PERANCANGAN PERANGKAT LUNAK BANTU UNTUK MEMAHAMI KRIPTOGRAFI METODE CAST-128

Budi Utama[1)], Bayu Sunanda [2)]
Jl. Sei Deli No. 18 Medan, Telp. 061-4567111 Fax. 061-4527548
e-mail: budiutama@yahoo.com[1)]

Abstrak

Teknik pengamanan data yang dikenal sebagai kriptografi saat ini sudah berpengaruh dan meluas penggunaannya. Terdapat banyak sekali algoritma dari kriptografi. Masalah utama yang dihadapi untuk seorang mahasiswa yang mengambil mata kuliah ini adalah sulit untuk memahami cara kerja dari suatu algoritma kriptografi dikarenakan kompleksitas dan melibatkan banyak perhitungan.

CAST merupakan block ciphers yang termasuk dalam keluarga dari DES (Data Encryption System) yang menggunakan substitusi dan permutasi (dikenal sebagai Substitution–Permutation Network atau SPN) dalam perhitungan key dan proses enkripsi dan dekripsi. Algoritma CAST sendiri terdapat dua versi yaitu CAST-128 dan CAST-256 dimana keduanya dibedakan atas panjang kunci yang digunakan. Panjang Key maksimum yang diperbolehkan dalam CAST sebesar 128 bit atau 16 karakter. Selain itu CAST-128 memperbolehkan ukuran kunci bervariasi dari 40 bit sampai 128 bit dengan penambahan 8-bit. Sedangkan panjang plain text yang dapat dienkripsi dan didekripsi adalah sebesar 64 bit (8 karakter) dan mendukung semua jenis plain text.

Tugas akhir ini menjelaskan cara kerja dari CAST-128, desain prosedur enkripsi dari CAST-128, perhitungan dari Key Schedule dengan menggunakan Substitution Boxes (S-Boxes), cara kerja algoritma enkripsi dan dekripsi dalam CAST-128, Hasil implementasi dari algoritma CAST-128 dibuat sebuah program yang sekaligus berfungsi sebagai program pembelajaran untuk memahami algoritma CAST-128 dengan proses pembentukan kunci, enkripsi dan dekripsi algoritma CAST-128.

Kata kunci : Kriptografi, CAST

Abstract

Data security technique known as cryptographic now influential and widespread use. There are a lot of cryptographic algorithms . The main problem encountered for a student taking this course is difficult to understand the workings of a cryptographic algorithm due to the complexity and involves a lot of calculations .

CAST are block ciphers that are included in the family of the DES (Data Encryption System), which uses substitution and permutation (known as – Substitution Permutation Network or SPN) in the calculation of the key and the encryption and decryption process . CAST algorithm itself there are two versions of the CAST - 128 and CAST - 256 both of which are divided into key length used . Key length in CAST maximum allowed is 128 bits or 16 characters . Additionally CAST - 128 allows key sizes vary from 40 bits to 128 bits with the addition of 8 - bit . The length of the plain text to be encrypted and decrypted is equal to 64 bits (8 characters) and supports all kinds of plain text .

This thesis describes the workings of the CAST - 128 , the design procedure CAST - 128 encryption from the calculation of the Key Schedule using Substitution Boxes (S - Boxes) , how the encryption and decryption algorithms in CAST - 128 , CAST implementation results of algorithm - 128 created a program that also functions as a learning program to understand the CAST - 128 algorithm with the process of the formation of the key , encryption and decryption algorithms CAST - 128 .

Keywords*: Quality of Service, Customer Satisfaction*

[1] Pendahuluan

Peneliti mencoba menawarkan suatu langkah untuk pemahaman konsep kerja dari CAST-128 dengan mengembangkan suatu perangkat lunak ajar yang berkenaan dengan metode tersebut. Perangkat lunak ajar tersebut dirancang dengan memanfaatkan fasilitas pemrograman visual sehingga metode CAST-128 dapat dipahami dengan mudah dan menjadi lebih interaktif.

