PERANCANGAN APLIKASI KEAMANAN PESAN TEKS DENGAN MENGGUNAKAN ALGORITMA TRIPLE DES

Nurmarlina Siregar

STMIK Budi Darma
Jl. Sisingamangaraja No. 338 Sp. Limun Medan
http : // www.stmik-budidarma.ac.id // Email : Nur.chan84@gmail.com

ABSTRACT

Triple DES (Triple Data Encryption Standard) is one of the symmetrical algorithms on cryptography that is used to secure data in the form of text messages by encoding text messages. The process carried out in encoding the data, namely the encryption process and the decryption process. The triple DES algorithm is an algorithm for the development of the DES (Data Encryption Standard) algorithm. The difference between DES and triple DES lies in the length of the key used. In DES it uses one key that is 56-bit long, while triple DES uses three keys that are 168-bits in length (each of which is 56 bits long).

In triple DES, the three keys used can be mutually independent (K1, K2 and K3) or only two keys are mutually independent and one other key is the same as the first key (K1, K2 and K3 = K1). Because the level of confidentiality of the DES algorithm lies in the length of the key used, the use of the triple DES algorithm is considered safer than the DES algorithm. To facilitate the use of the triple DES algorithm, a triple DES algorithm program was created with computer software tools, namely Microsoft Visual Studio 2008 which can encrypt and decrypt text messages.

Keywords: DES Triple (Triple Data Encryption Standard), DES (Data Encryption Standard), cryptography, encryption, decryption, key

ABSTRAK

Triple DES (Triple Data Encryption Standard) merupakan salah satu algoritma simetris pada kriptografi yang digunakan untuk mengamankan data berupa pesan teks dengan cara menyandikan pesan teks. Proses yang dilakukan dalam penyandian datanya, yaitu proses enkripsi dan proses dekripsi. Algoritma triple DES adalah suatu algoritma pengembangan dari algoritma DES (Data Encryption Standard). Perbedaan DES dengan triple DES terletak pada panjangnya kunci yang digunakan. Pada DES menggunakan satu kunci yang panjangnya 56-bit, sedangkan pada triple DES menggunakan tiga kunci yang panjangnya 168-bit (masing-masing panjangnya 56 bit).

Pada triple DES, tiga kunci yang digunakan bisa bersifat saling bebas (K1, K2 dan K3) atau hanya dua buah kunci yang saling bebas dan satu kunci lainnya sama dengan kunci pertama (K1, K2 dan K3 = K1). Karena tingkat kerahasiaan algoritma DES terletak pada panjangnya kunci yang digunakan, maka penggunaan algoritma triple DES dianggap lebih aman dibandingkan dengan algoritma DES. Untuk memudahkan penggunaan algoritma triple DES, maka dibuat suatu program algoritma triple DES dengan alat bantu software komputer, yaitu Microsoft Visual Studio 2008 yang dapat mengenkripsi dan mendekripsi pesan teks.

Kata kunci : Triple DES (Triple Data Encryption Standard), DES (Data Encryption Standard), kriptografi, enkripsi, dekripsi, kunci.

1. PENDAHULUAN

Pesan teks merupakan kumpulan unsur-unsur penting yang berguna. Tetapi, pesan teks sangat rentan terhadap pencurian ataupun penyadapan. Dan banyak orang yang ingin mendapatkan data dan informasi secara ilegal tanpa melalui prosedur resmi. Oleh karena itu, keamanan pesan teks sangat diperlukan untuk mencegah informasi tersebut sampai pada pihak-pihak lain yang tidak berkepentingan yang mengakibatkan terjadinya kebocoran atau penyalahgunaan informasi. Maka, perlu dirancang suatu sistem keamanan yang berfungsi untuk melindungi sistem informasi tersebut.

