ANALISIS APLIKASI KRIPTOGRAFI UNTUK SISTEM KEAMANAN PENYIMPANAN DATA ATAU INFORMASI HASIL-HASIL PENELITIAN YANG BERSIFAT RAHASIA

Nova Bintoro Ekaputra

A11.2003.01558

Mahasiswa Jurusan Teknik Informatika Strata 1

Universitas Dian Nuswantoro Semarang (UDINUS)

ABSTRAK

ANALISIS APLIKASI KRIPTOGRAFI UNTUK SISTEM KEAMANAN PENYIMPANAN DATA ATAU INFORMASI HASIL-HASIL PENELITIAN YANG BERSIFAT RAHASIA. Salah satu cara yang digunakan untuk pengamanan data dan atau informasi adalah menggunakan sistem kriptografi. Aplikasi ini, menggunakan algoritma MARS dengan modus ECB (Electronic Code Book). MARS sebagai salah satu kandidat AES(Advanced Encrypted Standard), memiliki kelebihan yaitu mempunyai tingkat keamanan dan proses kecepatan yang tinggi. Hal ini menjadikan algoritma MARS sebagai pilihan terbaik untuk proses enkripsi yang diperlukan oleh dunia informasi menuju abad berikutnya. Algoritma MARS menggunakan kunci 128 bit dan proses enkripsinya terdiri dari 32 ronde. Program ini dirancang dengan menyediakan unit sarana pengiriman file, baik untuk file yang telah dienkripsi maupun jenis file biasa. Hasil pengujian menunjukkan bahwa program ini dapat berjalan sesuai dengan spesifikasi rancangannya.

Kata-kata kunci: Kriptografi, MARS, enkripsi, ECB dan AES

ABSTRACT

CRYPTOGRAPHY APPLICATION FOR SECURITY SYSTEM OF DATA OR INFORMATION OF SECREAT RESEARCH RESULT. The cryptography system is one of several techniques that was used to severe the dataq and or information storage. This application used MARS algorithm in ECB mode. The MARS algorithm is one of a candidate for AES, which have some advantage in high security and process. This advantage of MARS to be the best choice for encrypted process in information world which will be used in the future age. The MARS algorithm used keyword of 128 bit and encrypted process of 32 round.

This program was designed that prepare a file sender facility which is a file already or not encrypted yet. The results of testing showed that the program can be run as their specification design.

Keywords : Cryptography, MARS , encryption, ECB and AES

PENDAHULUAN

Pengetahuan teknologi informasi dan komputerisasi begitu berkembang pesat saat ini. Seiring dengan perkembangan tersebut, maka memungkinkan dari berbagai lapisan masyarakat, termasuk para cracker dan penjahat lainnya dapat akses keberbagai computer yang diinginkannya. Oleh karenanya kebutuhan akan keamanan kerahasiaan data atau informasi menjadi meningkat pula. Ini semua diperlukan untuk menghindari kejahatan terhadap komputer yang timbul sebagai akibat dari kemajuan teknologi tersebut. Terlebih kalau data tersebut terdistribusi dalam suatu jaringan komputer.

Ada berbagai jenis data yang perlu diamankan kerahasiaanya antara lain : data atau dokumen rahasia Negara, data strategis pengembangan usaha perusahaan swasta maupun pemerintah, data atau informasi hasil-hasil penelitian dan lain lain. Rusak atau hilangnya data hasil penelitian yang dideroleh dengan waktu dan biaya yang tinggi tentu saja sangat tidak diinginkan terlebih lebih bila data tersebut merupakan data atau informasi pengembangan usaha atau industri yang strategis.

Untuk menghindari terjadinya kejahatan terhadap komputer, maka diperlukan suatu keamanan yang baik, sehingga data yang terdapat pada komputer menjadi lebih aman. Salah satu cara yang paling baik adalah dengan menggunakan kriptografi. Terdapat berbagai macam algoritma dalam kriptografi, namun tidak semua mampu memberikan jaminan keamanan dan kerahasiaan yang baik terhadap pesan. Salah satu algoritma yang mampu memberikan jaminan keamanan data adalah algoritma MARS, untuk enkripsi data dalam berbagai aplikasi.

