ENKRIPSI DAN DEKRIPSI DATA DENGAN ALGORITMA 3 DES (*TRIPLE DATA ENCRYPTION STANDARD*)

Drs. Akik Hidayat, M.Kom
Jurusan Matematika FMIPA Universitas Padjadjaran
Jl. Raya Bandung-Sumedang km 21 Jatinangor

ABSTRAK

3DES (*Triple Data Encryption Standard*) merupakan salah satu algoritma simetris pada kriptografi yang digunakan untuk mengamankan data dengan cara menyandikan data. Proses yang dilakukan dalam penyandian datanya, yaitu proses enkripsi dan proses dekripsi. Algoritma 3DES adalah suatu algoritma pengembangan dari algoritma DES (*Data Encryption Standard*). Perbedaan DES dengan 3DES terletak pada panjangnya kunci yang digunakan. Pada DES menggunakan satu kunci yang panjangnya 56-bit, sedangkan pada 3DES menggunakan 3 kunci yang panjangnya 168-bit (masing-masing panjangnya 56-bit). Pada 3DES, 3 kunci yang digunakan bisa bersifat saling bebas ($K_1 \neq K_2 \neq K_3$) atau hanya dua buah kunci yang saling bebas dan satu kunci lainnya sama dengan kunci pertama ($K_1 \neq K_2$ dan $K_3 = K_1$). Karena tingkat kerahasiaan algoritma 3DES terletak pada panjangnya kunci yang digunakan, maka penggunaan algoritma 3DES dianggap lebih aman dibandingkan dengan algoritma DES.

Untuk memudahkan penggunaan algoritma 3DES, maka dibuat suatu program algoritma 3DES dengan alat bantu *software* komputer, yaitu Matlab 7.0.4 yang dapat mengenkripsi dan mendekripsi file yang berekstensi .txt.

Kata kunci : 3DES (*Triple Data Encryption Standard*), DES (*Data Encryption Standard*), kriptografi, enkripsi, dekripsi, kunci.

ABSTRACT

Triple Data Encryption Standard (TDES) is one of the symmetrical algorithm of cryptography used to protect data by encoding data. Process in encoding data is encryption and decryption process. 3DES Algorithm is a development algorithm of DES algorithm (Data Encryption Standard). DES different with 3DES because of length keys that used. DES used one key with length 56-bits while 3DES used three keys with length 168-bits (each length 56-bits). Three keys that used in 3DES may independent ($K_1 \neq K_2 \neq K_3$) or two keys independent which one key equal to first key ($K_1 \neq K_2$ dan $K_3 = K_1$). Because of level secret of 3DES algorithm laying in used length keys, the usage of 3DES assumed more peaceful compared to DES algorithm. 3DES algorithm was arranged in Matlab 7.0.4 in order to make easy in encryption and decryption process with file extension .txt.

***Keywords** : 3DES (Triple Data Encryption Standard), DES (Data Encryption Standard), cryptography, encryption, decryption, key.*

1. LATAR BELAKANG MASALAH

Sesuai dengan perkembangan zaman diperlukan suatu cara untuk mengamankan data dan informasi. Salah satu cara untuk mengamankan data adalah dengan cara merubah data tersebut ke dalam bentuk data yang lain yang tidak dapat dimengerti oleh pihak lain, yaitu dengan cara penyandian.

Dalam kriptografi terdapat beberapa algoritma yang dapat menyandikan data. Algoritma yang paling terkenal adalah algoritma DES. DES ditetapkan sebagai standard untuk melindungi data dan informasi. Tetapi, DES dianggap sudah tidak aman lagi, karena dengan perangkat keras khususnya kuncinya dapat ditemukan dalam waktu beberapa hari. Kemudian IBM yang membuat algotima DES mengembangkan DES menjadi 3DES. 3DES juga banyak digunakan dan penggunaannya lebih aman dibandingkan DES. Dalam paper ini dibahas tentang enkripsi dan

dekripsi data dengan algoritma 3DES, dengan lama waktu yang diperlukan dan kecepatannya, serta kekuatan 3DES terhadap serangan *brute force*.