Salah satu pengamanan data yang berupa pesan teks dari kriptografi adalah algoritma *DES*. *DES* ditetapkan sebagai *standard* untuk melindungi pesan teks. Tetapi *DES* dianggap sudah tidak aman lagi, karena dengan perangkat keras khususnya kuncinya dapat ditemukan dalam waktu beberapa hari. Kemudian *DES* dikembangkan menjadi *3 DES (Triple DES). 3DES (Triple Data Encryption Standard)* merupakan salah satu algoritma simetris pada kriptografi yang digunakan untuk mengamankan suatu data berupa pesan teks. Proses yang dilakukan dalam penyandian pesan teksnya, yaitu proses enkripsi dan proses dekripsi, prosesnya adalah mengulang algoritma *DES* sebanyak tiga kali, sesuai dengan pemilihan kuncinya dan urutan proses yang dipilih. Algoritma triple DES termasuk kedalam kriptografi modern, karena penyandian modern berorientasi pada mode *bit*.

Berdasarkan uraian latar belakang di atas, terdapat beberapa permasalahan yang menjadi titik utama pembahasan, diantaranya adalah sebagai berikut:

1. Bagaimana proses enkripsi dan dekripsi algoritma *triple DES*?
2. Bagaimana menerapkan algoritma *triple DE*S untuk mengamankan pesan teks?
3. Bagaimana merancang suatu aplikasi yang mampu mengenkripsi dan mendekripsi pesan teks berdasarkan algoritma *triple DES*?

Berdasarkan rumusan masalah yang ada maka dapat diberi batasan-batasan sehingga pembahasannya lebih terarah. Adapun batasan-batasan masalah yang menjadi acuan dalam penelitian ini adalah sebagai berikut:

1. Jumlah karakter plainteks dan cipherteks yang dienkripsi dan didekripsi pada penelitian ini adalah 8 karakter atau 64 bit.
2. Algoritma yang digunakan adalah algoritma *triple DES* 3 kunci dalam bentuk sederhana.
3. Bahasa pemrograman yang digunakan untuk membangun aplikasi enkripsi dan dekripsi pesan teks ini adalah *Microsoft Visual Studio 2008*.
4. Aplikasi yang dirancang hanya untuk enkripsi dan dekripsi pesan teks.
5. Tidak membahas proses pengiriman pesan.

Adapun tujuan dari penelitian ini adalah sebagai berikut:

1. Menjelaskan proses enkripsi dan dekripsi algoritma *triple DES.*
2. Menerapkan algoritma *triple DES* dalam mengamankan pesan teks.
3. Merancang aplikasi enkripsi dan dekripsi pesan teks berdasarkan algoritma *t riple DES*.

2. METODOLOGI PENELITIAN

2.1 Kriptografi

Kriptografi *(cryptography)* berasal dari bahasa Yunani: "*cryptos*" artinya "*secret*" (rahasia), sedangkan "*graphein*" artinya "*writing*" (tulisan). Jadi, kriptografi berarti "*secret writing*" (tulisan rahasia). Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, serta otentikasi (Rinaldi Munir, 2006).

Kriptografi pada awalnya dijabarkan sebagai ilmu yang mempelajari bagaimana menyembunyikan pesan. Namun pada

pengertian modern kriptografi adalah ilmu yang bersandarkan pada teknik matematika untuk berurusan dengan keamanan informasi seperti kerahasiaan, keutuhan data dan otentikasi entitas. Jadi, pengertian kriptografi modern adalah tidak saja berurusan dengan penyembunyian pesan, namun lebih pada sekumpulan teknik yang menyediakan keamanan informasi (Rifki Sadikin, 2012).

2.2 Algoritma *DES*

DES merupakan standar sandi blok dengan kunci simetri yang sudah cukup tua. DES pertama kali diadopsi menjadi standar sandi blok oleh NIST *(National Institute of Standards and Technology)* melalui publikasi FIPS-PUBS 46 *(Federal Information Processing Standard 46)* pada tahun 1977, (NIST, 1995b). Meskipun sudah lama, DES masih dipakai karena cukup aman bagi beberapa jenis serangan.

DES merupakan salah satu contoh sandi Feistel, *Feistel* pada tahun 19973 mengusulkan sebuah struktur sandi blok untuk mendekati kemampuan sandi blok ideal yaitu memilki karakteristik difusi dan konfusi (Rifki Sadikin, 2012).