Tujuan dari perancangan dan pembuatan program ini adalah untuk mendapatkan suatu program aplikasi kriptografi menggunakan algoritma MARS guna memberikan suatu keamanan data atau informasi yang baik pada sebuah komputer, maupun saat ditransmisikan pada jaringan Rumusan rancangan yang dibuat adalah sebuah program aplikasi komputer yang dapat memberikan keamanan serta kerahasiaan data pada komputer.

Program ini memiliki kendali pengamanan data berupa sistem enkripsi dan dekripsi yang menggunakan symmetric key. Program ini dirancang dengan menggunakan bahasa pemrograman

Visual Basic 6.0. Berdasarkan rumusan rancangan di atas, maka dirancang suatu program yang terdiri dari beberapa unit seperti : unit text editor, unit kriptografi (unit enkripsi dan dekripsi) dan unit pengiriman file dalam jaringan.

Spesifikasi dari perancangan program aplikasi kriptografi dengan menggunakan algoritma MARS adalah :

1. Unit Text Editor

Kemampuan yang dimiliki text editor ini sama seperti Notepad dalam MS Windows.

2. Unit Kriptografi

Unit kriptografi ini terbagi menjadi tiga bagian :

a. unit enkripsi : digunakan untuk mengkode data atau pesan asli (plaintext) yang terdapat dalam text editor menjadi suatu bentuk data atau pesan tertentu (ciphertext). Pengguna dapat langsung mengenkripsi data atau pesan yang dalam bentuk file tersebut dengan cara memilih unit enkripsi file.

b. Unit dekripsi : digunakan untuk mengembalikan ciphertext, yang terdapat dalam text editor, menjadi plaintext. Sama seperti unit enkripsi, unit dekripsi juga menyediakan unit dekripsi file bagi pengguna yang ingin mengenkripsi data yang sudah ada dalam suatu file tertentu.

c. Unit kunci (key) : unit kunci menggunakan kunci simetris (symmetric key), artinya kunci untuk enkripsi harus sama dengan kunci untuk dekripsi.

3. Unit Pengiriman File di dalam Jaringan

Unit pengiriman file digunakan untuk mengirim atau menerima suatu pesan yang telah dienkripsi atau didekripsi, dari satu komputer ke komputer lainnya. Hanya saja pengiriman file ini terbatas dalam ruang lingkup Local Area Network (LAN).

TEORI

Untuk mengenkripsi dan mendekripsi data, kriptografi menggunakan suatu algoritma (cipher) dan kunci (key). Cipher adalah fungsi matematika yang digunakan untuk mengenkripsi dan mendekripsi. Sedangkan kunci merupakan sederetan bit yang diperlukan untuk mengenkripsi dan mendekripsi data. Kriptografi modern dapat memecahkan masalah algoritma tersebut diatas yaitu dengan algoritma kunci. Kunci ini dapat berupa sembarang nilai dari sejumlah angka. Dengan demikian tingkat keamanan dari algoritma yang menggunakan kunci adalah berdasarkan kerahasiaan kuncinya, tidak berdasarkan detail dari algoritma itu sendiri

Element Pembangun Algoritma MARS

Tipe-3 Feistal Network

MARS memiliki panjang blok 128 bit dengan ukuran word 32 bit. Hal ini menunjukkan bahwa setiap blok terdiri dari 4 word. Dalam berbagai struktur network, yang mempunyai kemampuan untuk menangani 4 word dalam satu blok adalah tipe-3 Feistal network.

Operasi yang digunakan algoritma MARS

MARS cipher menggunakan berbagai macam operasi pada 32-bit word yaitu :

1. **Penjumlahan, pengurangan, perkalian dan XOR.**

Ini merupakan operasi yang sangat sederhana, yang digunakan untuk menggabungkan nilai data dan nilai kunci.

2. **Fixed Rotation**

Rotasi berdasarkan nilai tertentu yang sudah ditetapkan. Dalam hal ini nilai rotasi untuk transformasi kunci adalah 13 posisi dengan pergerakan rotasi ke kiri. Untuk r-function adalah 5 dan 13 posisi dengan pergerakan rotasi ke kiri, 24 posisi untuk forward mixing dengan pergerakan rotasi ke kanan dan 24 posisi untuk backward mixing dengan pergerakan rotasi ke kiri.