Dengan alasan tersebut maka peneliti tertarik untuk merancang suatu program yang mampu mengamankan *file* atau data dengan mengambil judul "**Perangkat Lunak Bantu Untuk Memahami Kriptografi Metode CAST-128**".

Tujuan dari penelitian ini merancang perangkat lunak untuk pembelajaran dengan metode CAST-128, mencari berbagai keterangan yang berhubungan dengan CAST-128, membantu para pemakai untuk dapat melihat tahapan-tahapan yang dibutuhkan dalam proses kriptografi dengan metode CAST-128.

[2] Tinjauan Pustaka

Komputer merupakan mesin yang memproses fakta atau data menjadi informasi. Komputer di gunakan orang untuk meningkatkan hasil kerja dan memecahkan berbagai masalah. Yang menjadi pemroses data atau pemecah masalah itu adalah perangkat lunak. (Ivan Sudirman, 2003: 1)

Decryption (dekripsi) adalah kebalikan dari enkripsi, yakni transformasi dari data yang telah dienkripsi (*cipher text*) kembali ke bentuk semula (*plain text*). *Encryption* and *decryption* umumnya membutuhkan penggunaan sejumlah informasi yang rahasia, yang sering disebut kunci (*key*). Untuk beberapa mekanisme enkripsi, kunci yang sama digunakan untuk enkripsi dan dekripsi, sementara beberapa mekanisme lain, kunci yang digunakan berbeda pada saat enkripsi dan pada saat dekripsi. (http://en.wikipedia.org/cryptography)

Kriptografi adalah Ilmu atau seni yang berfungsi menyembunyikan informasi menggunakan enkripsi. Kriptologi yang berasal dari bahasa Yunani *kryptos* dan logos yang artinya kata tersembunyi. Sedangkan kriptografer adalah individu atau siapa saja yang mempraktekkan kriptografi. (http://en.wikipedia.org/cryptography)

Kriptanalisis adalah Ilmu yang menganalisa algoritma kriptografi dengan penekanan sungguh-sungguh mengenali kelemahan-kelemahan algoritma kriptografi. Kriptanalis adalah Individu atau siapa-siapa yang menggunakan kriptanalisis untuk mengidentifikasi dan menggunakan kelemahan-kelemahan dalam algoritma kriptografi. (http://en.wikipedia.org/cryptography)

CAST-128 (disebut juga CAST-5) termasuk kelas algoritma yang dikenal dengan *Feistel Ciphers* karena semua operasi hampir mirip dengan *Data Encryption Standard* (DES). Algoritma CAST ditemukan oleh Carlisle Adams dan Stafford Tavares pada tahun 1997 di Kanada. Dan nama CAST merupakan inisial dari nama penemunya yaitu **C**arlisle **A**dams dan **S**tafford **T**avares. CAST dipatenkan oleh *Entrust Technologies* yang berkedudukan di Kanada. CAST-128 mendukung panjang kunci 40, 48, 56, 64, hingga 128 *bit* dengan ukuran blok 64 *bit*. Penggunaan CAST terutama pada PGP v5 yang merupakan *ciphers default* pada PGP (*Pretty Good Privacy*) yang lebih ditujukan sebagai standar keamanan pada *electronic mail* (*e-mail*). Standarisasi dari algoritma CAST-128 dicantumkan pada RFC (*Request For Comment*) yaitu RFC-2144. (Adams, 1995: 133-144)

Pada saat ini telah terdapat algoritma lanjutan dari CAST-128 yaitu CAST-256 (disebut juga dengan CAST6). CAST-256 dirancang oleh Carlisle Adams, Howard Heys, Stafford Tavares, dan Michael Wiener. Arsitektur CAST-256 dibangun berdasarkan CAST-128.

Tujuan perancangan CAST-256 adalah untuk diajukan sebagai salah satu algoritma AES yang diadakan oleh NIST pada tahun 1999. Bersama-sama dengan algoritma Rijndael, Serpent, RC6, dan Twofish, CAST-256 berhasil keluar sebagai lima besar algoritma yang disetujui NIST untuk dijadikan sebagai standar AES. Tetapi pada akhirnya algoritma yang keluar sebagai pemenang adalah Rijndael.