2.3 Algoritma *Triple DES*

Triple DES atau *3DES* menggunakan *DES* tiga kali. Penggunaan tiga langkah ini penting untuk mencegah *meet-in-the middle attack* sebagaimana pada *double DES*. Bentuk sederhana *Triple DES* (Rinaldi Munir, 2006) adalah:

Enkripsi: $C = E_{k_3}(E_{K_2}(E_{K_1}(P)))$

Dekripsi:

$P = D_{K_1}(D_{K_2}(D_{K_3}(C)))$

Varian ini umum dikenal sebagai mode EEE (untuk enkripsi) karena pada proses enkripsi semuanya menggunakan enkripsi. Untuk menyederhanakan *interoperabiliy* antara *DES* dan *3DES*, maka langkah ditengah (pada proses enkripsi *3DES*) diganti dengan dekripsi (mode *EDE*). Dengan pengubahan ini, maka dibuat beberapa versi *3DES*. Versi pertama *3DES* menggunakan 2 buah kunci, *K*1 dan *K*2 (Rinaldi Munir, 2006).

Enkripsi: $C = E_{K_1}(D_{K_2}(E_{K_1}(P)))$

Dekripsi:

$P = D_{K_1}(E_{K_2}(D_{K_1}(C)))$

Enkripsi DES tunggal dengan kunci *K* dapat dinyatakan sebagai *3DES-EDE* dengan *K*1= *K*2= *K*. Penggunaan enkripsi pada langkah ditengah tidak mempengaruhi keamanan algoritma. Versi kedua adalah menggunakan 3 buah kunci, *K*1, *K*2, dan *K*3 (Rinaldi Munir, 2006).

Enkripsi: $C = E_{K_1}(D_{K_2}(E_{K_3}(P)))$

Dekripsi: $P = D_{K_3}(E_{K_2}(D_{K_1}(C)))$

3. HASIL DAN PEMBAHASAN

3.1 Analisa

Keamanan merupakan aspek yang sangat penting dalam proses pengiriman pesan teks yang sifatnya rahasia. Karena pesan teks sangat rentan terhadap pencurian ataupun penyadapan. Oleh karena itu, keamanan sangat penting untuk mencegah informasi tersebut sampai pada pihak-pihak lain yang tidak berkepentingan yang mengakibatkan terjadinya kebocoran atau penyalahgunaan data dan informasi.

Analisa ini nantinya akan diterapkan dengan melakukan penginputan berupa plainteks, dimana plainteks nantinya dapat diinputkan oleh pengirim. Setelah itu pada plainteks tersebut dilakukan proses enkripsi sebanyak tiga kali enkripsi dengan menggunakan tiga buah kunci yang berbeda. Hasil yang didapatkan dari proses enkripsi dengan menggunakan kunci pertama dienkripsi kembali dengan kunci kedua dan menghasilkan cipherteks yang kedua, kemudian dienkripsi lagi dengan kunci ketiga sehingga menghasilkan cipherteks atau teks sandi. Kemudian pada cipherteks dilakukan proses dekripsi agar penerima pesan mengerti tentang isi pesan yang diterimanya. Inputan yang digunakan dalam proses dekripsi ini adalah cipherteks yang dihasilkan atau didapatkan dari proses enkripsi sebelumnya. Proses dekripsi ini juga dilakukan sebanyak tiga kali enkripsi dan kunci ataupun algoritma yang digunakan adalah kunci ataupun algoritma yang sama pada proses

enkripsinya yaitu cipherteks yang ketiga didekripsi dengan kunci ketiga dan mendapatkan plainteks, setelah itu plainteksnya didekripsi dengan kunci kedua dan mendapatkan plainteks, dan terakhir adalah plainteks yang kedua didekripsi dengan kunci pertama dan mendapatkan plainteksnya atau pesan asli.