2. PERUMUSAN MASALAH

Berdasarkan uraian latar belakang, masalah yang dapat diidentifikasi penulis adalah:

a. Bagaimana cara mengenkripsi dan mendekripsi suatu data dengan menggunakan algoritma 3DES (*Triple data Encryption Standard*).
b. Berapa besar ukuran file yang dapat dienkripsi dan didekripsi dengan menggunakan algoritma 3DES dalam waktu satu detik.
c. Mengetahui kekuatan algoritma 3DES terhadap serangan *brute force*.

3. TEORI PENDUKUNG

3.1 Operator Logika

Operator biner identik dengan bit pada komputer, yang melibatkan angka 0 dan angka 1. Operator yang digunakan pada algoritma 3DES adalah XOR. Operator XOR digunakan untuk dua inputan. Jika kedua inputan nilainya sama maka nilai outputnya 0, dan jika kedua inputan nilainya berbeda maka nilai outputnya 1.

Tabel 3.1 Operator XOR

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

3.2 Dasar Matematika

3.2.1 Relasi Fungsi

Definisi 3.1 Suatu relasi *f* dari A ke B dikatakan suatu fungsi apabila setiap $x \in A$ dipasangkan atau dipetakan pada tepat satu unsur di B (*Bartle, 1994*).

Definisi 3.2 $f : A \to B$ disebut fungsi *injektif* atau satu-satu apabila

$$x_1 \neq x_2 \Rightarrow f(x_1) \neq f(x_2)$$

atau apabila

$$f(x_1) = f(x_2) \Rightarrow x_1 = x_2$$ (*Bartle, 1994*).

Proses enkripsi dan proses dekripsi dapat dinyatakan dalam notasi matematika sebagai berikut:

$$E_K(P) = C \quad \text{dan} \tag{2.1}$$

$$D_K(C) = P \tag{2.2}$$

dan keseluruhan dapat dinyatakan sebagai:

$$D_K(E_K(P)) = P \tag{2.3}$$

Relasi antara himpunan P (plainteks) dengan himpunan C (cipherteks) harus merupakan fungsi korespondensi satu-satu (*one to one relation*). Maksudnya, dalam proses dekripsi hanya ada satu elemen C yang menyatakan satu elemen P.

3.3 Proses *Padding*

Proses *padding* adalah suatu proses penambahan byte-byte *dummy* pada byte-byte sisa yang masih kosong pada blok plainteks, disimpan pada posisi paling terakhir.

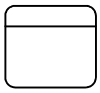
3.4 Alat Perancangan Sistem

Alat perancangan sistem merupakan suatu alat bantu untuk memudahkan penjelasan cara kerja algoritma dan aliran data yang akan disandikan.

3.4.1 Diagram Alur Data (DAD)

Diagram alur data adalah bentuk diagram dari alur data yang diproses. Berikut adalah simbol-simbol dari diagram alur data.

Tabel 3.2 Simbol-simbol Diagram Alur Data (*Kendall and Kendall, 2003*)

Simbol Diagram Alur Data	Fungsi
	Menunjukkan proses.
	Menunjukkan perpindahan data dari satu titik ke titik yang lain.
	Menunjukkan orang, mesin atau perangkat. Dapat berupa sumber data atau tujuan data.
	Menunjukkan data dalam bentuk media penyimpanan fisik.

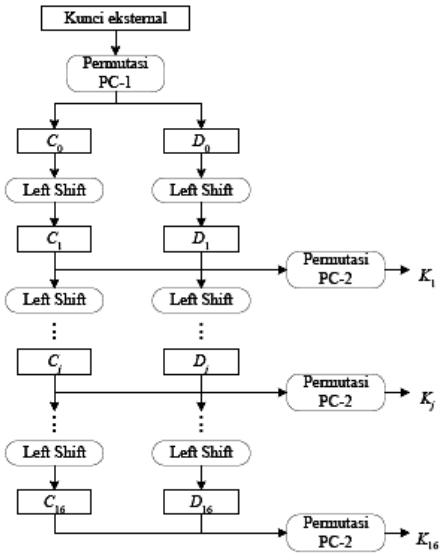
4. PEMBAHASAN

4.1 Data Encryption Standard

DES beroperasi pada ukuran blok 64-bit. DES mengenkripsikan 64-bit plainteks menjadi 64-bit cipherteks dengan menggunakan 56-bit kunci internal yang dibangkitkan dari kunci eksternal yang panjangnya 64-bit.