3. **Data Dependent Rotation**

Rotasi berdasarkan nilai yang ditentukan berdasarkan 5 bit terendah (berkisar antara 0 dan 31) dari word data, misalkan nilai rotasi r = 5 bit terendah dari M maka nilai rotasi r akan sangat tergantung dengan nilai 5 bit terendah dari M.

S-Box

S-box merupakan suatu tabel substitusi yang digunakan pada kebanyakan block cipher lainnya seperti MARS. S-box memiliki ukuran input dan output yang bervariasi, dan dapat disusun secara random atau menurut algoritma tertentu.

MARS menggunakan tabel tunggal yang terdiri dari 512 word yang mengandung 32-bit, yang disebut dengan S-box.

Algoritma Enkripsi dan Dekripsi MARS

Jumlah blok untuk input yang digunakan dalam enkripsi data pada algoritma MARS adalah 128 bit. Sebelum enkripsi blok dimulai, satu blok masukan dibagi menjadi empat word data dimana setiap word data terdiri dari 32-bit data. Untuk selanjutnya keseluruhan operasi internal dilakukan pada 32-bit data atau satu word data. Proses enkripsi dapat dilakukan terhadap semua jenis file dengan mengoperasikannya ke dalam bit-bit biner terlebih dahulu. Proses enkripsi dari algoritma MARS dilakukan dalam 3 tahap yang merupakan bagian struktur cipher dari algoritma MARS yaitu : *forward mixing, cryptographic core* (transformasi kunci S-Box utama), dan *backward mixing*. Proses dekripsi MARS dilakukan kebalikan dari proses enkripsinya.

Struktur Cipher Algoritma MARS

Struktur cipher pada MARS dibagi dalam 3 tahap yakni:

1. **Tahap pertama** adalah *forward mixing*, berfungsi untuk mencegah serangan terhadap chosen plaintext. Terdiri dari penambahan sub kunci pada setiap word data, diikuti dengan delapan iterasi dari S-box. Dalam tahap ini, pertama-tama sebuah sub kunci ditambahkan pada setiap word data dari plaintext, dan kemudian dilakukan delapan iterasi tipe-3 feistel mixing, dikombinasikan dengan operasi mixing tambahan. Dalam setiap iterasi digunakan sebuah word data (disebut source word) untuk memodifikasi tiga word data (disebut target word).

Keempat byte dari source word digunakan sebagai indeks ke dalam 2 S-box, S0 dan S1, yang masingmasing mengandung 256 word 32 bit kemudian nilai S-box entri akan di-XOR-kan, atau ditambahkan pada ketiga word data yang lain. Keempat byte dari source word dinotasikan dengan b0, b1, b2, b3 (dimana b0 adalah byte terendah dan b3 adalahbyte tertinggi). b0,b2 digunakan sebagai indeks untuk S-box S0 dan b1,b3 sebagai indeks untuk S-box S1. S0[b0] di-XOR-kan dengan target word pertama, lalu S1[b1] ditambahkan dengan target word yang sama. S0[b2] ditambahkan dengan target word kedua dan S1[b3] di-XOR-kan dengan target word ketiga. Terakhir source word dirotasikan sebanyak 24 posisi ke kanan. Untuk iterasi berikutnya keempat word data dirotasikan, sehingga target word pertama saat ini menjadi source word berikutnya, target word kedua saat ini menjadi target word pertama berikutnya, target word ketiga saat ini menjadi target word kedua berikutnya dan source word saat ini menjadi target word ketiga berikutnya.

