APLIKASI KRIPTOGRAFI MENGGUNAKAN ALGORITMA RIJNDAEL

Wahyu Frans Muninggar
Teknik Informatika, Institut Teknologi Nasional Malang
wfransm@gmail.com

ABSTRAK

Informasi merupakan kumpulan beberapa data yang telah diolah sedemikian rupa, sehingga bermanfaat bagi orang atau organisasi yang membutuhkannya. Informasi dapat berupa dokumen yang berisi laporan keuntungan, laporan kerugian, biodata perusahaan, dan neraca. Dalam mengirim *file* ke tujuan melalui jaringan komputer sangat beresiko jika tidak dilengkapi dengan sistem keamanan yang baik.

Advance Encryption Standard (AES) adalah suatu metode enkripsi yang mengubah teks asli menjadi teks yang tidak dibaca disebut juga dengan *ciphertext*. Message Digest 5 (MD5) adalah suatu metode kriptografi untuk memabandingkan nilai hash teks asli dengan teks yang telah diterima. Fungsi *hash* mengubah masukkan jumlah variable berapapun menjadi keluaran yang panjangnya tetap.

Penelitian ini bertujuan untuk mengetahui waktu proses yang dibutuhkan AES dalam proses enkripsi dan menciptakan aplikasi yang dapat mengirim *file* melalui jaringan komputer secara rahasia, cepat, dan aman.

Kata kunci : *Informasi, AES, MD5, hash.*

1. PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi telah menjadikan informasi sebagai kebutuhan pokok bagi masyarakat atau organisasi. Informasi merupakan elemen penting bagi sebuah perusahaan, karena informasi dapat membantu suatu perusahaan untuk menjalankan roda perusahaan agar berjalan dengan baik.

Masalah dan ancaman yang terjadi pada proses pengiriman ataupun mendapatkan informasi yang bersifat rahasia, bila informasi tersebut tersebar luas karena adanya penyadapan, pencurian, dan pemalsuan informasi melalui jaringan komputer akan menyebabkan kerugian bagi pemilik informasi. Salah satu cara untuk melindungi data atau informasi dari tindak kejahatan tersebut adalah dengan menggunakan konsep kriptografi.

Organisasi atau perusahaan memerlukan informasi yang dapat dikirim melalui jaringan komputer. Kemudahan tersebut tidak didukung dengan pengamanan yang baik. Sehingga organisasi atau perusahaan membutuhkan sistem keamanan yang baik untuk mengamankan data penting mereka, seperti laporan keuntungan, laporan kerugiaan, neraca, data proyek, desain grafis, dan data internal perusahaan lainnya.

Untuk itu peneliti berupaya mewujudkan implementasi keamanan data dengan menggunakan metode enkripsi Advance Encryption Standard (AES) dan Message Digest (MD5) ke dalam suatu aplikasi yang mudah digunakan, yang berjudul "Aplikasi Kriptografi Menggunakan Algoritma Rijndael".

1.2 Rumusan Masalah

Rumusan masalah membuat aplikasi enkripsi *file* menggunakan metode Advance Encryption Standard (AES) dengan menggunakan bahasa pemrograman vb.net 2010.

1.3 Batasan Masalah

1. Rancangan program aplikasi tersebut menggunakan bahasa pemrograman vb.net 2010.
2. Metode AES digunakan untuk proses enkripsi dan proses dekripsi data.

1.4 Tujuan

Tujuan pembuatan aplikasi adalah membuat aplikasi yang dapat mengirim *file* yang sudah dienkripsi melalui jaringan komputer dan menjaga kerahasiaan dan keaslian data saat proses pengiriman *file* ke penerima.

2. TINJAUAN PUSTAKA

2.1 Kriptografi Pada AES

Pemanfaatan pada metode Advance Encryption Standard (AES) dapat mengatasi sebagian dari kelemahan sistem jaringan komputer, sehingga dapat mengurangi resiko pencurian dan pemalsuan data. Metode Advance Encryption Standard (AES) berbeda dengan Data Encrption Standard (DES) yaitu AES menggunakan transformasi *affine* untuk fungsi S-box (Kromodimoeljo, 2010). Sebenarnya *polynomial field* merupakan cara pandang atau implementasi dari *finite field*, dengan kata lain setiap *finite field* dapat diimplementasi sebagai *polynomial field*. Ada 10, 12, dan 14 putaran (round) dalam AES, jumlah putaran tersebut sesuai dengan ukuran kunci yang digunakan. Setiap putaran terdapat penggantian byte yang sama seperti DES, peralihan, campur jalur, dan penambahan subkunci yaitu XOR bagian kunci dengan keputusan putaran.