Beberapa kesamaan CAST-256 dengan CAST-128 antara lain pada fungsi perputaran yang digunakan, S-Box, serta tiap putaran menggunakan sepasang *subkey* yaitu *subkey* 5-*bit* Kr sebagai *rotation key* dan 32-*bit* Km sebagai *masking key*.

Panjang kunci CAST-128 adalah sebesar 128 *bit* (16 karakter) selain itu CAST juga memperbolehkan ukuran panjang kunci bervariasi dari 40 *bit* sampai 128 *bit* dengan penambahan 8-*bit* artinya panjang kunci minimum dari CAST adalah 40 *bit* (5 karakter) dan maksimum 128 *bit* (16 karakter). Sama halnya dengan DES sebelum dilakukan proses enkripsi atau dekripsi maka terlebih dahulu dilakukan penjadwalan atau penghitungan kunci. Jumlah kunci yang dihasilkan adalah dari K_1 sampai dengan K_{32}. Dalam *key scheduling*, kunci akan dipecah menjadi 8 *bit* sebanyak 16 buah. Untuk panjang kunci yang kurang dari 128 *bit* maka setelah kunci diubah dalam bentuk biner maka ditambahkan 0 sampai mencapai 128 *bit* pada bagian akhir kunci (bagian *rightmost* atau *Least Significant Bits*). Selanjutnya akan dioperasikan kunci dengan S-Boxes dengan operator "XOR" sampai didapat 32 buah *key*. Dalam *key scheduling* ini hanya digunakan S-Boxes 5 sampai S-Boxes 8. Sedangkan S-Boxes 1 sampai S-Boxes 4 digunakan dalam proses enkripsi dan dekripsi. Secara detail proses *key scheduling* akan dijelaskan pada bagian penguraian algoritma.

[3] Metode Penelitian

Pengumpulan data yang dilakukan oleh peneliti dalam penelitian kepustakaan dimana peneliti mengambil bahan dan sumber-sumber yang berkaitan dengan topik yang dibahas dengan mencari di buku-buku, artikel, materi perkuliahan, dan *website-website* yang ada di *Internet*. Data tersebut diperoleh berasal dari *website* yang membahas algoritma kompresi data khususnya kriptografi dan metode CAST-128. Teknik analisis data berupa analisis contoh perhitungan metode CAST-128 baik untuk proses enkripsi dan proses dekripsi.

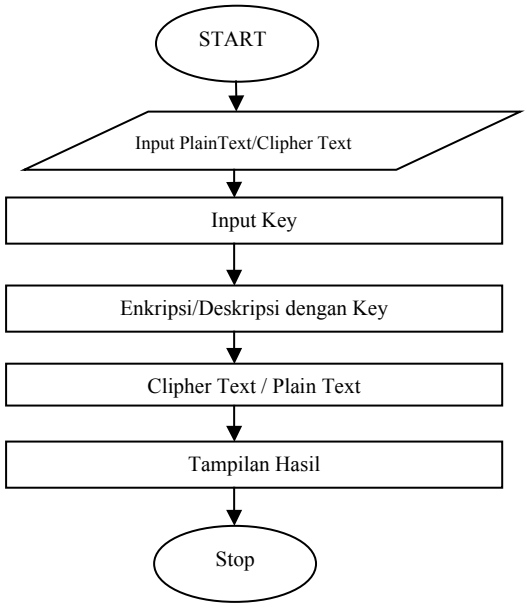
Aplikasi ini dikembangkan menggunakan metodologi *Rapid Application Development* (RAD) dengan tahapan sebagai berikut:

a. Melakukan pengumpulan berbagai data dan informasi yang berkaitan dengan program yang akan dirancang.
b. *Planning* yaitu tahapan merencanakan rancangan aplikasi yang akan dibuat dan mengumpulkan algoritma-algoritma yang digunakan dalam merancang aplikasi.
c. *Prototype* yaitu membuat bentuk *user interface* berdasarkan atas tahapan *planning* di atas.
d. *Analysis* yaitu melakukan analisis terhadap *prototype* yang dirancang jika terdapat kesalahan maka dilakukan koreksi.
e. *Design* yaitu merancang bentuk *prototype* yang telah disempurnakan dengan menggunakan salah satu bahasa pemrograman yang mendukung RAD. Pada tahapan *design* ini jika terdapat ketidaksesuaian maka dapat diulangi langkah pada bagian *prototype* dan *analysis*.
f. *Implementation* yaitu mengimplementasikan *prototype* yang telah dirancang, melakukan *testing* dan perbaikan.

Pada bagian analisis kebutuhan sistem ini dilakukan penguraian atas kebutuhan apa yang diperlukan sehingga program pembelajaran ini dapat dirancang dan tersedia fasilitas apa saja.

Pada tahapan ini dilakukan pengujian terhadap sistem yang dirancang dengan menggunakan berbagai *input string* agar mendapatkan hasil yang diinginkan.

Gambar 3.1 memperlihatkan algoritma perancangan sistem pembelajaran meteode CAST-128.



Gambar 3.1 Algoritma Perancangan Sistem

Algoritma merupakan langkah-langkah maupun urutan bertahap dan spesifik dari suatu masalah. Algoritma ini kemudian diterjemahkan ke dalam program dengan menggunakan bahasa pemrograman tertentu. Algoritma digunakan untuk menganalisa dan menjelaskan urutan dan hubungan antara kegiatan-kegiatan yang akan ditempuh. Selain itu algoritma juga berfungsi untuk menyelesaikan suatu permasalahan sehingga tercapai tujuan yang diinginkan.

3.2.1 Algoritma Menu Utama

Load Skin
Select Case Tombol
    Case Tombol “Pembelajaran”: Show Form Menu Pembelajaran
    Case Tombol “Algoritma CAST-128”: Show Form Algoritma CAST-

128
Case Tombol “Keluar” : Exit Program
End Select

3.2.2 Algoritma Menu Pembelajaran

Load Skin
Select Case Tombol
Case Tombol “Kriptografi”: Show Form Tutorial 1
Case Tombol “Operasi pada CAST-128”: Show Form Tutorial 2
Case Tombol “CAST-128”: Show Form Tutorial 3
Case Tombol “Menu Utama”: Show Menu Utama
End Select

3.2.3 Algoritma Menu CAST-128

Load Skin
Select Case Tombol
Case Tombol “Pembentukan Kunci”: Show Form Generasi Kunci
Case Tombol “Enkripsi”: Show Form Enkripsi
Case Tombol “Dekripsi”: Show Form Dekripsi
Case Tombol “Menu Utama”: Show Menu Utama

End Select

3.2.4 Algoritma *Form Tutorial* 1

Load Skin
Load File Teori.htm ke komponen WebBrowser

3.2.5 Algoritma *Form Tutorial* 2

Load Skin
Load File Matematika.htm ke komponen WebBrowser

3.2.6 Algoritma *Form Tutorial* 3

Load Skin
Load File CAST-128.htm ke komponen WebBrowser

3.2.7 Algoritma *Form* Generasi Kunci

Load Skin
Tampilkan algoritma Generasi Kunci
Select Case Tombol
    Case Tombol "Generasi Kunci": Call Prosedure Generasi Kunci
    Case Tombol "Simpan": Call Prosedure Save
    Case Tombol "Cetak": Call Prosedure Print
    Case Tombol "Tutup": Unload Form
End Select

3.2.8 Algoritma Form Enkripsi

Load Skin
Tampilkan algoritma Enkripsi dan Fungsi F
Select Case Tombol
    Case Tombol "Proses": Call Prosedure Enkripsi
    Case Tombol "Simpan": Call Prosedure Save
    Case Tombol "Cetak": Call Prosedure Print
    Case Tombol "Tutup": Unload Form
End Select

