An Implementation of Audio Security Using DES Algorithm

Abdul Wahid[1], Retantyo Wardoyo[2]
[1] Computer Science Study Programme, Gadjah Mada University, Yogyakarta
[2] Computer Science Study Programme, Gadjah Mada University, Yogyakarta
Email: [2] rw@ugm.ac.id

Abstract

Data security is an important problem in computer technology. This paper discusses security system for audio data. This technology is crucial because the multimedia technology has been improved very fast. One of the common audio format forms is wave audio format. The wave format is an uncompressed file format which is for RIFF specification owned by Microsoft. It is used for saving multimedia file. By using DES algorithm, the wave data could be encrypted for hiding information contained in the data. DES algorithm is chosen in this research because DES algorithm is one of the best symmetrical cryptography algorithms and it has been used world wide. This research is expected to give contribution to the audio security concept, especially for audio data security using wave file format.

Keywords : *audio security, DES algorithm, wave format*

1. Pengantar

Pada era globalisasi ini, komputer merupakan suatu perangkat yang sangat dibutuhkan untuk proses pengolahan data, agar data yang diolah tersebut dapat secara efektif dan efisien dalam memberikan informasi yang diperlukan oleh seorang pengguna.

Perkembangan komputer yang saat ini sangat pesat mengakibatkan sistem keamanan (*security*) oleh pemakai (*user*) diperlukan terutama keamanan data. Metode sistem pengamanan data yang lazim digunakan dengan *password* yang terdiri dari beberapa karakter masih sering terjadi kebobolan atau ditembus khususnya pada *password* dengan jumlah karakter yang minim.

Pengamanan cara lain yaitu dengan enkripsi, yaitu dengan menghasilkan data baru tanpa memperhatikan jumlah *byte* yang baru dihasilkan. Enkripsi adalah sebuah proses yang melakukan perubahan sebuah kode dari yang bisa dimengerti menjadi sebuah kode yang tidak bisa dimengerti (tidak terbaca). Proses sebaliknya dikenal dengan nama dekripsi. Kelebihan dari enkripsi ini adalah tidak terjadinya kehilangan *byte*, sehingga bila terjadi proses pembalikan maka akan didapat kembali data sesuai dengan data aslinya. Salah satu algoritma kriptografi yang digunakan dalam penelitian ini adalah algoritma *DES (Data Encryption Standard)*.

Karena fasilitas dan kemudahan yang dimiliki oleh internet maka internet untuk saat ini sudah menjadi barang yang tidak asing lagi. Dengan berkembangnya dunia multimedia, maka kriptografi untuk menyembunyikan pesan yang menggunakan *file-file* multimedia sangat penting. Lalu lintas *file-file* multimedia di internet sudah lumrah dan banyak digunakan oleh para pengguna internet. Salah satu jenis *file* multimedia yang populer adalah *file* dengan format *wav*. Semenjak 6-7 tahun terakhir, *file* audio dengan format ini menjadi yang terpopuler hingga sekarang. Walaupun jenis kompresi memiliki kualitas yang lebih baik, namun sifat kosmopolit dari *wav* belum dapat tersaingi hingga saat ini. Demikian telah ditegaskan oleh Soehono [1].

Alternatif untuk mengirimkan data *audio* ketempat lain melalui komunikasi data adalah dengan menyimpan data *audio* tersebut kedalam file *audio* kemudian mengirimkan *file audio* tersebut ketempat tujuan. *File audio* yang paling banyak digunakan saat ini adalah *file audio* dengan

format *wav*. Hampir semua software media perekaman *audio* akan secara *default* merekam *audio* kedalam *file audio* dengan format *wav*.

Untuk itu dianggap perlu untuk membuat suatu sistem pengamanan data audio (*audio security*) dengan format *wav* agar pada saat pengiriman keamanannya bisa terjamin. Pada sebuah sistem *audio security* yang berfungsi untuk menyembunyikan informasi dalam bentuk *audio file* khususnya yang berformat *wave*, agar dapat menghindari *ilegal user*/akses *user* yang tidak diinginkan maka diperlukan adanya metode khusus untuk menyembunyikan data *audio* tersebut.