Dalam kriptografi terdapat fungsi *hash* yaitu sering disebut juga fungsi satu arah (*one way function*). Message Digest 5 (MD5) merupakan suatu algoritma *hash* yang didesain oleh Profesor Ronald Rivest dari MIT (Rivest, 1994). Proyek MD5CRK didistribusikan dengan tujuan untuk menunjukkan kelemahan dari MD5 dengan

menemukan kerusakan kompresi menggunakan *brute force*.

Hash MD5 lama, sebelum serangan-serangan diungkap, dinilai aman untuk saat ini. Khususnya pada d*igital signature* masih layak digunakan. Pengguna dapat memanfaatkan atau mempercayai MD5 sebagai keamanan pada jaringan komputer.

2.2 Keamanan Informasi Jaringan Komputer

Jaringan komputer diibaratkan sebagai jalur utama dari teknologi informasi yang menyediakan layanan keamanan bagi pengguna, sehingga informasi-informasi penting pada suatu jaringan tidak dapat disalahgunakan atau diakses oleh pengguna lain yang tidak berhak mengaksesnya.

Berdasarkan analisis ditemukan kelemahan-kelemahan keamanan, baik itu pada *tools* pengamanan jaringan yang digunakan pada setiap *client*/*host* maupun pada instalasi *hardware* jaringannya. Dalam membangun suatu jaringan komputer hal yang harus dipahami dan diperhatikan dengan baik adalah mengenai keamanan informasi, salah satu aspek penting dalam keamanan informasi adalah aspek confidentiality, aspek ini menjamin kerahasiaan informasi pengguna jaringan komputer sehingga informasi tersebut tidak dapat diakses oleh pihak-pihak yang tidak memiliki hak untuk mengaksesnya (Syafrizal, 2007). Banyak terjadinya tindakan kejahatan informasi-informasi terhadap pengguna seperti *username* dan *password* dari suatu akun atau data-data penting disebabkan oleh karena tidak adanya sistem keamanan yang mendukung dalam suatu jaringan komputer, pastinya akan berakibat fatal terhadap pengguna jaringan tersebut.

Keamanan informasi terdiri dari perlindungan terhadap aspek-aspek berikut (Syafrizal, 2007):

1. Confidentiality (kerahasian), artinya informasi dari pengguna jaringan komputer terjamin kerahasiaannya sehingga pihak yang tidak memiliki hak untuk mengakses informasi tersebut tidak dapat mengaksesnya.
2. Integrity (keutuhan), artinya informasi yang dikirim akan sampai kepada pihak yang tepat secara utuh tanpa di intercept oleh pihak ketiga dengan maksud untuk melakukan manipulasi terhadap informasi tersebut.
3. Availability (ketersediaan), artinya user dapat mengakses informasi yang dibutuhkan melalui layanan yang tersedia tanpa terjadi gangguan.

2.3 Aplikasi Keamanan Informasi Jaringan Komputer

Aplikasi keamanan yang dapat menyembunyikan atau mengacak data informasi yang sebelum dapat dibaca menjadi tidak dapat dibaca oleh manusia disebut juga dengan enkripsi. Enkripsi merupakan sebuah proses yang melakukan perubahan sebuah kode dari yang bisa dimengerti menjadi sebuah kode yang tidak bisa dimengerti (Wahana Komputer, 2003 : 17).

3. ANALISIS DAN PERANCANGAN

3.1 Analisis

Pada saat pengiriman *file* berupa dokumen penting melalui jaringan komputer terdapat ancaman yang tidak dapat dihindarkan, yaitu : penyadapan, pencurian, dan pemalsuan data informasi oleh orang yang tidak berkepentingan, maka diperlukan aplikasi yang dapat melindungi data-data dengan baik saat proses pengiriman sampai ke penerima yang bersangkutan.

3.1.1 Sistem yang sudah ada

Pengguna menggunakan jaringan LAN untuk menghubungkan komputer A dengan komputer B, kemudian komputer A menjalankan aplikasi Enkripsi dan komputer B menjalankan aplikasi Dekripsi. Pada aplikasi Enkripsi melakukan proses mengubah pesan yang ada di dalam *file* menjadi pesan tidak terbaca (*ciphertext*), kemudian hasil enkripsi disimpan dalam *file* dengan format **.txt* atau **.word* lalu dikirimkan dengan memasukkan IP yang dituju. Kemudian *file* diterima oleh aplikasi Dekripsi untuk melakukan proses dekripsi yaitu mengubah pesan yang tidak dapat dibaca (*ciphertext*) menjadi dapat dibaca (*plaintext*) oleh pengguna.