2. **Tahap kedua** adalah "*cryptographic core*". Untuk menjamin bahwa proses enkripsi dan dekripsi mempunyai kekuatan yang sama, delapan iterasi pertama ditunjukkan dalam “forward mode" dan delapan iterasi terakhir ditunjukkan dalam "backward mode”. Cryptographic core pada MARS cipher menggunakan tipe-3 feistal network yang terdiri dari enam belas iterasi. Dalam setiap iterasi digunakan E-function (E adalah notasi dari expansion) yang mengkombinasikan perkalian, data dependent rotation dan S box lookup. Fungsi ini menerima input satu word data dan menghasilkan tiga word data sebagai output dengan notasi L, M dan R. Dalam setiap iterasi digunakan satu word data sebagai input untuk E-function dan ketiga output word data dari E-function ditambahkan atau di-XOR-kan ke ketiga word data yang lama. Kemudian source word dirotasikan sebanyak 13 posisi ke kiri. Ketiga output dari E-function digunakan dengan cara yang berbeda dalam delapan iterasi pertama dibandingkan dengan delapan iterasi terakhir. Dalam delapan iterasi pertama, output pertama dan kedua dari E-function ditambahkan dengan target word pertama dan kedua, output ketiga di-XOR-kan dengan target word ketiga. Dalam delapan iterasi terakhir, output pertama dan kedua dari E-function ditambahkan dengan target word ketiga dan kedua, output ketiga di-XOR-kan dengan target word pertama. E-function menerima input satu word data (32 bit) dan menggunakan dua atau lebih sub kunci untuk menghasilkan tiga word data sebagai output. Dalam fungsi ini digunakan tiga variabel sementara, yang dinotasikan dengan L, M dan R (left, middle, dan right). R berfungsi untuk menampung nilai source word yang dirotasikan

sebanyak 13 posisi ke kiri. M berfungsi untuk menampung nilai source word yang dijumlahkan dengan sub kunci pertama. Sembilan bit terendah dari M digunakan sebagai indeks untuk 512 entry S-box. L berfungsi untuk menampung nilai yang sesuai dengan S box entry. Sub kunci kedua (mengandung integer) akan dikalikan dengan R dan kemudian R dirotasikan sebanyak 5 posisi ke kiri (5 bit tertinggi menjadi 5 bit terendah dari R setelah rotasi). Lalu L di-XOR-kan dengan R dan lima bit terendah dari R digunakan untuk nilai rotasi r dengan nilai antara 0 dan 31, dan M dirotasikan ke kiri sebanyak r posisi. R dirotasikan sebanyak 5 posisi ke kiri dan di-XOR-kan dengan L. Terakhir, lima bit terendah dari R diambil sebagai nilai rotasi r dan L dirotasikan kekiri sebanyak r posisi. Output word pertama dari E-function adalah L, kedua adalah M dan ketiga adalah R.

3. **Tahap terakhir** adalah *backward mixing*, berfungsi untuk melindungi serangan kembali terhadap chosen chipertext. Tahap ini terdiri dari delapan iterasi tipe-3 Feistel mixing (dalam backward mode) dengan berbasis S-box, diikuti dengan pengurangan sub kunci dari word data. Hasil pengurangan inilah yang disebut dengan ciphertext. Tahap ini merupakan invers dari tahap forward mixing, word data diproses dalam urutan yang berbeda dalam backward mode. Seperti halnya pada forward mixing, di setiap iterasi pada backward mixing juga digunakan sebuah source word untuk memodifikasi tiga target word. Keempat byte dari source word dinotasikan dengan b0, bl, b2, b3 sebelumnya (dimana b0 adalah byte terendah dan b3 adalah byte tertinggi). b0, b2 digunakan sebagai index ke dalam S-box S1 dan b1, b3 sebagai index ke dalam S-box S0. S1[b0] di-XOR-kan dengan target word pertama, dan S0[b3] dikurangkan dengan target word kedua. S1 [b2] dikurangkan dengan target word ketiga dan S0[b1] di-XOR-kan dengan target word ketiga juga. Terakhir source word dirotasikan sebanyak 24 posisi ke kiri. Untuk iterasi berikutnya keempat word data dirotasikan sehingga target word pertama saat ini menjadi source word berikutnya, target word kedua saat ini menjadi target word pertama berikutnya, target word ketiga saat ini menjadi target word kedua berikutnya dan source word saat ini menjadi target word ketiga berikutnya.

