCII bersifat universal. Contohnya 124 untuk karakter l. ASCII selalu digunakan oleh komputer dan alat komunikasi lain untuk menunjukkan teks. Kode ASCII sebenarnya memiliki komposisi biner sebanyak 8 bit. Di mulai dari 0000 0000 sampai 1111 1111. Total kombinasi yang dihasilkan sebanyak 256 dimulai dari kode 0 hingga 255 dalam sistem bilangan desimal.

3. ANALISIS DAN PENGEMBANGAN PERANGKAT LUNAK

Metode Pengembangan perangkat lunak Enkripsi Data Pada File Teks Dengan Algoritma Rsa (*Rivest Shamir Adleman*) menggunakan metode air terjun *(waterfall)* dengan tahapan, *analysis* (analisis), *design* (perancangan), *development* (pembangunan), dan *testing* (pengujian) [6].

1. Analisis dan definisi persyaratan.
   Mekanisme algoritma RSA adalah melakukan pemfaktoran bilangan yang sangat besar. Untuk membangkitkan kedua kunci, yang dipilih dua bilangan prima acak yang besar. Schema yang dikembangkan oleh Rivest, Shamir dan Adleman yang mengekpresikan bahwa *plaintext* dienkripsi menjadi block-block yang setiap block memilki nilai bilangan biner yang diberi symbol "n", *plaintext* block "M" dan *chipertext* block "C". Untuk melakukan enkripsi pesan "M" pesan dibagi ke dalam block – block *numeric* yang lebih kecil dari pada "n" (data biner dengan pangkat terbesar), jika bilangan prima yang panjangnya 200 digit dan dapat menambah beberapa bit 0 dikiri bilangan untuk menjaga agar pesan tetap kurang dari nilai "n".
2. Perancangan sistem dan perangkat lunak.
   Proses perancangan sistem membagi persyaratan dalam sistem perangkat keras atau perangkat lunak. Kegiatan ini menentukan arsitektur secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan deskripsi abstraksi sistem perangkat lunak yang mendasar dan hubungan-hubungannya.
3. Implementasi dan pengujian unit.
   Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian unit melibatkan verifikasi bahwa setiap unit telah memenuhi spesifikasinya.
4. Integrasi dan pengujian sistem.
   Unit program atau program individual diintegrasikan dan diuji sebagai sistem yang lengkap untuk menjamin bahwa persyaratan sistem telah tepenuhi. Setelah pengujian sistem, perangkat lunak dikirim kepada pelanggan.

4. HASIL DAN PEMBAHASAN

Berikut cara kerja algoritma RSA dengan contoh pesan berupa "HARI INI" :

Angka p dan q yang dipilih p = 47 dan q =71

$n = p \cdot q = 3337$

dan

$\phi(n) = (p - 1)(q - 1) = 3320$

Kunci publik yang dipilih e = 79, karena 79 relatif prima dengan 3320. Lalu hitung kunci dekripsi d.

$d = (1 + k \times 3320) / 79$

Dengan mencoba nilai-nilai k = 1,2,3, …, diperoleh nilai d yg bulat adalah 1019 dengan mengunakan k =25.

Kunci publik : (e = 79, n = 3337)

Kunc *private* : (d = 1019, n = 3337)

"HARI INI" dalam sistem desimal (pengkodean ASCII) adalah

M = 7265827332737873

Lalu M dipecah menjadi blok yang lebih kecil, misalnya dipecah menjadi enam blok yang berukuran 3 digit :

M1 = 726	M4 = 273
M2 = 582	M5 = 787
M3 = 733	M4 = 003

Nilai-nilai m_i ini masih terletak di dalam selang [0, 3337 – 1] agar transformasi menjadi satu-ke-satu. Dengan kunci publik e = 79 dan n = 3337 dapat mengenkrispsikan setiap blok *plaintext* sebagai berikut:

$C_1 = 726^{79} \bmod 3337 = 215$; $C_2 = 582^{79} \bmod 3337 = 776$;
$C_3 = 733^{79} \bmod 3337 = 1743$; $C_4 = 273^{79} \bmod 3337 = 933$;
$C_5 = 787^{79} \bmod 3337 = 1731$; $C_6 = 003^{79} \bmod 3337 = 158$;

Jadi, cipherteks yang dihasilkan adalah

C = 215 776 1743 933 1731 158.

Dekripsi dilakukan dengan menggunakan kunci privat d = 1019

$M_1 = 215^{1019} \bmod 3337 = 726$; $M_2 = 776^{1019} \bmod 3337 = 582$;
$M_3 = 1743^{1019} \bmod 3337 = 733$; $M_4 = 933^{1019} \bmod 3337 = 273$;
$M_5 = 1731^{1019} \bmod 3337 = 787$; $M_6 = 158^{1019} \bmod 3337 = 003$;

Blok plainteks yang lain dikembalikan dengan cara yang serupa. Akhirnya diperoleh kembali *plaintext* semula

M = 7265827332737873

yang dalam sistem pengkodean ASCII adalah

M = HARI INI

Jumlah digit pada blok – blok *plaintext* atau M bisa kita tentukan menurut yang kita inginkan, seperti menggunakan 2 digit setiap blok, 3 digit setiap blok, atau 4 digit setiap blok. Tetap dengan ketentuan blok-blok *numeric* yang lebih kecil dari pada n (Ariyus Dony,2006). Pada contoh penelitian ini menggunakan blok *plaintext* atau M dengan 3 digit. Perbedaan antara penggunaan 2 digit, 3 digit, atau 4 digit setiap blok adalah di tingkat kesulitan penghitungannya. Semakin besar digit yang dipangkat oleh e (kunci publik) atau d (kunci private) maka proses penghitungan akan lebih sulit. Keamanan algoritma RSA didasarkan pada sulitnya memfaktorkan bilangan besar menjadi faktor-faktor primanya (Munir Rinaldi, 2006).

5. KESIMPULAN

a. Aplikasi ini hanya bisa mengenkripsi dan mendepkripsikan file teks yang berformat .txt.
b. Aplikasi ini hanya bisa mengenkripsi file teks yang panjang karakternya tidak lebih dari 1000 karakter.
c. Aplikasi ini jika mengenkripsi file teks yang isinya berbaris baris, maka hasil dekripsinya akan menjadi satu baris.

DAFTAR PUSTAKA

[1] Raharjo, Budi. (2004). *Keamanan Sistem Informasi Berbasis Internet, Handbook Keamanan*. Bandung- PT Insan Infonesia & Jakarta - PT NDOCISC.
[2] Ariyus, Dony (2006). Kriptografi, *Keamanan Data dan Komunikasi*. Yogyakarta. Graha Ilmu.
[3] Munir, Rinaldi. (2006). *Kriptografi*. Bandung. Informatika.
[4] Ronald L, Rivers, Adi Shamir, Yael Tauman (2001). *How to Leak a Secret*. Proceedings of the 7th International Conference in the Theory and Aplication of Cryptology and Information Security : Advance in Cryptology.
[5] Stinson, Douglas R (2006),*Chryptography Theory and Practice 3rd Edition.*Chapman & Hall/CRC.
[6] Sommerville, Ian (2003). *Software Engineering, Rekayasa Perangkat Lunak*. Jakarta. Erlangga.