Metode *audio security* pada sebuah sistem pengamanan data suara yang akan dirancang yaitu dengan mengambil nilai tertentu dalam *file* yang berformat *wave* kemudian mengakumulasi sebagaimana sehingga menjadi data berupa deretan nilai yang bisa *dienkripsi* dan dengan nilai inilah yang akan menjadi variabel penting dalam metode *enkripsi* audio ini.

Pada sistem yang di rancang, terbentuknya sistem *audio security* yang lebih kuat dan sulit ditembus dalam keamanan data suara dengan menghindari peluang terjadinya kesalahan pada saat penterjemahan kembali dan lebih menekankan pada proses pengamanan data suara. Proses yang dilakukan dengan memanfaatkan nilai *subchunk2* data yang ada pada *file* berformat *wave* serta untuk proses pengamanan datanya maka sistem mengambil, mengkodekan, dan menyimpan nilai–nilai variabel data suara tersebut dengan *algoritma DES*.

Digunakan algoritma *DES* karena algoritma ini merupakan salah satu metode enkripsi dengan algoritma *simetris* yang paling baik dan sangat sulit ditembus oleh para *cryptanalysis* terutama jika pada sistem digunakan metode *DES* bertingkat. Metode ini juga yang paling banyak digunakan didunia untuk algoritma *simetris* sebagai algoritma untuk mengenkripsi berbagai macam data atau informasi. Selain itu setelah mempelajari kelebihan dan kekurangan jenis-jenis algoritma kriptografi yaitu *DES, Triple DES, RC-4, Blowfish,* dan *AES* maka *DES* dianggap algoritma yang paling baik digunakan pada sistem *audio security* karena pertimbangan tingkat keamanan dan kecepatan prosesnya.

Dalam penelitiannya, Ackerman [2], menegaskan bahwa keamanan sebuah data pada abad ini menjadi hal yang sangat penting dan sangat dibutuhkan. Untuk itu Enkripsi data yang kuat dalam sebuah sistem komputer menjadi solusi untuk penyelesaian masalah tersebut. Salah satu cara untuk memperkuat enkripsi adalah dengan enkripsi dalam level bertingkat.

Untuk Pemahaman dasar mengenai Enkripsi data khususnya yang menggunakan Algoritma *DES* maka akan dibahas tentang pengertian dasar enkripsi itu sendiri serta alur dan prinsip kerja dari metode Algoritma *DES* [3]. Algoritma *DES* dirancang untuk menulis dan membaca berita blok data yang terdiri dari 64 bit dibawah control kunci 48 bit. Dalam pembacaan data harus dikerjakan dengan menggunakan kunci yang sama dengan waktu menulis data, dengan penjadwalan alamat kunci bit yang diubah sehingga proses membaca adalah kebalikan dari proses menulis [4].

Kompleksitas algoritma *DES* yang berdasar fungsi *non linear S-box* menjadikan *DES* sulit dipahami. Oleh karena itu dianggap perlu untuk dijelaskan tahap demi tahap enkripsi–enkripsi *DES* beserta aplikasinya. Demikian telah ditegaskan oleh Hendrayani [5].

2. Cara Penelitian

Interface software sistem *Audio security* ini diimplementasikan dengan mengambil nilai data *audio* meliputi *format audio,bit per sample, sample rate,*jumlah *channel* dan *subchunk2 data* dan kemudian melakukan proses enkripsi dekripsi dari nilai yang didapatkan tersebut. Pada implementasinya yang dienkripsi dalam hal ini adalah bagian data dari *file wav* saja karena dianggap akan lebih aman dari serangan. Dengan hanya mengenkripsi datanya saja maka *header* yang formatnya sama untuk semua tipe file *wav* tidak akan menjadi lubang untuk diserang oleh kriptanalisis.