Notasi yang digunakan dalam cipher :

1. D[ ] adalah sebuah array untuk 4 word 32-bit data. Inisial D berisi *plaintext* dan pada akhir proses enkripsi berisi *ciphertext.*

2. K[ ] adalah array untuk *expanded key*, terdiri dari 40 word 32 bit
3. S[ ] adalah sebuah *S-box*, terdiri dari 512 word 32-bit

Perluasan Kunci

Perluasan kunci berfungsi untuk membangkitkan sub kunci dari kunci yang diberikan oleh pemakai yaitu array *k*[ ] yang terdiri dari n 32-bit word (dimana *n* adalah jumlah words dari 4 sampai 14) ke dalam array K[] sebanyak 40 words.

MARS menerima satu blok *key* awal sebesar 128 bit untuk mendapatkan *key* lain yang digunakan dalam proses enkripsi dan dekripsi. Untuk panjang kunci yang kurang dari 128, maka dilakukan *padding bits* yaitu proses penambahan untuk menambahkan satu dan sisanya ditambahkan dengan nol sampai mencapai panjan kunci yang seharusnya.

Key expansion menyediakan 40 words *key* (K[0],.....,K[39]) yang terdiri dari 32-bit. Empat kunci pertama digunakan untuk proses *forward mixing* dan empat kunci terakhir digunakan pada proses *backwards mixing.*

Komponen Rancangan Program Aplikasi Kriptografi

Perancangan program aplikasi kriptografi ini terdiri dari beberapa modul yang didalamnya memiliki beberapa komponen, diantaranya:

Modul *Splash*

Digunakan sebagai pengenalan awal produk, dan sebagai tanda bahwa *user* sudah memasuki program aplikasi.

Modul *Login*

Digunakan untuk melakukan proses *login*. Dalam modul ini, *user* diharuskan memasukkan *user name* dan *password* yang dimiliki. *User* yang sudah terdaftar pada basis data *user*, dapat memasuki program aplikasi.

Modul *Main*

Digunakan untuk melakukan proses enkripsi dekripsi terhadap pesan yang diinginkan maupun dalam bentuk data dalam file.

Modul *Send*

Digunakan untuk mengirimkan pesan melalui jaringan dengan cara memilih file yang ingin dikirim dan memasukkan alamat IP dari komputer tujuan.

Modul *About*

Digunakan untuk melihat keterangan mengenai program aplikasi ini.

Modul *Help*

Digunakan untuk melihat petunjuk penggunaan program.

ANALISA DAN PEMBAHASAN

Pengujian terhadap program aplikasi kriptografi menggunakan algoritma MARS dengan modus ECB yang bekerja di dalam jaringan ini dilakukan dengan tujuan supaya program ini dapat berfungsi dengan baik dan memastikan apakah hasilnya sesuai dengan spesifikasi dari rancangan.

Pengujian program aplikasi kriptografi ini menggunakan metode pengujian *black box testing*. Metode pengujian ini tidak memperhatikan struktur internal atau sifat dari sebuah program atau modul. *Black box testing* menggunakan strategi dengan melakukan pengujian pemasukan data secara menyeluruh.

Dengan pengujian *black box*, data yang dimasukkan lalu diproses oleh program aplikasi yang dibuat. Pengujian ini dilakukan agar dapat diketahui apakah fungsi dari program aplikasi menghasilkan *output* yang benar dan sesuai dengan spesifikasi rancangan. Jika pada waktu pengujian program, *output* yang dihasilkan tidak sesuai dengan kebutuhan fungsionalnya, berarti masih terdapat kesalahan pada program aplikasi tersebut, dan selanjutnya akan dilakukan perbaikan (*debugging*) untuk memperbaiki kesalahan yang terjadi setelah proses pengujian program. Proses kerja ini juga dapat disebut dengan *Trial and Error.*

Perangkat pengujian program aplikasi kriptografi ini menggunakan dua buah komputer yang berfungsi sebagai pengirim dan penerima. Spesifikasi perangkat keras yang digunakan dalam pengujian adalah sebagai berikut :

1. Komputer dengan *processor* Pentium III 850 Hz
2. *Harddisk* 20 GB dan 20 GB

3. Memori DDR PC 2100 256 MB dan PC 100 512 MB

4. Dua monitor 15”

5. Dua *mouse* Logitech PS/2

6. Dua *keyboard* Logitech PS/2

Spesifikasi perangkat lunak yang digunakan dalam pengujian adalah sebagai berikut :

1. Sistem Operasi Windows XP Professional dan Windows 98 SE.

2. Program aplikasi Microsoft Visual Basic 6.0 Enterprise Edition.

Proses pengujian program aplikasi kriptografi ini dilakukan pada setiap modul untuk meyakinkan apakah program aplikasi yang telah dikembangkan dapat berjalan dengan baik dan sesuai dengan tujuan awal sehingga layak untuk digunakan.