3.1.2 Kebutuhan Software dan Hardware

Analisa kebutuhan dalam pembuatan aplikasi Enkripsi dan Dekripsi adalah sebagai berikut :

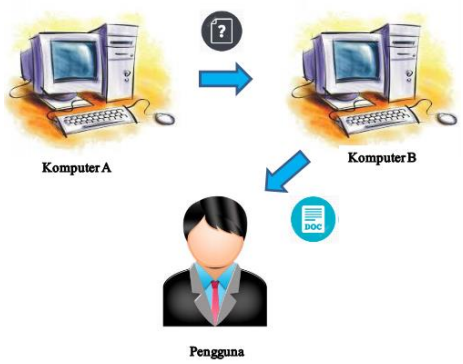
1. Software
   a. Sistem Operasi Windows 8.
   b. Microsoft Visual Studio 2010.
   c. My WIFI Router.
2. Hardware
   a. Processor Intel Core 2 Duo.
   b. Memory RAM 1GB.
   c. Hardisk 700 GB.

3.2 Perancangan

Pada perancangan aplikasi Enkripsi dan Dekripsi memanfaatkan jaringan LAN atau juga bisa menggunakan Wi-Fi dari komputer sebagai perantara komunikasi antara komputer A dan komputer B.

3.2.1 Desain Sistem

Pada desain sistem menjelaskan perancangan aplikasi Enkripsi dan Dekripsi seperti pada gambar dibawah ini :

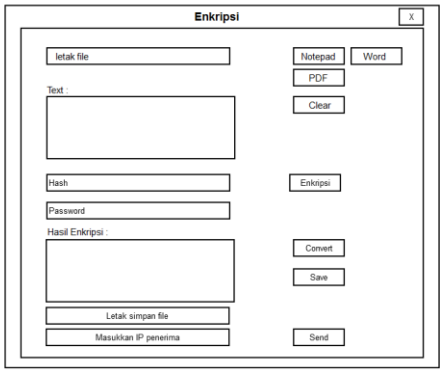


Gambar 1. Blok diagram sistem

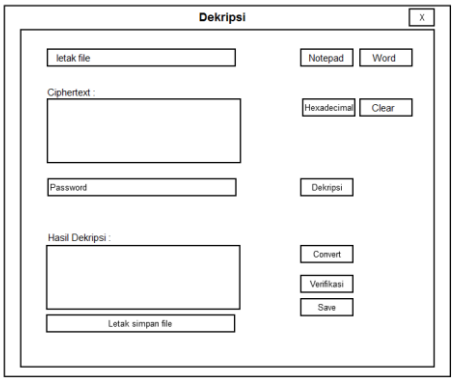
Pada gambar 1 komputer A pengguna membuka aplikasi enkripsi *file*, kemudian memilih *file* yang akan dienkripsi. Setelah proses enkripsi selesai lalu pengguna mengirim *file* tersebut dengan memasukkan IP tujuan yang akan menerima *file* yang sudah dienkripsi. Pada komputer B menerima data dari komputer A kemudian melakukan proses dekripsi agar data informasi dapat dibaca oleh pengguna yang bersangkutan.

3.2.2 Layout

Pada layout menjelaskan tentang perancangan desain yang akan diterapkan pada aplikasi Enkripsi dan aplikasi Dekripsi seperti gambar dibawah ini :



Gambar 2. Layout aplikasi enkripsi



Gambar 3. Layout aplikasi dekripsi

Pada gambar 2 dan gambar 3 terdapat elemen label, *textbox*, dan *button* yang akan diterapkan pada form aplikasi tersebut.

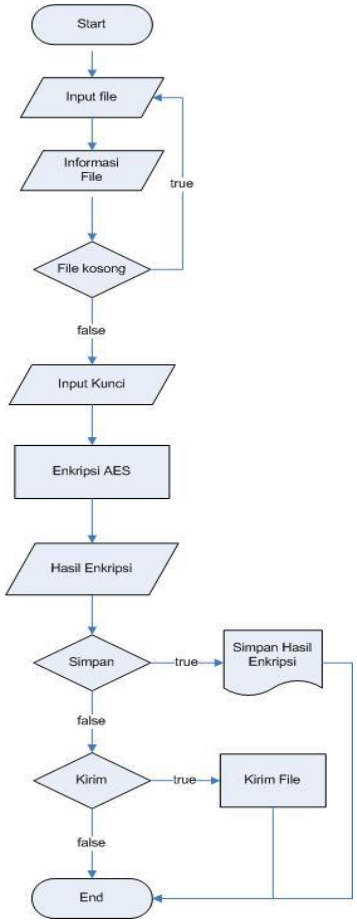
3.2.3 Perhitungan Algoritma Rijndael

Langkah-langkah perhitungan algoritma Rijndael untuk proses enkripsi adalah sebagai berikut:

1. AddRoundKey: melakukan XOR antara state awal (*plaintext*) dengan *cipherkey*. Langkah ini disebut juga Initial Round.
2. Putaran sebanyak Nr – 1 kali. Proses dilakukan pada setiap putaran adalah:
   a. SubBytes: substitusi *byte* dengan menggunakan tabel substitusi (S-box).
   b. ShiftRows: pergeseran baris-baris *array state* secara *wrapping*.
   c. MixColumns: mengacak data pada masing-masing kolom *array state*.
   d. AddRoundKey: melakukan XOR antara *state* sekarang Round Key.
3. Final round: proses untuk putaran terakhir:
   a. SubBytes
   b. ShiftRows
   c. AddRoundKey