Alasan dipilihnya *Borland Delphi* sebagai *software* untuk membangun sistem ini adalah karena pada software tersebut terdapat komponen-komponen tambahan yang dapat memudahkan dalam membangun sistem ini. Komponen tambahan tersebut yang digunakan pada sistem adalah komponen *Twave* untuk pengelolahan file–file *wav* dan komponen *DCPcrypt* untuk proses kriptografi atau proses enkripsi dekripsi dengan algoritma *DES*. Kedua komponen tersebut merupakan komponen yang *open source* atau dapat diperoleh secara gratis pada situs http://www.cityinthesky.co.uk dan http://ccrma.stanford.edu.

3. Dasar Teori

3.1 File Wave

Wave merupakan format file tidak dimampatkan yang diperuntukkan bagi spesifikasi *RIFF* milik Microsoft yang berguna untuk menyimpan file multimedia. RIIF kependekan dari *Resources Interchange File Format*. RIFF itu sendiri adalah suatu media yang digunakan untuk menyimpan tipe data – tipe data yang berbeda, antara lain: *Audio / visual interleaved data (.avi), waveform data (.wav), bitmapped data (.rdi), midi information (.rmi), color palette (.pal), multimedia movie (.rmn), animated cursor (.ani), a bundle of another riff files (.bnd).*

Menurut Craig [6] sebuah file RIFF dimulai dengan file header diikuti dengan chunk data yang berurutan. Seringkali sebuah file wave hanya terdiri sebuah file RIFF dengan sebuah chunk wave yang terdiri dari 2 sub chunk yaitu sebuah *chunk fmt* dan sebuah *chunk data. Chunk fmt* menjelaskan tentang format data sedangkan *chunk* data berisi sampel data. Pada format *wave* bentuk seperti ini biasa disebut dengan bentuk *Canonical.*

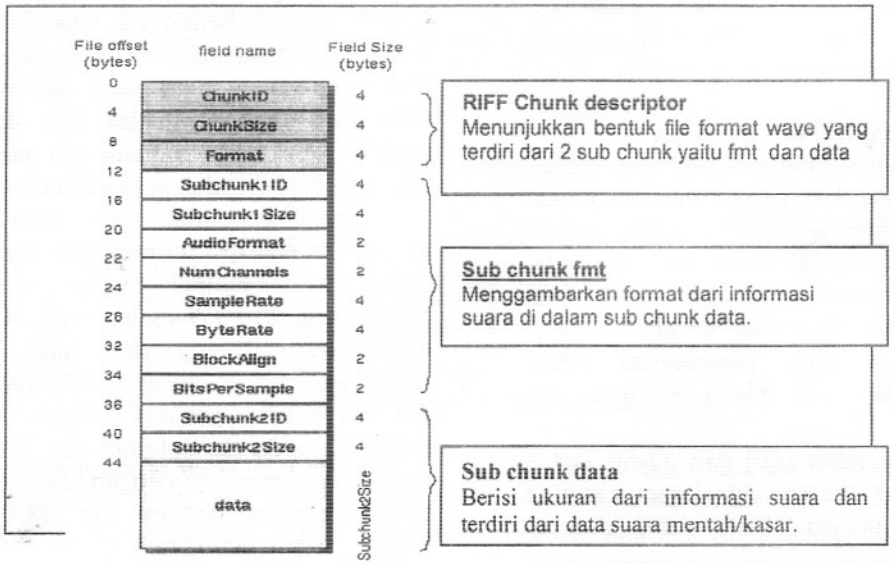
3.2 Struktur File WAV

Struktur file WAV ditunjukkan oleh Gambar 1.