3.2.9 Algoritma Form Dekripsi

Load Skin
Tampilkan algoritma Dekripsi dan Fungsi F
Select Case Tombol
    Case Tombol "Proses": Call Prosedure Dekripsi
    Case Tombol "Simpan": Call Prosedure Save
    Case Tombol "Cetak": Call Prosedure Print
    Case Tombol "Tutup": Unload Form
End Select

[4] Hasil dan Pembahasan

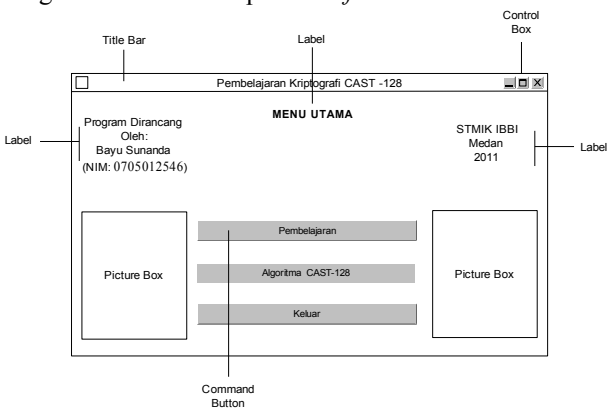
4.1. Perancangan

Secara umum program dapat melakukan operasi melakukan enkripsi pada *string* dan bagian yang lain secara terpisah digunakan untuk menjelaskan teori kriptografi dan cara kerja dari algoritma CAST-128 sehingga secara umum program ini dapat dijadikan sebagai program pembelajaran khusus untuk metode CAST-128.

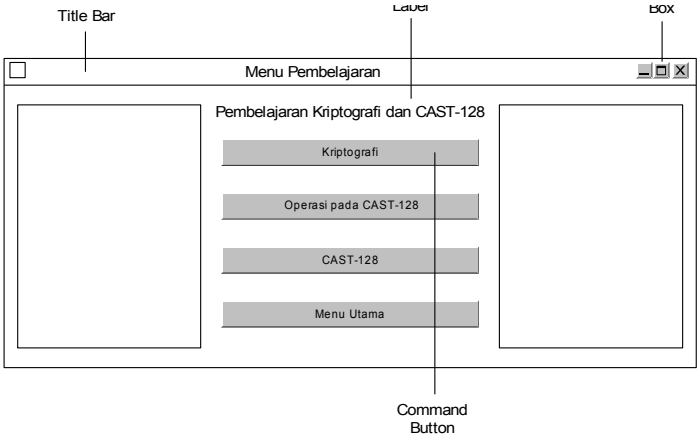
Form yang dirancang mencakup: *form* menu, *form* pembelajaran, *form* algoritma CAST−128, *form* enkripsi, *form* dekripsi, *form* generasi kunci, *form* tutorial 1, dan *form* tutorial 2 serta *form* tutorial 3. Khusus tiga *form* yang disebutkan di bagian akhir digunakan untuk mengajarkan teori kriptografi, operasi yang digunakan pada CAST−128, dan teori algoritma enkripsi dan dekripsi dari CAST-128. Ini mempunyai tujuan agar *user* yang setelah membaca topik yang ada di tiga *form* ini dapat langsung mencoba dengan berbagai nilai kunci, *plain text*, dan *cipher text* pada *form* berikutnya yaitu *form* enkripsi, *form* dekripsi, dan *form* generasi kunci.

Pertama sekali *form* yang dirancang adalah *form* menu. Bentuk rancangan dari *form* ini dapat dilihat pada gambar 4.1. *Form* ini disebut juga *form* utama karena dari *form* ini semua *form* yang lain dapat diakses. Setiap tampilan ke *form* yang lain apabila *user* menutup *form* maka tampilan akan dikembalikan ke *form* ini.