Pengujian program dilakukan dengan membuka *MARS.exe*. Pengujiandilakukan dengan menggunakan metode *black box testing* dimulai dari *form* Splashhingga *form* About.

***Form* Splash**

Form ini berfungsi sebagai awal dari jalannya program aplikasi. *Form* ini hanya tampil sesaat (+ 8 detik) dikarenakan adanya *timer*. *Form* ini dapat berjalan dengan baik.

***Form* Login**

Form ini dibuat untuk memasukkan user name dan password. Hanya user name yang ada pada basis data password saja yang dapat masuk dan kemudian masuk lagi ke *form* Main. Jika *user name* tidak ada pada basis data pemakai atau *password* yang dimasukkan salah, maka akan diberikan pesan kesalahan. *Form* ini dapat berjalan dengan baik.

***Form* Main**

Form ini merupakan *form* utama dari program aplikasi ini. Pada *form* ini terdapat menu *editor*, *toolbar*, serta *status bar*. Pada tampilan *form* ini tampak sebuah *text area*, sebuah *Menubar*, sebuah *Toolbar*, dan sebuah *status bar*. *Menu bar* terdiri atas menu *File, Edit*, dan *Help*. Sedangkan toolbar terdiri atas tombol *New, Open, Save, Print, Send, Cut, Copy, Paste, Encrypt Text, Decrypt Text, Encrypt File, Decrypt File,* dan *Help*. Tampilan pada StatusBar

adalah status program aplikasi, misalnya status “Not Connected” bila komputer tidak sedang terhubung ke komputer lain. Pengujian *form* utama ini sesuai dengan spesifikasi rancangannya dan telah berjalan dengan baik.

Form Open

Pengujian *form* ini dilakukan dengan memilih menu item *Open* pada *Submenu File*. Menu item ini bisa pula dipilih dengan penekanan tombol *Open* pada toolbar. Jika menu item ini dipilih, maka muncul tampilan *form.* Pengujian *form Open* ini sesuai dengan spesifikasi rancangannya dan berjalan dengan baik.

Form Save

Pengujian *form* ini dilakukan dengan pemilihan menu item *Save* pada *Toolbar*. Jika dipilih menu item *Save* dan *File* itu baru akan disimpan, Maka tampilan *form Save*. Pengujian *form Save* ini sesuai dengan spesifikasi rancangannya dan berjalan dengan baik.

Form Print

Pengujian tampilan *form* ini dilakukan dengan pemilihan menu item *Print* pada *Submenu File*, atau dengan penekanan tombol *Print* pada *Toolbar*. Pada *Frame Select Printer* dipilih jenis *Printer* sesuai dengan yang ingin dipakai, misalnya HP Deskjet 690C dan jika tombol *Print* ditekan, maka *Printer* akan mencetak *text* yang ada pada *Text Area*. Pengujian *form Print* ini sesuai dengan spesifikasi rancangannya dan berjalan dengan baik.

Form Encrypt

Pengujian *form* ini dibagi menjadi 2 bagian, yaitu pengujian *form Encrypt Text* dan *form Encrypt File*. Pembagian ini bertujuan untuk membedakan proses enkripsi, seperti *form Encrypt Text* digunakan untuk melakukan proses enkripsi pada *text* yang terdapat pada *text area*, sedangkan *form Encrypt File* khusus digunakan untuk melakukan proses enkripsi langsung dalam bentuk *file*. Pada *Form Encrypt Text*, dilakukan dengan penulisan *text* yang akan dienkripsi pada *text Area* lalu memasukkan kunci enkripsi pada *form encryption key*. Pengujian *form Encrypt* ini sesuai dengan spesifikasi rancangannya dan berjalan dengan baik.