3.2 Penerapan Algoritma *Triple DES*

Proses Enkripsi :

Rumus untuk proses enkripsinya adalah :

$L_1 = R_{1-1}$

$R_1 = L_{1-1} \oplus F(R_{1-1}, K_1)$

Misalnya: Plainteksnya= LINAIMUT

Kemudian dienkripsi dengan kunci pertama (K1)= MAHMUDAN

Setelah itu, diproses kedalam 16 putaran:

Untuk hasil putaran pertama adalah sebagai berikut:

$R_1 = L_0 \oplus f[(R]_0, K_1) =$ 1000 0011 0011 1010 0011 1001 0110 1100

Hasil dari putaran pertama menjadi masukan untuk putaran kedua dan hasil putaran kedua menjadi masukan untuk putaran ketiga sampai putaran ke-16. Adapun hasil dari putaran ke-16 yaitu:

R_{16} = 1001 1001 0100 1001 1000 1101 1010 1110

Maka Cipherteks untuk kunci pertama adalah 7E A3 8D 55 E8 23 10 65

Kemudian cipherteks yang pertama dienkripsi kembali dengan kunci kedua (K2) yaitu !! 4 W y › ¼ ß ñ

Untuk proses enkripsinya sama seperti proses enkripsi kunci pertama. Hasil dari proses enkripsi kunci kedua adalah 58 B A1 68 84 25 99 DA

Setelah cipherteks yang kedua didapatkan, kemudian dienkripsi dengan kunci ketiga (K3) yaitu MARLINAN

Proses enkripsinya sama seperti proses enkripsi kunci pertama dan kunci kedua. Hasil proses enkripsi kunci ketiga adalah :

Hexadesimal = A 9A 7E 83 11 F8 AC 7E

Desimal = 10 154 126 131 17 248 172 126

Karakter = 10 š ~ *f* ¬ ~

Proses Dekripsi :

Untuk proses dekripsinya adalah cipherteks yang ketiga didekripsi dengan kunci kunci ketiga (K3), yaitu diproses kedalam 16 putaran sehingga hasil dekripsinya adalah 58 B A1 68 84 25 99 DA

Setelah hasil dekripsi untuk kunci ketiga didapatkan, kemudian didekripsi lagi dengan kunci kedua (K2), proses dekripsi untuk kunci kedua caranya sama seperti proses dekripsi kunci ketiga. Adapun hasil dari dekripsi kunci kedua adalah 7E A3 8D 55 E8 23 10 65

Kemudian hasil dekripsi kunci pertama didekripsi lagi dengan kunci pertama (K1), proses dekripsinya sama seperti proses dekripsi kunci ketiga dan kunci kedua. Adapun hasil dari dekripsi kunci pertama adalah :

Hexadesimal = 4c 49 4E 41 49 4D 55 54

Desimal = 76 73 78 65 73 77 85 84

Karakter = LINAIMUT ➔ Plainteks (pesan asli)

4. IMPLEMENTASI PENELITIAN

4.1 Algoritma

Algoritma merupakan suatu metode yang digunakan untuk menyelesaikan suatu masalah. Algoritma merupakan langkah-langkah untuk merancang program yang dinyatakan dalam bahasa yang dapat dimengerti.

Adapun urutan langkah-langkah dalam enkripsi dan dekripsi adalah sebagai berikut:

1. Algoritma Enkripsi

   Input :

   P : $\leftarrow L_i, \quad R_i$

   Plainteks (teks asli)

   K : K_1 ← Kunci

   Output :

   C_i : $\leftarrow R_i, \quad L_i$

   Cipherteks (teks sandi)

   Proses :

   $L_i = R_{i-1}$

   $R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$

L = Sub-blok kiri (*left*)
R = Sub-blok kanan (*right*)
i = 1, 2, ..., *r*; *r* adalah jumlah putaran
K_i = Kunci pada putaran ke-*i*
f = Fungsi transformasi

$$R_1 = L_{1-1} \oplus F(R_{1-1}, K_1)$$
.
.
.
$$R_{16} = L_{16-1} \oplus F(R_{16-1}, K_{16})$$

Input :
K : K_2 ←——— Kunci
Output :
C_i : ←——— R_i, L_i
Cipherteks (teks sandi)
Proses :

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$$

Input :
K : K_3 ←——— Kunci
Output :
C_i : ←——— R_i, L_i
Cipherteks (teks sandi)
Proses :