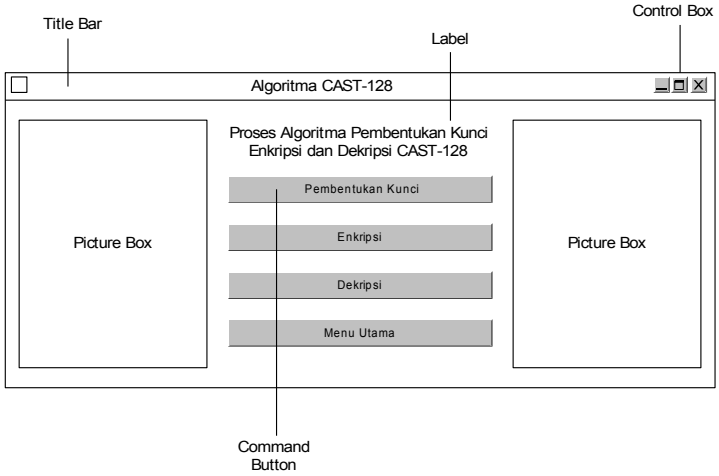
Secara umum *form* ini dibentuk dengan menggunakan komponen grafis seperti *label*, *command button*, dan *picture box*. Bagian yang dibuat dengan menggunakan *label* adalah teks yang mempunyai *caption* seperti "MENU UTAMA", "STMIK IBBI Medan 2011", dan "Program Dirancang oleh: Bayu Sunanda (NIM: 0705012546)". Selain itu pada *layout* terlihat dua buah kotak merupakan objek *picture box* yang akan ditempatkan gambar untuk memperindah *form*.



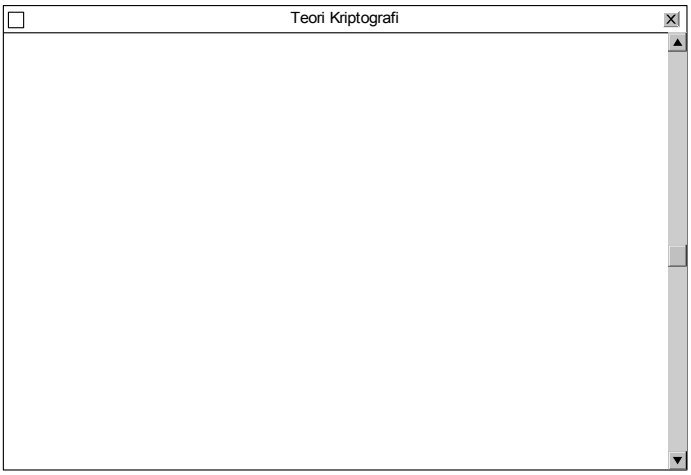
Gambar 4.1 Rancangan *Form* Menu Utama



Gambar 4.2 Rancangan *Form* Menu Pembelajaran



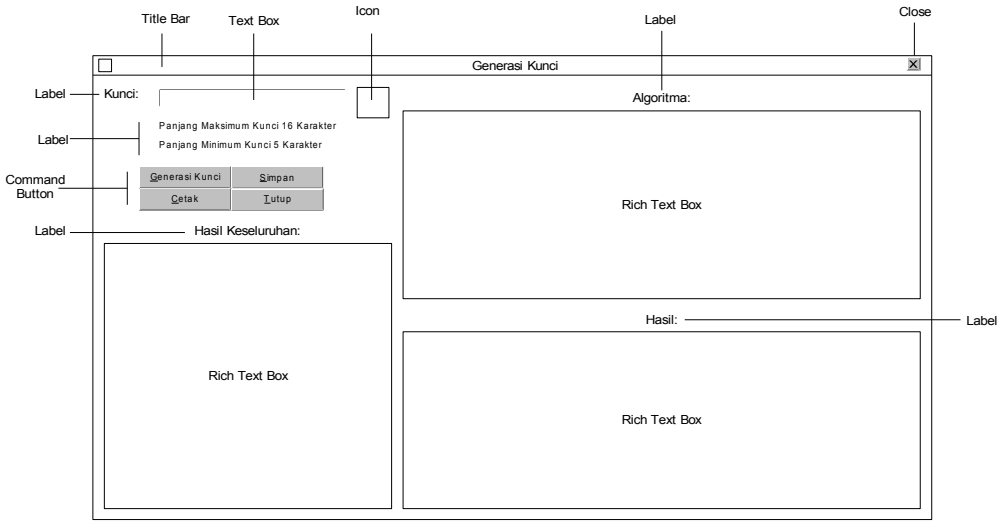
Gambar 4.3 Rancangan *Form* Menu Algoritma CAST-128