3.2.4 Flowchart Aplikasi Enkripsi

Berikut ini adalah flowchart pada bagian aplikasi Enkripsi :

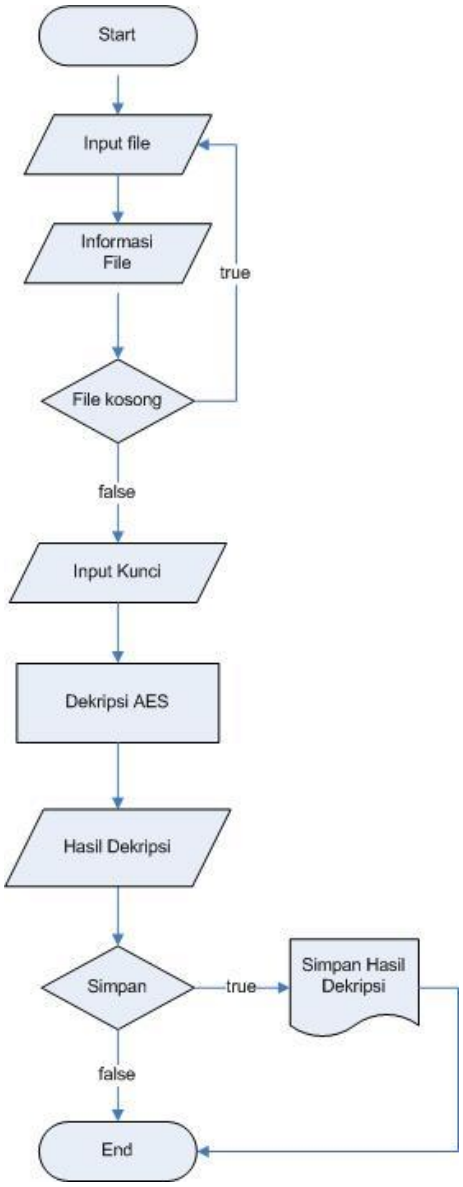


Gambar 4. Flowchart aplikasi enkripsi

Pada gambar 4 mulai dari start aplikasi, membuka *file*, lalu tampil isi teks. Kemudian memasukkan kunci sebanyak 16 digit. Proses enkripsi berjalan untuk mengubah *plaintext* menjadi *cipehertext*. Hasil dari enkripsi dapat disimpan atau tidak

3.2.5 Flowchart Aplikasi Dekripsi

Berikut ini adalah flowchart pada aplikasi Dekripsi :

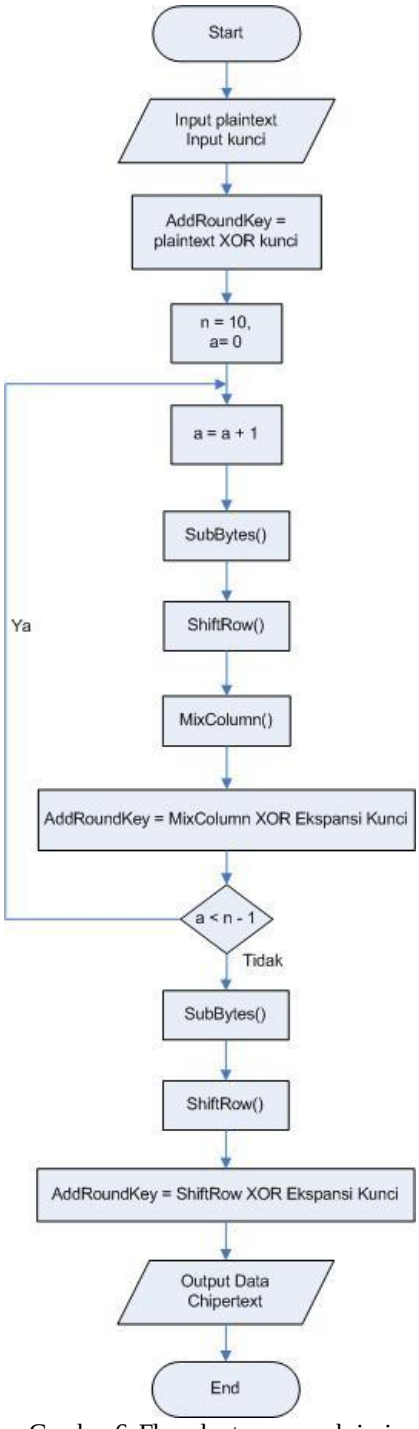


Gambar 5. Flowchart aplikasi dekripsi

Pada gambar 5 dimulai dari start aplikasi, membuka *file*, lalu tampil isi teks. Kemudian memasukkan kunci sebanyak 16 digit. Proses dekripsi berjalan untuk mengubah *cipehertext* menjadi *plaintext*. Hasil dari dekripsi dapat disimpan atau tidak.