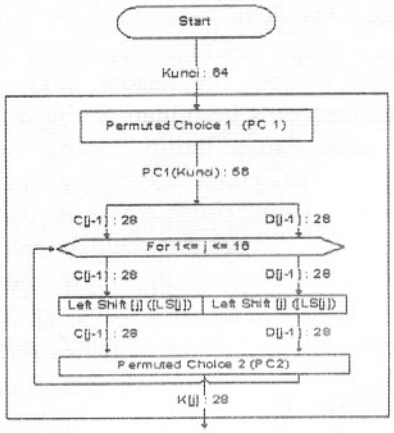
3.3 Kerangka Proses Algoritma DES

3.3. 1 Pemrosesan Kunci

Adapun pemrosesan kunci dalam melakukan penjadwalan pada kunci dalam membuka suatu aplikasi yang ada pada Gambar 2. Diagram Blok Pemrosesan Kunci. Dimana pada proses penjadwalan kunci meminta sebuah kunci 64-bit (8 karakter) dari pengguna. Setiap bit ke 8 digunakan sebagai *bit parity*, dan penjadwalan kunci rahasia (*secret key – scheduling*) dimaksudkan untuk menyusun 16 buah kunci yang akan dimasukkan pada *DES*, baik pada enkripsi maupun deskripsi.



Gambar 1. Struktur File Wav



Gambar 2. Diagram blok pemrosesan kunci

Sehingga proses yang terjadi dalam pemrosesan kunci tersebut, merupakan permutasi yang dilakukan pada kunci 64-bit. Pada tahapan ini, bit-bit *parity* tidak dilibatkan, sehingga bit kunci tereduksi menjadi 56-bit. Bit 1 merupakan kunci 56 merupakan bit 57 kunci dan seterusnya dari *permuted choice 1.* kemudian ouputnya dibagi menjadi dua bagian. 28-bit pertama C[0] dan 28-bit karakter terakhir D[0]. Dari C[0] dan D[0] kemudian dihitung sub-sub kunci untuk setiap iterasi, yang dimulai dengan j=1. Untuk setiap j, rotasi kekiri sekali atau dua kali dijalankan pada C[j-1] dan D[j-1] untuk mendapatkan C[j] dan D[j] dalam setiap iterasi secara terus-menerus hingga ke-16 kunci berhasil tersusun.

3.3.2 Proses Enkripsi dan Deskripsi

Pada proses Enkripsi data yang masuk mengambil blok data 64-bit. Apabila terjadi blok data kurang dari 64-bit, maka penambahan harus dilakukan agar memadai untuk penggunaan. Maka permutasi awal dilakukan pada blok data tersebut dengan mengacu pada blok pemrosesan initial kemudian blok data dibagi menjadi dua bagian.

32-bit pertama L[0] dan 32-bit kedua R[0] kemudian ke-16 sub kunci tersebut dioperasikan dengan blok data, dimulai dengan j=1. R[j-1] dikembangkan menjadi 48-bit menurut pemilihan ekspansi setelah itu E(R[j-1]) di-XOR dengan K[j]. Hasil dari E(R[j-1]) XOR kemudian dipecah menjadi delapan blok 6-bit. Kelompok 1-6 disebut B[1]. Bit 7-12 disebut B[2], dan seterusnya. Jumlah bit dikurangi dengan penukaran nilai-nilai yang ada dalam tabel S untuk setiap B[j]. dimulai dengan j =1, setiap nilai dalam tabel S memiliki 4 bit. Diambil bit ke 1 dan bit ke 6 dari B[j] bersama-sama menjadi nilai 2 bit. Kemudian diambil bit ke 2 dan bit ke 5 dari B[j] sebagai nilai 4 bit dan hasil proses ini adalah S[j][m][n] untuk setiap B[j] sehingga iterasi yang diperlukan sebanyak 8 kali. Pada Gambar 3 terlihat diagram Blok proses Enkripsi.