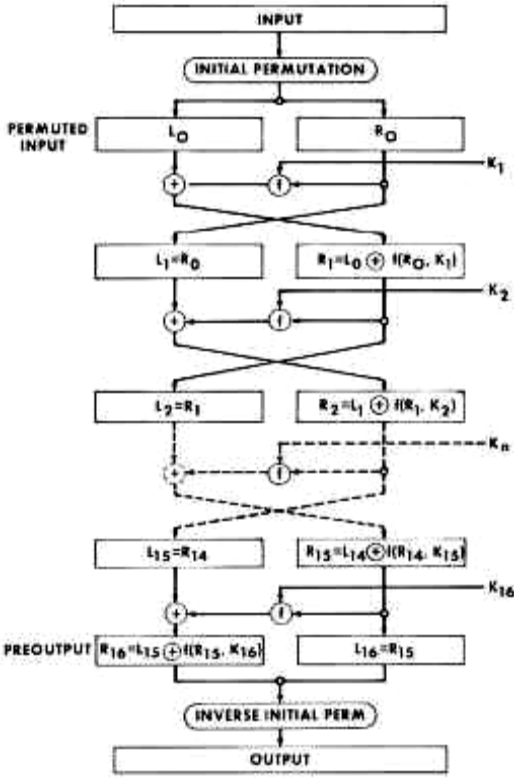
4.1.1 Proses Kunci

Kunci eksternal yang diinputkan akan diproses untuk mendapatkan 16 kunci internal. Pertama, Kunci eksternal yang panjangnya 64-bit disubstitusikan pada matriks permutasi kompresi PC-1. Dalam permutasi ini, setiap bit kedelapan (*parity bit*) dari delapan byte diabaikan. Hasil permutasi panjangnya menjadi 56-bit, yang kemudian dibagi menjadi dua bagian, yaitu kiri (C_0) dan kanan (D_0) masing-masing panjangnya 28-bit. Kemudian, bagian kiri dan kanan melakukan pergeseran bit pada setiap putaran sebanyak satu atau dua bit tergantung pada tiap putaran. Pada proses enkripsi, bit bergeser kesebelah kiri (*left shift*). Sedangkan untuk proses dekripsi, bit bergeser kesebelah kanan (*right shift*). Setelah mengalami pegeseran bit, C_i dan D_i digabungkan dan disubstitusikan pada matriks permutasi kompresi dengan menggunakan matriks PC-2, sehingga panjangnya menjadi 48-bit. Proses tersebut dilakukan sebanyak 16 kali secara berulang-ulang.



Gambar 4.1 Proses Pembangkitan Kunci-kunci Internal DES (*Stinson, 1995*)

4.1.2 Proses Enkripsi



Gambar 4.2 Proses Enkripsi DES (*NIST, 2004*)

Plainteks yang diinputkan pertama akan disubatitusikan pada matriks permutasi awal (*initial permutation*) atau IP panjangnya 64-bit. Kemudian dibagi menjadi dua bagian, yaitu kiri (*L*) dan kanan (*R*) masing-masing panjangnya menjadi 32-bit. Kedua bagian ini masuk ke dalam 16 putaran DES. Satu putaran DES merupakan model jaringan Feistel, secara matematis jaringan Feistel dinyatakan sebagai berikut:

$$L_i = R_{i-1} \quad ; 1 \le i \le 16 \tag{4.1}$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, k_i) \tag{4.2}$$

Bagian *R* disubstituaikan pada fungsi ekspansi panjangnya menjadi 48-bit kemudian di-XOR-kan dengan kunci internal yang sudah diproses sebelumnya pada proses pembangkitan kunci (pada putaran pertama menggunakan kunci internal pertama, dan seterusnya). Hasil XOR kemudian disubstitusikan pada *S-box* yang dikelompokkan menjadi 8 kelompok, masing-masing 6-bit hasilnya menjadi 4-bit. Kelompok 6-bit pertama menggunakan S_1, kelompok 6-bit kedua menggunakan S_2, dan seterusnya. Setelah proses *S-box* tersebut panjangnya menjadi 32-bit. Kemudian disubstitusikan lagi pada matriks permutasi *P-box*, kemudian di-XOR-kan dengan bagian *L*. Hasil dari XOR tersebut disimpan untuk bagian *R* selanjutnya. Sedangkan untuk bagian *L* diperoleh dari bagian *R* yang sebelumnya. Proses tersebut dilakukan 16 kali.