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$$

2. Algoritma Dekripsi
Input :
C : ←——— R_i, L_i
Cipherteks (teks sandi)
K_i : K_1, K_2, K_3 ←——— Kunci
Output :
P_i : ←——— L_i, R_i
Plainteks

Proses :

$$R_i = L_{i-1}$$
$$L_i = R_{i-1} \oplus F(R_{i-1}, K_i)$$

L = Sub-blok kiri (*left*)
R = Sub-blok kanan (*right*)
i = 1, 2, ..., *r*; *r* adalah jumlah putaran
K_i = Kunci pada putaran ke-*i*
f = Fungsi transformasi

$$L_1 = R_{1-1} \oplus F(L_{1-1}, K_1)$$
.
.
.
$$L_{16} = R_{16-1} \oplus F(L_{16-1}, K_{16})$$

Input :
K : K_2 ←——— Kunci
Output :
P_i : ←——— L_i, R_i
Plainteks

Proses :

$$R_i = L_{i-1}$$
$$L_i = R_{i-1} \oplus F(R_{i-1}, K_i)$$

Input :
K : K_3 ←——— Kunci
Output :
P_i : ←——— L_i, R_i
Plainteks (teks asli)

Proses :

$$R_i = L_{i-1}$$
$$L_i = R_{i-1} \oplus F(R_{i-1}, K_i)$$

4.2 Implementasi

Implementasi adalah suatu tindakan atau pelaksanaan dari sebuah rencana yang sudah disusun secara terperinci. Implementasi biasanya dilakukan setelah perencanaaan sudah dianggap selesai.

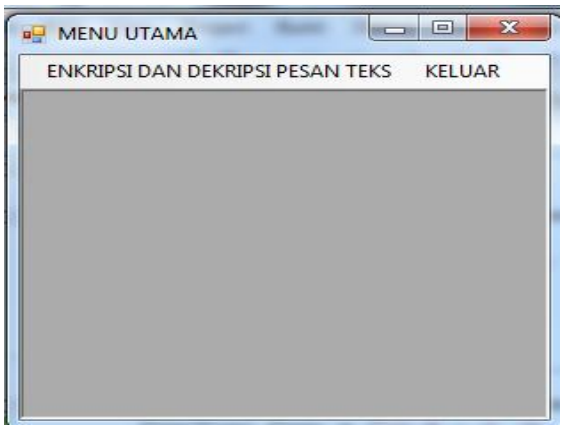
1. *Form Login*

Tampilan *login* merupakan tampilan yang muncul untuk penginputan *user* dan *password*. Tampilan *login* dapat dilihat pada gambar 1

Gambar 1 Form Login

2. Tampilan Menu Utama

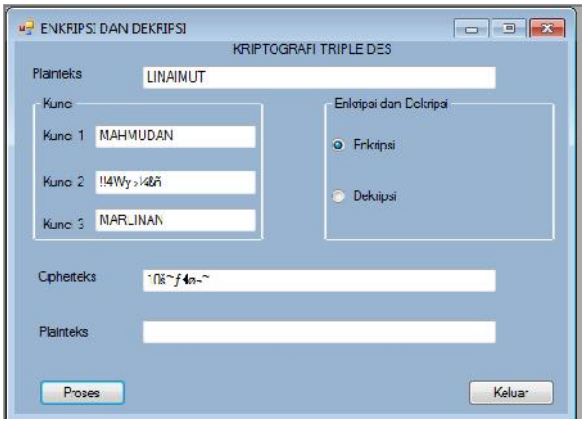
Tampilan menu utama merupakan tampilan yang muncul setelah dilakukan *login* untuk pengamanan teks. Tampilan ini dapat dilihat pada gambar 2



Gambar 2 Tampilan form menu utama

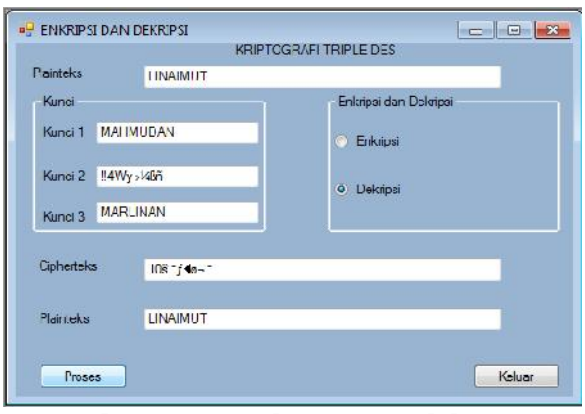
3. Tampilan Form Enkripsi dan Dekripsi Pesan Teks