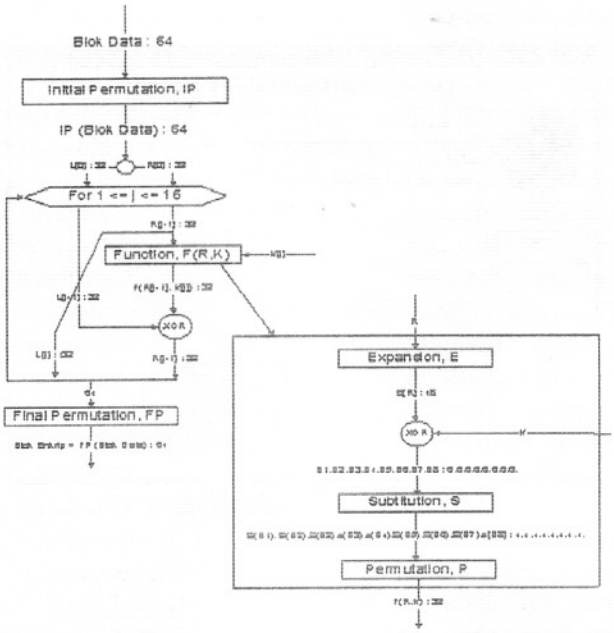
Tetapi pada Proses Deskripsi bekerja di mulai dari blok data Enkripsi yang diolah menjadi data yang sebelumnya di Input, seperti terlihat pada Gambar 4 di atas.

4. Hasil dan Analisa Pengujian Sistem

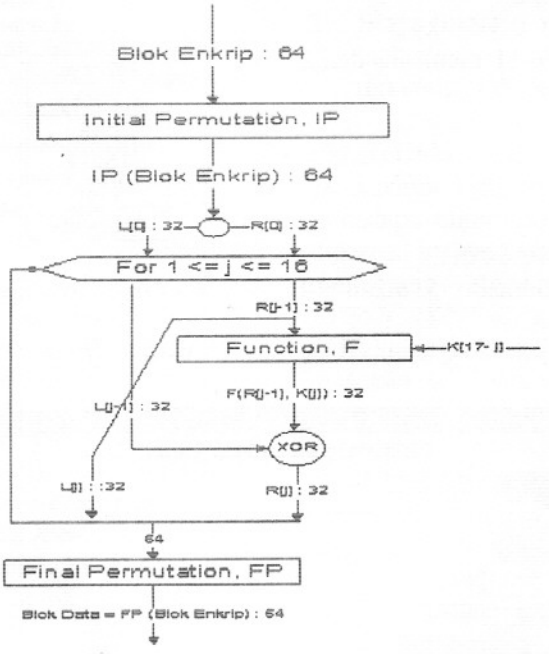
4.1 Proses Enkripsi

Pada gambar 5 memperlihatkan *file wave* yang telah dienkripsi. Pada gambar tersebut bisa kita lihat informasi mengenai file wave yang telah dienkripsi meliputi *audio format, bit per sample, sample rate,* dan jumlah *chanel.* Selain itu bisa dilihat juga *wave* asli dan *wave cipher* dalam bentuk kode *ASCII.* Demikian juga suara dari wave asli dan wave ciphernya bisa kita dengarkan melalui media player yang ada pada panel sehingga bisa dibedakan suara keduanya.

Pada penelitian ini akan dicoba melakukan enkripsi terhadap *file wav* dengan tiga buah sampel yaitu *wahid.wav, aagym.wav,* dan *kemesraan.wav.* Ketiganya mempunyai panjang *file* yang berbeda. *Wahid.wav* dengan *durasi* 3 detik dan ukuran *file* 548 KB, *aagym.wav* dengan durasi 11 menit 33 detik dan ukuran file 15.011 KB serta *kemesraan.wav* dengan durasi 5 menit 39 detik dan ukuran file 58.460 KB.