3.2.6 Flowchart Proses Enkripsi

Gambar 6 adalah flowchart proses enkripsi AES 128 bit seperti pada gambar dibawah ini :

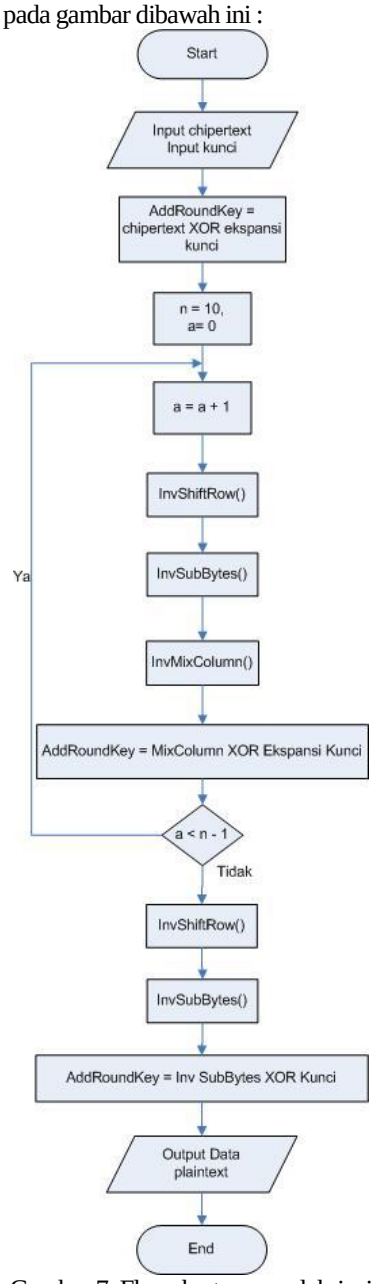


Gambar 6. Flowchart proses enkripsi

Pada gambar 6 terjadi perulangan pada fungsi *SubBytes*, *ShiftRow*, *MixColumn*, dan *AddRoundKey* sebanyak 9 kali. Menghasilkan output berupa *chipertext*.

3.2.7 Flowchart Proses Dekripsi

Gambar 7 adalah flowchart proses dekripsi AES 128 bit seperti pada gambar dibawah ini :



Gambar 7. Flowchart proses dekripsi

Pada gambar 7 terjadi perulangan pada fungsi *InvSubBytes*, *InvShiftRow*, *InvMixColumn*, dan *AddRoundKey* sebanyak 9 kali. Menghasilkan output berupa plaintext.

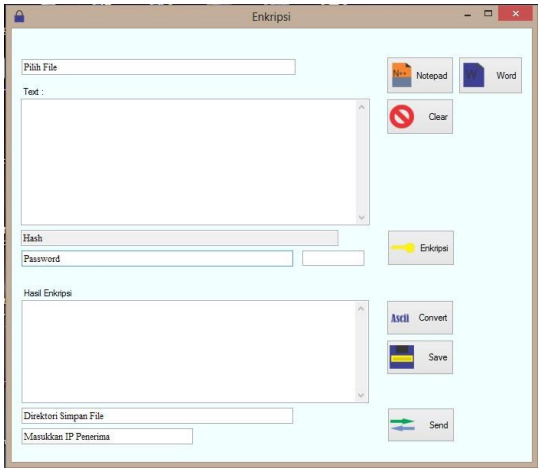
4. IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Hasil rancangan aplikasi enkripsi menggunakan Algoritma Rijndael yang telah didesain sebelumnya pada aplikasi Enkripsi dan aplikasi Dekripsi adalah sebagai berikut :

4.1.1 Tampilan Aplikasi Enkripsi

Berikut adalah tampilan aplikasi Enkripsi seperti gambar dibawah ini.

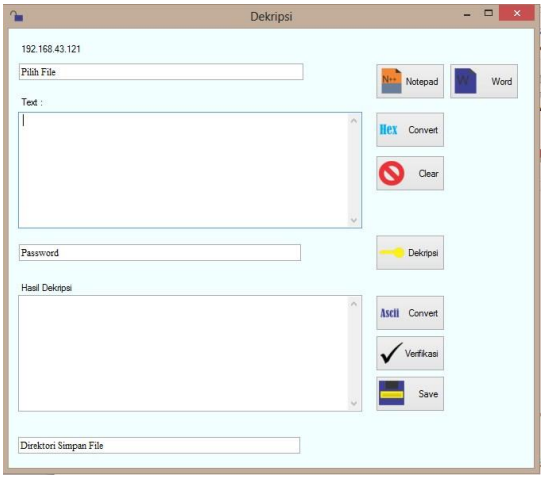


Gambar 6. Tampilan aplikasi enkripsi

Pada gambar 6 tampilan pada form aplikasi Enkripsi terdapat komponen label, *textbox*, tombol *open*, tombol enkripsi, tombol *save*, dan tombol *send*.