Gambar 4.4 Rancangan *Form Tutorial* 1

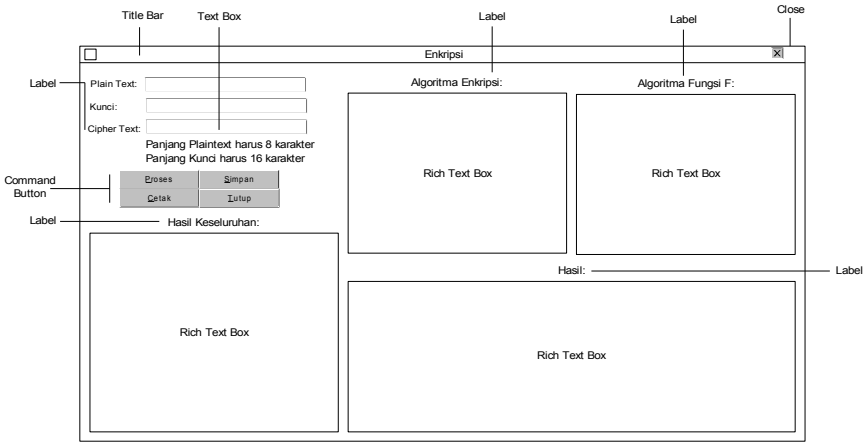
Form ini (Gambar 4.7) terdiri atas komponen *label*, *command button*, dan *text box*. *Label* pada *form* ini seperti biasa berfungsi sebagai teks keterangan. Hanya terdapat dua *label* dengan *caption* "Kunci:" dan "Algoritma". Satu buah *rich text box* sebagai tempat untuk menginput panjang kunci berupa *string*. Dua buah *rich text box* di sisi kanan masing-masing berfungsi untuk meletakkan teks algoritma dan untuk menampung proses perhitungan generasi kunci CAST-128. Sedangkan *rich text box* pada sisi kiri bawah berisi perhitungan akhir.

Tombol "Generasi Kunci" digunakan untuk melakukan proses generasi kunci. Tombol "Simpan" digunakan untuk menyimpan hasil perhitungan dalam bentuk *.TXT. Tombol "Cetak" digunakan untuk mencetak hasil perhitungan ke *printer*. Terakhir tombol "Tutup" digunakan untuk menutup *form* ini dan kembali ke menu.



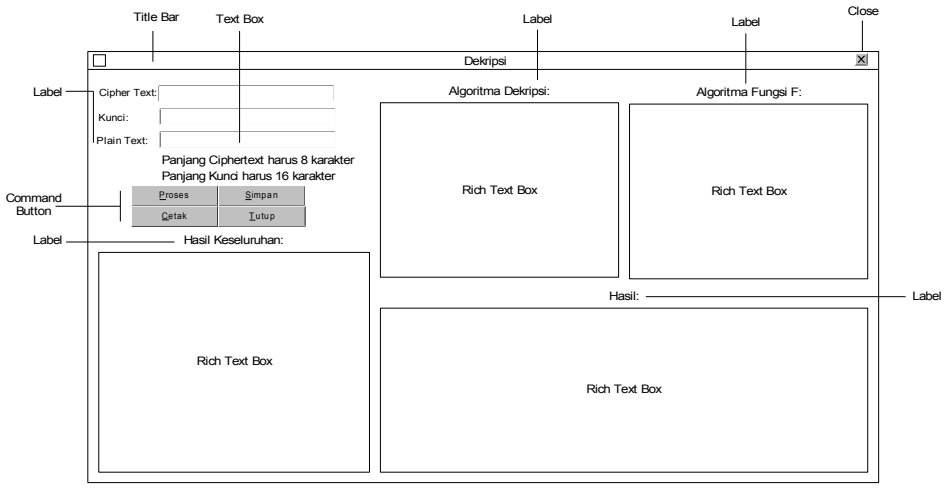
Gambar 4.7 Rancangan *Form* Generasi Kunci