Setelah 16 putaran selesai, bagian *L* dan *R* digabungkan dan disubstitusikan pada matriks permutasi awal balikan (*invers initial permutation*) atau IP^{-1}, hasilnya merupakan cipherteks 64-bit.

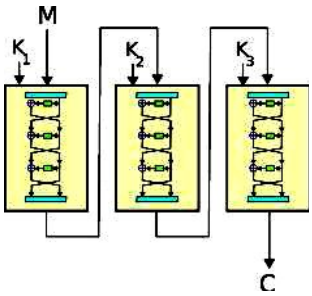
4.1.3 Proses Dekripsi

Proses dekripsi terhadap cipherteks merupakan kebalikan dari proses enkripsi. DES menggunakan algoritma yang sama untuk proses enkripsi dan dekripsi. Jika pada proses enkripsi urutan kunci internal yang digunakan adalah $k_1, k_2, ..., k_{16}$ maka pada proses dekripsi urutan kunci internal yang digunakan adalah $k_{16}, k_{15}, ..., k_1$.

4.2 Triple Data Encryption Standard

3DES (*Triple Data Encryption Standard*) merupakan suatu algoritma pengembangan dari algoritma DES (*Data Encryption Standard*). Pada dasarnya algoritma yang digunakan sama, hanya pada 3DES dikembangkan dengan melakukan enkripsi dengan implementasi algoritma DES sebanyak tiga kali. 3DES memiliki tiga buah kunci yang berukuran 168-bit (tiga kali kunci 56-bit dari DES). Pada algoritma 3DES dibagi menjadi tiga tahap, setiap tahapnya merupakan implementasi dari algoritma DES.

Tahap pertama, plainteks yang diinputkan dioperasikan dengan kunci eksternal pertama (K_1) dan melakukan proses enkripsi dengan menggunakan algoritma DES. Sehingga menghasilkan pra-cipherteks pertama. Tahap kedua, pra-cipherteks pertama yang dihasilkan pada tahap pertama, kemudian dioperasikan dengan kunci eksternal kedua (K_2) dan melakukan proses enkripsi atau proses dekripsi (tergantung cara pengenkripsian yang digunakan) dengan menggunakan algoritma DES. Sehingga menghasilkan prs-cipherteks kedua. Tahap terakhir, pra-cipherteks kedua yang dihasilkan pada tahap kedua, dioperasikan dengan kunci eksternal ketiga (K_3) dan melakukan proses enkripsi dengan menggunakan algoritma DES, sehingga menghasilkan cipherteks (C).



Gambar 4.3 Algoritma 3DES (*NIST, 2004*)

4.2.1 Pemilihan Kunci

Ada dua pilihan untuk pemilihan kunci eksternal algoritma 3DES, yaitu:

a. K_1, K_2, dan K_3 adalah kunci-kunci yang saling bebas

$$\mathbf{K_1 \neq K_2 \neq K_3 \neq K_1}$$

b. K_1 dan K_2 adalah kunci-kunci yang saling bebas, dan K_3 sama dengan K_1

$$\mathbf{K_1 \neq K_2} \text{ dan } \mathbf{K_3 = K_1}$$

(*NIST, 2004*)

4.2.2 Proses Enkripsi dan Dekripsi

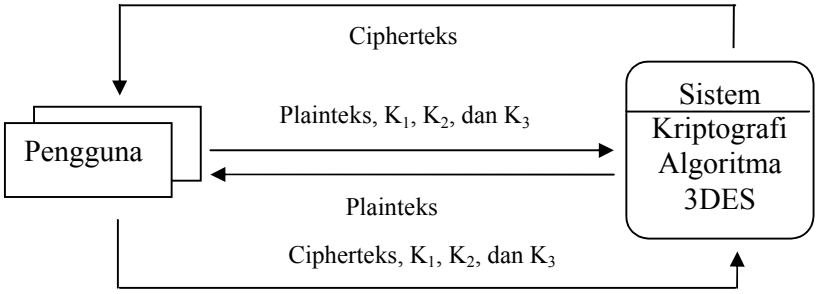
Proses enkripsi dan dekripsi algoritma 3DES dapat dicapai dengan beberapa cara, yaitu:

Tabel 4.4 Cara pengenkripsian dan pendekripsian

Cara	Enkripsi	Dekripsi
1	DES – EDE2 ▪ $K_1 \neq K_2, K_3 = K_1$ ▪ $C = E [D \{E (P, K_1), K_2\}, K_3]$	DES – DED2 ▪ $K_1 \neq K_2, K_3 = K_1$ ▪ $P = D [E \{D (C, K_3), K_2\}, K_1]$
2	DES – EEE2 ▪ $K_1 \neq K_2, K_3 = K_1$ ▪ $C = E [E \{E (P, K_1), K_2\}, K_3]$	DES – DDD2 ▪ $K_1 \neq K_2, K_3 = K_1$ ▪ $P = D [D \{D (C, K_3), K_2\}, K_1]$
3	DES – EDE3 ▪ $K_1 \neq K_2 \neq K_3 \neq K_1$ ▪ $C = E [D \{E (P, K_1), K_2\}, K_3]$	DES – DED3 ▪ $K_1 \neq K_2 \neq K_3 \neq K_1$ ▪ $P = D [E \{D (C, K_3), K_2\}, K_1]$
4	DES – EEE3 ▪ $K_1 \neq K_2 \neq K_3 \neq K_1$ ▪ $C = E [E \{E (P, K_1), K_2\}, K_3]$	DES – DDD3 ▪ $K_1 \neq K_2 \neq K_3 \neq K_1$ ▪ $P = D [D \{D (C, K_3), K_2\}, K_1]$

4.3 Perancangan Sistem

Perancangan dimulai dengan pembuatan diagram konteks, berupa gambaran sistem penerapan algoritma 3DES secara garis besar.



Gambar 4.5 Diagram Konteks

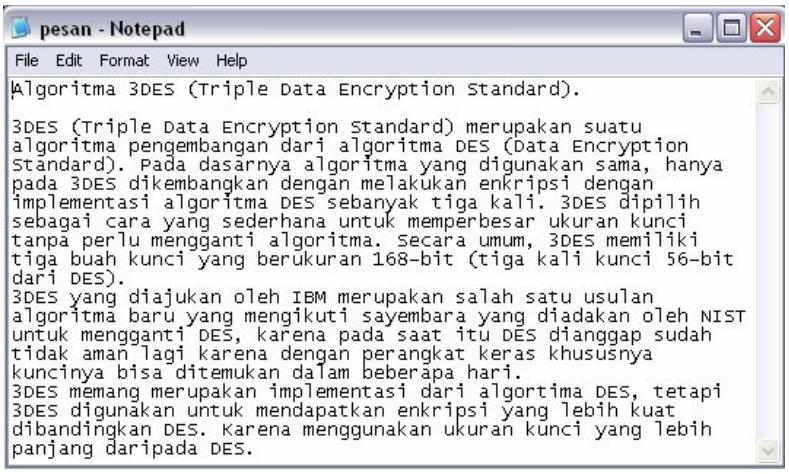
4.4 Hasil Program

Contoh file yang akan dienkripsi dan didekripsi berikut ini diambil dari file yang berekstensi .txt yang berukuran 1 KB (Kilo Byte) dan kunci yang digunakan adalah saling bebas ($K_1 \neq K_2 \neq K_3 \neq K_1$) yaitu:

- Kunci 1 : Enkripsi
- Kunci 2 : Keamanan
- Kunci 3 : Dekripsi

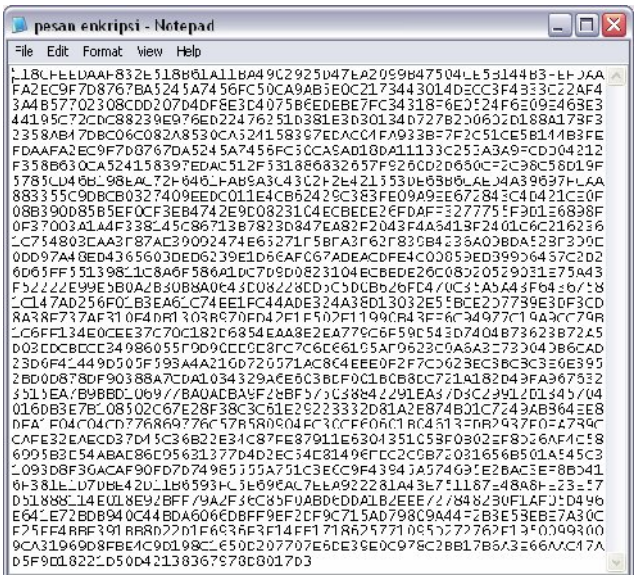
Cara pengenkripsian yang dipilih adalah DES – EDE3 dan cara pendekripsian yang dipilih adalah DES – DED3

Contoh file plainteks:



Aplikasi yang akan ditampilkan adalah sebagai berikut:

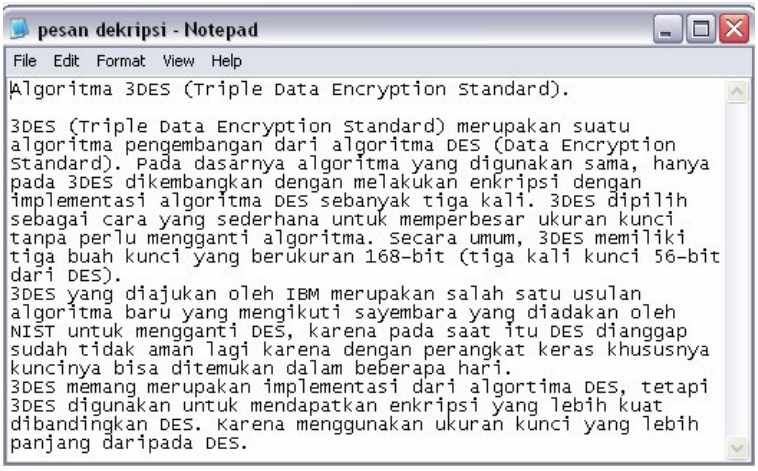
Contoh file cipherteks:



Cipherteks diatas akan didekripsikan kembali dengan menggunakan tiga buah kunci yang sama pada proses enkripsi.

Aplikasi yang akan ditampilkan adalah sebagai berikut:

Maka hasilnya akan sama dengan plainteks semula, yaitu:



Berikut akan ditampilkan proses file untuk algoritma DES dan algoritma 3DES, dengan kunci yang digunakan sebagai berikut:

- Kunci 1 : Software
- Kunci 2 : Komputer
- Kunci 3 : Hardware

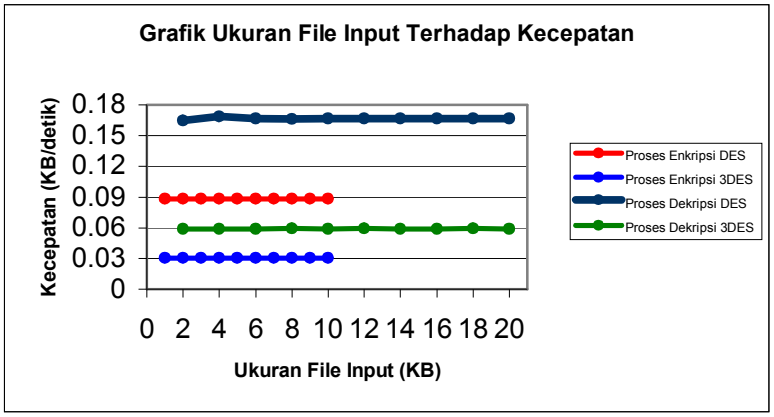
Tabel 4.6 Waktu Proses dan Kecepatannya untuk Proses Enkripsi dengan Algoritma DES dan Algoritma 3DES