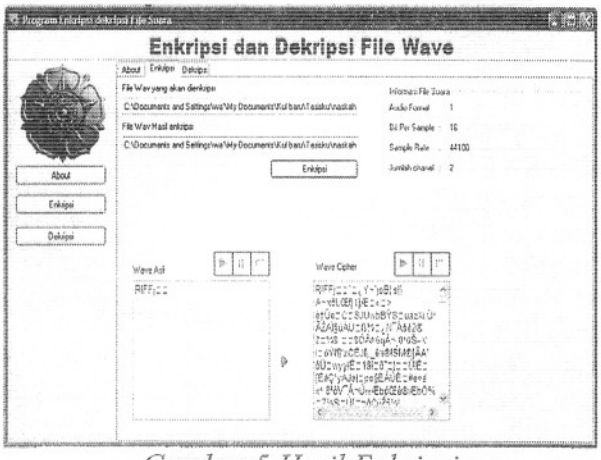


Gambar 3. Diagram blok proses enkripsi



Gambar 4. Diagram blok proses deskripsi

Dari percobaan yang dilakukan dapat diperoleh bahwa ketiga *file wav* tersebut dapat dienkripsi dengan sukses, namun waktu yang diperlukan untuk proses enkripsi berbeda–beda. *File* wahid.wav memiliki waktu proses yang paling kecil. Hanya dengan waktu tidak cukup satu detik proses telah berhasil dienkripsi. Sedangkan waktu proses enkripsi yang paling lama adalah *file kemesraan.wav*. Berikut adalah rincian data hasil enkripsi untuk masing–masing *file* yang yang dienkripsi.



Gambar 5 Hasil Enkripsi

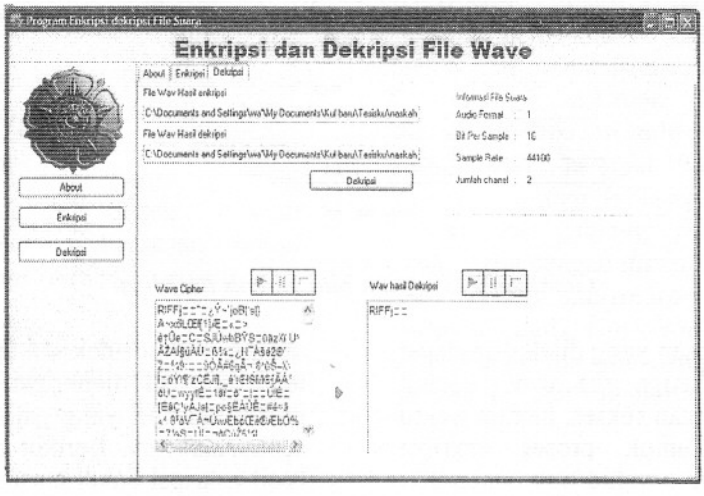
1. Wahid.wav (durasi : 3 detik, size : 548 KB)
   - Audio format : 1
   - Bit per sample : 16
   - Sample rate : 44100
   - Jumlah Channel : 2
   - Waktu Proses : 0.3 detik
2. aagym.wav (durasi : 11 menit 33 detik, size : 15.011 KB)
   - Audio format : 1
   - Bit per sample : 16
   - Sample rate : 11025
   - Jumlah Channel : 1
   - Waktu Proses : 7.64 detik
3. Kemesraan.wav (durasi : 5 menit 39 detik, size : 58.460 KB)
   - Audio format : 1
   - Bit per sample : 16
   - Sample rate : 44100
   - Jumlah Channel : 2
   - Waktu Proses : 34.94 detik

Tabel 1. Hasil pengujian program untuk beberapa ukuran file yang berbeda

Ukuran (Byte)	Waktu (Detik)	Byte/detik
548	0.3	1826.666
15011	7.64	1964.7906
8460	34.94	1673.154
Rata-rata		1724.3247

4.2 Proses Dekripsi

Gambar 6 menunjukkan hasil dekripsi *file wave*.