4.1.2 Tampilan Aplikasi Dekripsi

Berikut adalah tampilan aplikasi Dekripsi seperti gambar dibawah ini.



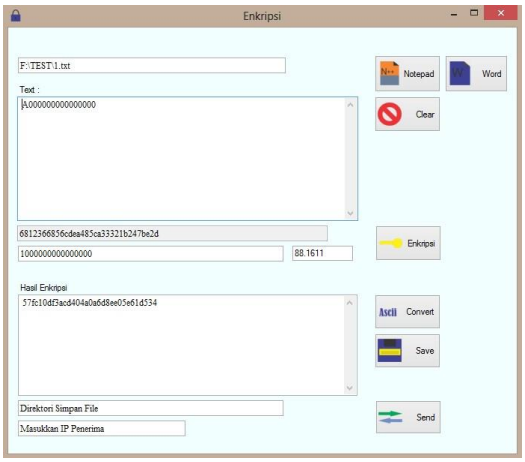
Gambar 7. Tampilan aplikasi dekripsi

Pada gambar 7 tampilan pada *form* aplikasi Enkripsi terdapat komponen label, *textbox*, tombol *open*, tombol dekripsi, dan tombol *save*.

4.2 Pengujian

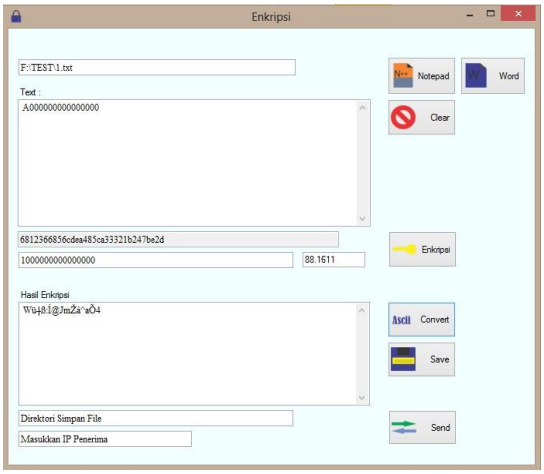
4.2.1 Pengujian Enkripsi dan Pengiriman File

Langkah pertama yang dilakukan adalah membuka *file* yang akan dienkripsi isi teks, misal tekan tombol Notepad untuk membuka *file* yang berformat **.txt*. Kemudian isi *password* sebanyak 16 digit, lalu tekan tombol enkripsi.



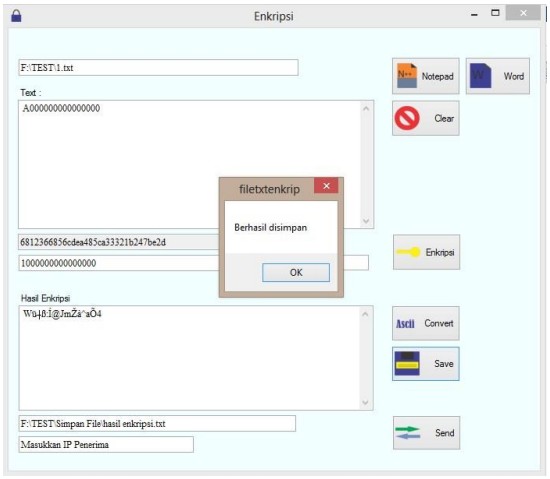
Gambar 8. Tampilan proses enkripsi

Maka hasil enkripsi masih berupa hexadecimal perlu diubah menjadi karakter ASCII dengan cara tekan tombol *convert* maka hasilnya seperti pada gambar 8.



Gambar 9. Tampilan konversi

Karakter telah dikonversi dalam bentuk ASCII seperti pada gambar 9.



Gambar 10. Tampilan simpan

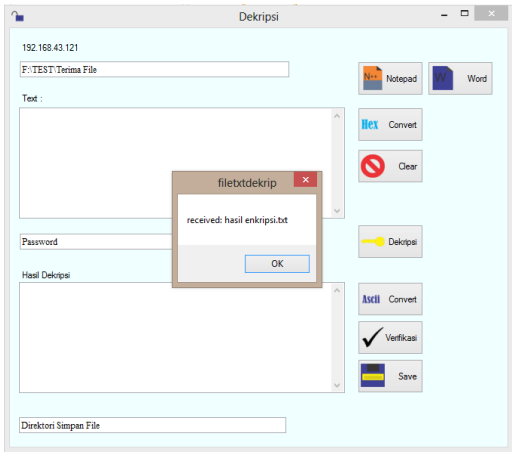
Menyimpan teks hasil enkripsi dengan cara menekan tombol *save*, maka muncul pesan "Berhasil disimpan" seperti pada gambar 10.