No	Nama File			Ukuran File (KB)		Waktu Proses (detik)		Kecepatan (KB/detik)	
	Input	Output							
		DES	3DES	Input	Output	DES	3DES	DES	3DES
1	P1.txt	EP1 DES.txt	EP1 3DES.txt	1	2	11.34	33.093	0.08818	0.03022
2	P2.txt	EP2 DES.txt	EP2 3DES.txt	2	4	22.658	66.197	0.08827	0.03021
3	P3.txt	EP3 DES.txt	EP3 3DES.txt	3	6	33.98	99.302	0.08829	0.03021
4	P4.txt	EP4 DES.txt	EP4 3DES.txt	4	8	45.26	132.324	0.08838	0.03023
5	P5.txt	EP5 DES.txt	EP5 3DES.txt	5	10	56.586	165.29	0.08836	0.03025
6	P6.txt	EP6 DES.txt	EP6 3DES.txt	6	12	67.924	198.463	0.08833	0.03023
7	P7.txt	EP7 DES.txt	EP7 3DES.txt	7	14	79.262	231.15	0.08831	0.03028
8	P8.txt	EP8 DES.txt	EP8 3DES.txt	8	16	90.733	264.882	0.08817	0.03020
9	P9.txt	EP9 DES.txt	EP9 3DES.txt	9	18	101.909	297.451	0.08831	0.03026
10	P10.txt	EP10 DES.txt	EP10 3DES.txt	10	20	113.342	330.389	0.08823	0.03027
Kecepatan Rata-rata								**0.08828**	**0.03024**

Tabel 4.7 Waktu Proses dan Kecepatannya untuk Proses Dekripsi dengan Algoritma DES dan Algoritma 3DES

No	Nama File				Ukuran File (KB)		Waktu Proses (detik)		Kecepatan (KB/detik)	
	Input		Output							
	DES	3DES	DES	3DES	Input	Output	DES	3DES	DES	3DES
1	EP1 DES.txt	EP1 3DES.txt	DP1 DES.txt	DP1 3DES.txt	2	1	12.135	33.887	0.16481	0.05902
2	EP2 DES.txt	EP2 3DES.txt	DP2 DES.txt	DP2 3DES.txt	4	2	23.726	67.731	0.16859	0.05906
3	EP3 DES.txt	EP3 3DES.txt	DP3 DES.txt	DP3 3DES.txt	6	3	36.015	101.707	0.16660	0.05899
4	EP4 DES.txt	EP4 3DES.txt	DP4 DES.txt	DP4 3DES.txt	8	4	48.062	135.303	0.16645	0.05913
5	EP5 DES.txt	EP5 3DES.txt	DP5 DES.txt	DP5 3DES.txt	10	5	59.978	169.244	0.16673	0.05909
6	EP6 DES.txt	EP6 3DES.txt	DP6 DES.txt	DP6 3DES.txt	12	6	72.044	202.868	0.16656	0.05915
7	EP7 DES.txt	EP7 3DES.txt	DP7 DES.txt	DP7 3DES.txt	14	7	84.009	236.904	0.16665	0.05910
8	EP8 DES.txt	EP8 3DES.txt	DP8 DES.txt	DP8 3DES.txt	16	8	95.941	270.864	0.16677	0.05907
9	EP9 DES.txt	EP9 3DES.txt	DP9 DES.txt	DP9 3DES.txt	18	9	107.959	304.506	0.16673	0.05911
10	EP10 DES.txt	EP10 3DES.txt	DP10 DES.txt	DP10 3DES.txt	20	10	119.877	338.645	0.16684	0.05906
Kecepatan Rata-rata									**0.16667**	**0.05908**

Dimana P adalah pesan, EM adalah enkripsi pesan, dan DP adalah dekripsi pesan.



Gambar 4.8 Grafik Ukuran File Input Terhadap Kecepatan

4.5 Tingkat Kerahasiaan Kunci

Semakin panjang kunci yang digunakan, semakin kuat tingkat kerahasiaannya. Algoritma 3DES menggunakan kunci yang panjangnya 168 bit, maka jumlah seluruh kombinasi kemungkinan kunci yang harus dicoba untuk memecahkan cipherteks adalah $2^{168} = 3,741\times10^{50}$ kali. Karena, ada 168 posisi pengisian bit yang masing-masing mempunyai dua nilai kemungkinan, yaitu 0 dan 1.