Gambar 6. Hasil Dekripsi

Semua *file wave* yang telah dienkripsi sebelumnya, dicoba untuk didekripsi sehingga file wave asli dapat ditemukan kembali.Apabila pada proses enkripsi dilakukan dengan berulang misalnya sebanyak tiga kali maka proses dekripsinya juga harus bertingkat sebanyak tiga kali pula untuk mendapatkan *wave* aslinya.Kemudian kunci yang digunakan untuk dekripsi haruslah sama dengan kunci untuk dekripsinya walaupun sebenarnya untuk proses dekripsi kunci akan dibaca dari belakang kedepan sedangkan untuk enkripsi kunci dibaca dari depan kebelakang.
Dari penelitian yang dilakukan diketahui bahwa waktu yang diperlukan untuk proses enkripsi dan dekripsi adalah sama. Hal ini disebabkan karena pada metode *des* langkah dan algoritma yang digunakan untuk enkripsi sama dengan langkah pada proses dekripsi.

4.3 Analisa Kemungkinan Serangan terhadap Sistem

Kompleksitas serangan terhadap suatu sistem dapat diukur dari beberapa hal sebagai berikut :

1. Kompleksitas data (*data complexity*)
   Semakin banyak data yang dibutuhkan untuk melakukan serangan, berarti semakin bagus algoritma kriptografi tersebut. Pada sistem ini untuk memecahkan kunci dengan sistem coba-coba maka jumlah data yang harus dicoba adalah misalkan dimasukkan kunci yang panjangnya 8 karakter, karakter dapat berupa angka (10 buah), huruf (26 huruf besar dan 26 huruf kecil), maka jumlah kunci yang harus dicoba adalah sebanyak 62 X 62 X 62 X 62 X 62 X 62 X 62 X 62 = 62^8 buah. Hal ini merupakan jumlah yang sangat besar untuk dilakukan pengujian.
2. Kompleksitas waktu (*time complexity*)
   Waktu yang dibutuhkan untuk melakukan serangan. Ini disebut juga faktor kerja (*work factor*). Semakin lama waktu yang dibutuhkan untuk untuk melakukan serangan, berarti semakin bagus algoritma kriptografi tersebut. Jika banyak pakar telah mencoba memecahkan algoritma selama 5 tahun setelah dipublikasikan dan tidak seorangpun berhasil, maka mungkin algoritma tersebut tangguh.
3. Kompleksitas ruang memori (*space /storage complexity*)
   Jumlah memori yang dibutuhkan untuk melakukan serangan. Semakin banyak memori yang yang dibutuhkan untuk untuk melakukan serangan, berarti semakin bagus algoritma kriptografi tersebut. Untuk sistem DES, diperlukan memori yang cukup besar untuk melakukan serangan karena algoritma yang digunakannya cukup rumit.

Sebuah algoritma kriptografi dikatakan aman (*computationally secure*) bila ia memenuhi kriteria berikut:

1. Persamaan matematis yang dengan operasi algoritma kriptografi sangat kompleks sehingga algoritma tidak mungkin dipecahkan secara analitik.
2. Biaya untuk memecahkan cipherteks melampaui nilai informasi yang terkandung di dalam cipherteks tersebut.

Teknik kriptanalisis linear diperkenalkan oleh Matsui dan Yamagishi pada tahun 1992 dalam sebuah penyerangan terhadap algoritma *FEAL* dan akhirnya diperbaiki oleh Matsui dan digunakan untuk menyerang algoritma *DES* pada acara *EUROCRYPT* di tahun 1993. Penyerangan terhadap algoritma *DES* tersebut memerlukan masukan 2^{47} pasangan plainteks/cipherteks yang diketahui dan kemudian mengalami perbaikan sehingga masukan yang dibutuhkan hanya 2^{43} pasang.