Form Decrypt

Pengujian *form Decrypt* juga dibagi menjadi 2 bagian, yaitu pengujian *form Decrypt Text* dan *form Decrypt File*. Pada *Form Decrypt Text*, dilakukan dengan penulisan *text* yang akan didekripsikan pada *text Area*, dan pada *form decryption key* disi dengan *key* yang dipakai untu melakukan proses dekripsi. Pengujian *form Decrypt* ini sesuai dengan spesifikasi rancangannya dan berjalan dengan baik.

Form Send

Pengujian tampilan *form* ini dilakukan dengan pemilihan menu item *Send* pada Submenu *File*, atau dengan penekanan tombol *Send* pada *Toolbar*. Pengujian *form Send* ini sesuai dengan spesifikasi rancangannya dan berjalan dengan baik.

Pengujian *Form Help*

Pengujian *form* ini dilakukan dengan pemilihan menu item *Contents* pada *Submenu Help*. Pengujian *form Help* ini sesuai dengan spesifikasi rancangannya dan berjalan dengan baik.

Pengujian *Form About*

Pengujian *form* ini dilakukan dengan pemilihan menu item *About* pada *Submenu Help*. Pengujian *form About* ini sesuai dengan spesifikasi rancangannya dan berjalan dengan baik.

Setelah semua form diuji coba dan dijalankan dapat berjalan dengan baik, maka dapat disimpulkan bahwa program aplikasi ini sangat *user friendly* dan terjamin keamanannya karena setiap user yang akan menggunakan program aplikasi ini harus memasukkan *username* dan *password*, dan apabila ada kesalahan pada saat pemasukkan *username* dan *password* yang tidak tepat maka program memberikan kesempatan untuk memperbaiki kembali *username* dan *password* sampai benar tetapi program ini mempunyai kelemahan yaitu tidak mempunyai fungsi authentifikasi. Dalam pengujian program aplikasi ini juga terbukti bahwa proses enkripsi dan dekripsi berlangsung dengan cepat pada panjang kunci 128 bit. Namun untuk panjang kunci lebih dari 128 bit belum dapat diimplementasikan. Pengujian implementasi ini dilakukan pada jaringan *peer to peer*, belum dapat diimplementasikan pada jaringan LAN.

KESIMPULAN

Kesimpulan yang diperoleh dari pembuatan program aplikasi ini adalah sebagai berikut :

Perancangan program aplikasi kriptografi dengan menggunakan algoritma MARS ini dapat memberikan keamanan data atau informasi baik pada saat disimpam pada komputer maupun saat ditransmisikan dalam jaringan komputer.

Data yang diproses dapat dipercaya kerahasiaannya karena telah diacak menggunakan iterasi sebanyak 32 ronde dan masing-masing ronde mengalami fungsi feistel, fungsi E dan S-box dengan panjang kunci 128 bit.

DAFTAR PUSTAKA

1. ANDI, Memahami Model Enkripsi dan Security Data, Yogyakarta: Wahana Komputer, 2003.

2. ANONIMUS, ASCII Table and Extended ASCII Table, www.asciitable.com, 22 Maret 2004.

3. ANONIMUS, *Randomness Testing of the Advanced Encryption Standard Finalist Candidates*, http://www.tropsoft. com/strongenc/mars.html, 14 Oktober 1999. 4. BAKER, RICHARD H., Network Security. Singapura: McGraw-Hill,1995.

5. BURWICK, CAROLYNN et al., MARS : A 128-BitBlockCipher, http://researchweb.watson.ibm.com /security /mars. html, 22 September 1999.

6. KUSUMO, ARIO SURYO. Buku Latihan, Microsoft Visual Basic 6.0. Jakarta : PT Elex Media Komputindo, 2000.

7. PFLEEGER, CHARLES P., Security in Computing. 2nd Edition. Upper Saddle River: Prentice Hall, 1997.

8. STALLING, WILLIAM., Cryptography and Network Security 3rd Edition. Upper Saddle River : Prentice Hall, 2003.

9. WEBER, RON., Information Systems Control and Audit, Upper Saddle River : Prentice Hall, 2000.