Gambar 11. Tampilan kirim

Langkah selanjutnya adalah mengisi IP penerima untuk menerima *file*, lalu tekan tombol *send* seperti pada gambar 11.

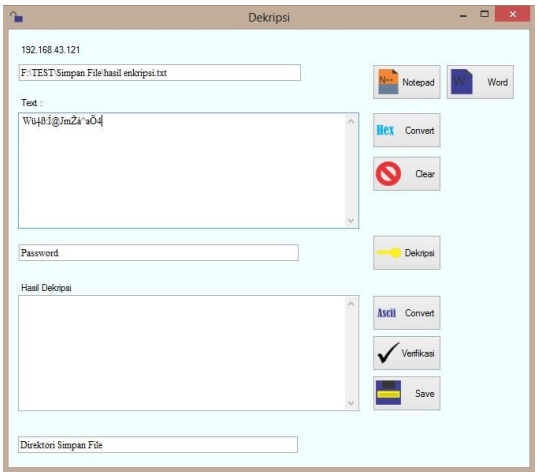
4.2.2 Pengujian Dekripsi dan Simpan File

Pada aplikasi Dekripsi telah menerima *file*, langkah selanjutnya adalah klik OK seperti pada gambar 12.



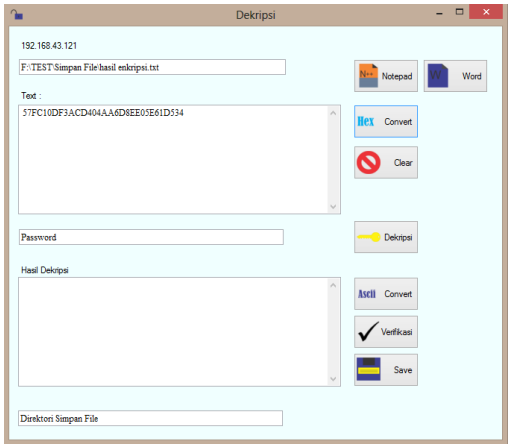
Gambar 12. Tampilan menerima

Menampilkan pesan telah diterima oleh aplikasi Dekripsi seperti pada gambar 12.



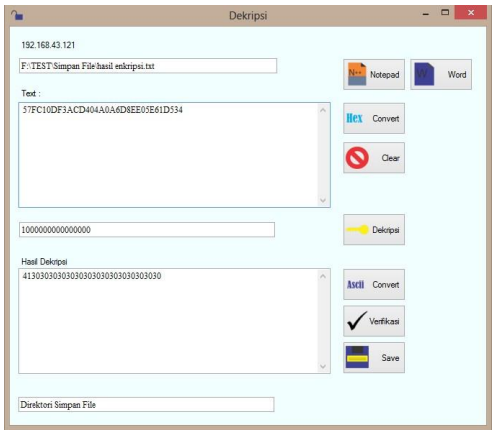
Gambar 13. Tampilan isi *file*

Membuka *file* yang telah diterima dengan cara menekan tombol Notepad jika file berformat **.txt* seperti pada gambar 13.



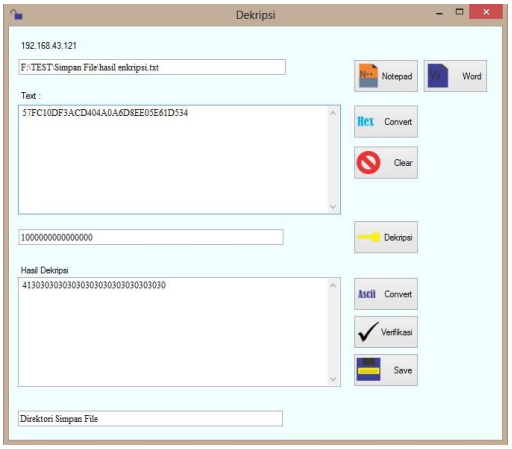
Gambar 14. Tampilan hexadesimal

Karakter yang di buka dari *file* masih dalam bentuk ASCII, perlu diubah ke hexadesimal dengan cara menekan tombol *Hecadecimal* seperti pada gambar 14.