4.6 Kekuatan Terhadap Serangan *Brute Force*

Brute force adalah teknik mencoba satu persatu kemungkinan kunci untuk memperoleh plainteks. Waktu yang diperlukan untuk mencoba seluruh kemungkinan kunci oleh serangan *brute force* adalah

$$\frac{2^{168}}{3600\times24\times366}=\frac{3.741\times10^{50}}{31.622.400}=1.183\times10^{43}$$ tahun (*Risanto, 2006*).

5. KESIMPULAN

1. Proses enkripsi dan dekripsi suatu data dengan algoritma 3DES dilakukan dengan cara mengimplementasikan algoritma DES sebanyak tiga kali, sesuai dengan pemilihan kuncinya dan urutan proses yang dipilih.
2. Waktu yang diperlukan untuk proses enkripsi dan dekripsi dipengaruhi oleh ukuran file, spesifikasi pada perangkat keras, dan proses lain yang sedang dilakukan oleh perangkat keras.
3. Plainteks yang diproses dengan kunci 1, kunci 2, dan kunci 3 menghasilkan cipherteks dengan jumlah karakter yang lebih besar, karena adanya proses padding dan disimpan dalam bentuk heksadesimal. Jika salah satu kunci atau ketiga kunci dirubah, maka cipherteks juga akan berubah.
4. Kecepatan untuk proses enkripsi dan dekripsi pada setiap pertambahan ukuran file input sebesar 1 KB, kecepatannya adalah sama. Untuk algoritma 3DES, pada proses enkripsi kecepatan rata-ratanya adalah 0.03024 KB/detik dan pada proses dekripsi kecepatan rata-ratanya adalah 0.05908 KB/detik. Sedangkan untuk algoritma DES, pada proses enkripsi kecepatan rata-ratanya adalah 0.08828 KB/detik dan pada proses dekripsi kecepatan rata-ratanya adalah 0.16667 KB/detik.
5. Untuk mendapatkan plainteks tanpa mengetahui kuncinya, jumlah kombinasi kemungkinan kunci yang harus dicoba adalah sebanyak $3,741 \times 10^{50}$ kali.
6. Waktu yang diperlukan untuk mencoba seluruh kemungkinan kunci oleh serangan *brute force* adalah $1,183 \times 10^{43}$ tahun.

6. DAFTAR PUSTAKA

Achmad, Ikhwanudin. 2007. *An Application of Generalized Inverse Matrices on the Hill Cipher*, (online), http://www.ikhwan.web.ugm.ac.id, (diakses 22 Januari 2008).

Away, Gunaidi A. 2006. *The Shortcut of Matlab Programming*. Bandung: Informatika.

Bartle, Robert G.1994. *Introduction to Real Analysis Second Edition.* Singapore: John Wiley.

Felix, Fidens. 2006. *Dasar Kriptografi,* (online), http://www.ilmukomputer.com, (diakses September 2007).

Hasan, Rusydi. 2003. *Mengenal Algoritma DES,* (online), http://www.ilmukomputer.com, (diakses September 2007).

Kendall and Kendall. 2002. *Analisis dan Perancangan Sistem Edisi ke-5 Jilid 2.* Terjemahan oleh Thamir Abdul Hafedh. 2003. Jakarta: Indeks.

Menezes, Alfred J. 1996. *Handbook of Applied Cryptography*. CRC Press.

NIST. 2004. *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*, (online), http://www.csrc.nist.gov, (diakses 22 Januari 2008).

Purcell, Edwin J. 2001. *Kalkulus Edisi ke-7 Jilid 1.* Terjemahan oleh I Nyoman Susila. 2001. Bandung: Interaksara.

Risanto. 2006. *Keamanan Data dengan Kriptografi Kunci Simetris Algoritma DES*. Skripsi tidak diterbitkan. Bandung: Program PascasarjanaUNPAD.

Stinson, Douglas. 1995. *Cryptography: Theory and Practice, (online),* http://www.easywebtech.com, (diakses 22 Januari 2008).