Kriptanalisis linear mempelajari relasi statistik linear antara bit-bit dari plainteks, cipherteks dan kunci untuk pengenkripsian. Relasi tersebut digunakan untuk memprediksi nilai-nilai bit-bit kunci, dengan diketahui pasangan plainteks/cipherteks yang ada. Namun pada penerapan enkripsi *DES* pada *audio security* ini tidak akan mungkin memprediksi nilai–nilai bit kunci karena satupun pasangan plainteks/cipherteks tidak akan bisa diketahui.

Aplikasi enkripsi dan dekripsi *file wave* yang dihasilkan dan diimplementasikan telah mampu melakukan enkripsi dan dekripsi *file wave* dengan panjang/*durasi* waktu yang tak terbatas tergantung dari kemampuan *hardware*. Jadi dengan sistem ini kita akan bisa mengenkripsi dan mendekripsi *file wave* tanpa batasan panjang dan besar *file*. Sistem ini juga bisa digunakan untuk melakukan enkripsi bertingkat, artinya aplikasi ini bisa digunakan untuk melakukan metode *DES* bertingkat untuk menjamin keamanan dan ketangguhan data.

Sedangkan kelemahan fungsi *interface software* ini adalah bahwa *software* hanya mampu mengenkripsi dan mendekripsi file audio dalam *format wav* saja, sedangkan untuk *format file* yang lain seperti *mp3, mp2, mp1, AAC* dan lain- lain belum bisa dilakukan.

5. Kesimpulan dan Saran

Setelah dilakukan percobaan proses enkripsi dan dekripsi pada sistem *audio security* yang telah dibangun yaitu dengan mengenkripsi tiga buah file *wav* dengan durasi dan ukuran file yang berbeda maka diperoleh kecepatan proses rata-rata adalah 1724,3247 byte/detik .

Sangatlah kecil kemungkinan untuk berhasil membuktikan kerentanan *DES*, jika algoritma *DES* tersebut diterapkan pada proses dekripsi suara karena data suara memiliki data *ASCII* yang acak yang tidak memiliki kata atau kalimat yang mempunyai arti tertentu yang bisa ditebak oleh kriptanalis ataupun diserang dengan menggunakan metode penyerangan yang ada sekarang yaitu metode *Exhaustive attack* atau *brute force attack* dan *Analytical attack*.

Untuk penelitian selanjutnya diharapkan bisa membangun suatu sistem yang mempunyai kemampuan untuk mengenkripsi data audio dengan berbagai macam format *file audio* seperti *mp3, mp2, AAC, wma, wmv, mid, midi* dan lain–lain agar kemampuannya lebih fleksibel. Di samping itu juga diharapkan mampu membangun sistem enkripsi dengan menyisipkan suara lain yang punya arti kedalam sistem enkripsi dengan menggunakan metode *steganografi* agar orang yang mendengarnya tidak mengetahui kalau data suara yang didapatkannya ternyata sebuah data suara yang terenkripsi.

Daftar Pustaka

[1] S. Soehono, *Audio Steganografi Menggunakan Mp3*, Tugas Akhir Kuliah Keamanan Sistem Informasi Teknik Elektro Institut Teknologi Bandung, 2006.

[2] W.M. Ackerman, Encryption: A 21st Century National Security Dilemma: Routledge, part of the Taylor & Francis Group: *International Review of Law, Computers & Technology*, Vol. 12, Number 2, 1998,

[3] D.R. Stinson, , *Cryptography : Theory and Practice*, Computer Science and Engineering Department and Center for Communication and Information Science. University of Nebraska, Lincoln, 1995.

[4] Reaffirmed, *Data Encryption Standard (DES – FIPS PUB 46-3)*, National Institute Standard and Technology, October 1999.

[5] E.N. Hendrayani, *Analisis Kerentanan Enkripsi DES*, Tesis Program Studi Ilmu Komputer Jurusan Ilmu – Ilmu Matematika dan Pengetahuan Alam UGM, 2002.

[6] Craig, *WAVE PCM Soundfile Format*, http://ccrma.stanford.edu, 2003, diakses tanggal 8 Januari 2006.