Tampilan menu enkripsi dan dekripsi merupakan tampilan yang digunakan untuk melakukan proses enkripsi dan proses dekripsi. Untuk tampilan proses enkripsi dapat dilihat pada gambar 3



Gambar 3 Tampilan proses enkripsi

Untuk tampilan dekripsi dapat dilihat pada gambar 4



Gambar 4 Tampilan proses dekripsi

5. KESIMPULAN

Adapun beberapa kesimpulan pada penelitian ini adalah :

1. Proses enkripsi dan dekripsi suatu pesan teks dengan algoritma *triple* DES dilakukan dengan cara mengimplementasikan algoritma DES sebanyak 3 kali, sesuai dengan pemilihan kuncinya dan urutan proses yang dipilih yaitu algoritma *triple* DES sederhana.
2. Dengan diterapkannya algoritma *triple* DES sederhana maka dapat meminimalisir terjadinya penyadapan dan perubahan pesan teks asli yang bersifat rahasia.
3. Program aplikasi ini dapat mengubah pesan teks asli menjadi pesan teks sandi, sehingga mempersulit pihak-pihak yang tidak berkepentingan untuk mengetahui pesan aslinya.

6. SARAN

Beberapa saran yang dapat dijadikan pertimbangan dalam mengembangkan penelitian ini adalah:

1. Penulis menyarankan agar program yang dibuat dapat mengenkripsi dan mendekripsi tidak hanya pada pesan teks saja melainkan berupa gambar, suara, dan video.
2. Diharapkan supaya algoritma *triple* DES sederhana ini digantikan dengan algoritma lain seperti algoritma AES dan RSA, agar isi pesan teks asli lebih sulit diketahui orang-orang yang tidak berkepentingan.
3. Diharapkan juga aplikasi ini bisa dikembangkan lagi dengan menggunakan bahasa pemrograman lain, seperti C++ dan Matlab.

REFERENSI

[1]. Abdul Kadir. (2013). Pengenalan Algoritma. Yogyakarta: Penerbit Andi.

[2]. Abdul Kadir dan Terra Ch. Triwahyuni. (2013). Pengantar

Teknologi Informasi Edisi Revisi. Yogyakarta. Penerbit Andi.

[3]. C. Widyo Hermawan (Ed). (2009). Visual Basic 2008. Yogyakarta. Penerbit Andi.

[4]. Dony Ariyus Dan Rum Andri K.R (2008). Komunikasi Data. Yogyakarta. Penerbit Andi.

[5]. Limbong, T. (2015). Pengujian Kriptografi Klasik Caesar Chipper Menggunakan Matlab. No. September, 2017.

[6]. Limbong, T., & Silitonga, P. D. P. (2017). Testing the Classic Caesar Cipher Cryptography using of Matlab. International Journal of Engineering Research & Technology, 6(2), 175–178. https://doi.org/10.17605/OSF.IO/PEMA5

[7]. Sadikin, Rifki (2012). Kriptografi Untuk Keamanan Jaringan. Yogyakarta. Penerbit Andi.

[8]. Rinaldi Munir. (2006). Kriptografi. Bandung. Penerbit Informatika.

[9]. Soetam Rizky. (2011). Konsep Dasar Rekayasa Perangkat Lunak. Jakarta. PT. Prestasi Pustakaraya.

[10]. Yuni Sugiarti. (2013). Analisis Dan Perancangan Unified Modelling Laguage (UML) Generate VB.6. Yogyakarta. Graha Ilmu.

[11]. Fauzi, Achmad dan Maulita, Yani , *Analisis Hybrid Cryptosystem Algoritma Elgamal Dan Algoritma Triple Des, 2016*