Gambar 15. Tampilan konversi Hexadesimal

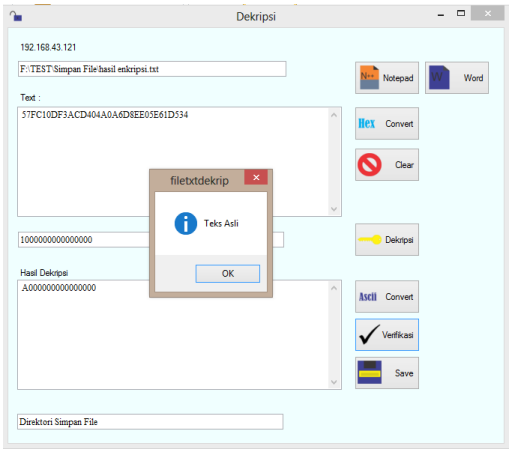
Kemudian memasukkan password sebanyak 16 digit, lalu tekan Dekripsi seperti pada gambar 15.



Gambar 16. Tampilan proses dekripsi

Setelah proses dekripsi selesai, maka hasilnya masih hexadesimal perlu diubah ke ASCII agar muncul teks yang sebenarnya seperti pada gambar 14.

.



Gambar 17. Tampilan pesan asli

Pada pengecekan teks yang dikirm asli atau palsu bisa dengan menekan tombol Verifikasi seperti pada gambar 17.

4.2.3 Pengujian Aplikasi di Sistem Operasi

Pengujian ini bertujuan untuk mengetahuo di sistem operasi yang dapat aplikasi ini berjalan dengan baik, untuk dijadikan rekomendasi. Ditujukan seperti pada Tabel 1.

Tabel 1. Hasil Pengujian Aplikasi di Sistem Operasi

No.	Fungsi	Sistem Operasi			
		B1	B2	B3	B4
1	Form Aplikasi Enkripsi	✓	✓	✓	✓
2	Form Aplikasi Dekripsi	✓	✓	✓	✓

Keterangan :

✓ : Sukses

X : Gagal

B1 : Win 7 32 bit
B2 : Win 7 64 bit
B3 : Win 8 32 bit
B4 : Win 8 64 bit

Dari pengujian pada tabel 1 bahwa aplikasi kriptografi ini dapat berjalan di OS Win 7 dan Win 8.

5. PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian dan pengujian yang telah dilakukan maka diperoleh kesimpulan sebagai berikut :

1. Aplikasi ini menerapkan Advance Encryption Standard (AES) dengan panjang kunci 128 bit.
2. Aplikasi menggunakan jaringan LAN agar dapat mengirim *file* dari aplikasi Enkripsi ke aplikasi Dekripsi.
3. Metode *hash* MD5 digunakan untuk membuktikan keaslian pesan yang diterima dengan pesan asli.

5.2 Saran

Adapun saran yang dapat diberikan untuk penyempurnaan aplikasi ini adalah sebagai berikut :

1. Aplikasi dapat digabungkan dengan metode kriptografi seperti RSA.
2. Aplikasi dapat terhubung dengan koneksi internet agar jangkauan lebih luas.
3. Aplikasi dapat menerapkan metode *hash* seperti SHA1.

DAFTAR PUSTAKA

[1] Ariyus, Dony. 2006. Krip tografi Keamanan Data dan Komunikasi. Yogyakarta : Graha Ilmu.

[2] Ariyus, Dony. 2008. PENGANTAR ILMU KRIPTOGRAFI Teori Analisis dan Implementasi. Yogyakarta : Andi Kromodimoeljo, Sentot. 2010. Teori dan Kriptografi: SPK IT Consulting.

[3] Bahri, Saipul, Diana Diana, and P. S. Dian. "Studi dan Implementasi Pengamanan Basis Data Menggunakan Metode Enkripsi MD5." SKRIPSI MAHASISWA TI S1 (2012).

[4] Kristanto, Andri. 2003. Keamanan Data Pada Jaringan Komputer. Bandung : Gaya Media.

[5] Munir, Rinaldi. 2004. Bahan Kuliah IF5054 Kriptografi. Departemen Teknik Informatika Institut Teknologi Bandung

[6] Munir, Rinaldi. 2006. Kriptografi. Bandung: Penerbit Informatika.

[7] Kromodimoeljo, Sentot. 2009. Teori dan aplikasi kriptografi. SPK IT Consulting.

[8] Sugiarto, I., Santoso, P., & Susanto, A. 2009. Aplikasi Tcp-Ip untuk Mengendalikan Gerak WEBCAM. In Seminar Nasional Aplikasi Teknologi Informasi (SNATI).

[9] Surian, Didi. Algoritma Kriptografi AES Rijndael. Jurnal Teknik Elektro, Vol. 8 N0 2, 97-101.

[10] Syafrizal, Melwin. 2007. ISO 17799: Standar Sistem Manajemen Keamanan Informasi, Seminar Nasional Teknologi 2007 (SNT 2007), ISSN: 1978-9777, Issue.

[11] Wicaksono, Prasetyo Andy. 2006. Studi Pemakaian Algoritma RSA Dalam Proses enkripsi dan aplikasinya. Bandung : Institut Teknologi Bandung.