form enkripsi (Gambar 4.8) yang berfungsi untuk menjelaskan proses enkripsi CAST-128. Seperti biasa *form* ini terdiri atas empat jenis komponen yaitu *label*, *command button*, *text box*, dan *rich text box*. Bagian *label* masing-masing dengan *caption* "*Plain Text*:", "*Cipher Text*:", "Kunci", "Hasil", "Algoritma Enkripsi:", "Algoritma Fungsi F", dan "Hasil Keseluruhan". Sedangkan tiga buah *text box* masing-masing berfungsi sebagai input *plain text*, *cipher text*, dan *kunci*. Empat buah *rich text box* digunakan untuk meletakkan algoritma enkripsi CAST-128 dan algoritma fungsi F, serta menampung proses perhitungan enkripsi CAST-128. Tombol "Proses" untuk melakukan proses enkripsi. Tombol "Simpan" digunakan untuk menyimpan hasil perhitungan dalam bentuk *.TXT. Tombol "Cetak" digunakan untuk mencetak hasil perhitungan ke *printer*. Terakhir tombol "Tutup" digunakan untuk menutup *form* ini dan kembali ke menu.



Gambar 4.8 Rancangan *Form* Enkripsi

form dekripsi (Gambar 4.9) yang berfungsi untuk menjelaskan proses dekripsi CAST-128. Seperti halnya dengan *form* enkripsi, *form* ini terdiri atas empat jenis komponen yaitu *label*, *command button*, *text box* dan *rich text box*. Bagian label masing-masing dengan *caption* "*Plain Text*:", "*Cipher Text*:", "Kunci", dan "Algoritma Dekripsi:", "Algoritma Fungsi F", dan "Hasil Keseluruhan". Sedangkan tiga buah *text box* masing-masing berfungsi sebagai input *plain text*, *cipher text*, *kunci*. Empat buah *rich text box* digunakan untuk meletakkan algoritma dekripsi CAST-128 dan algoritma fungsi F serta menampung proses perhitungan dekripsi CAST-128. Tombol "Proses" untuk melakukan proses dekripsi. Tombol "Simpan" digunakan untuk menyimpan hasil perhitungan dalam bentuk *.TXT. Tombol "Cetak" digunakan untuk mencetak hasil perhitungan ke *printer*. Terakhir tombol "Tutup" digunakan untuk menutup *form* ini dan kembali ke menu.



Gambar 4.9 Rancangan *Form* Dekripsi

4.1.1 Perancangan *Module*

Module Function berguna untuk mendeklarasikan semua fungsi yang berhubungan dengan operasi enkripsi dan dekripsi serta generasi kunci. Fungsi-fungsi tersebut termasuk fungsi *rotate* dan *shift*, *add* dan s*ubtract* serta fungsi konversi bilangan seperti desimal ke biner atau sebaliknya, desimal ke heksadesimal atau sebaliknya. Perancangan *module* ini juga melibatkan penulisan kode dan hasilnya semua fungsi-fungsi di atas di simpan dalam satu *file* modMain.bas.

4.2 Implementasi Sistem

Pada implementasi ini diperlukan persiapan perangkat keras (*hardware*) dan persiapan perangkat lunak (*software*).

4.2.1 Persiapan Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan sebagai pendukung untuk membuat perangkat lunak pembelajaran CAST–128 memiliki spesifikasi sebagai berikut:

1. *Processor Pentium III 533 MHz* atau yang lebih tinggi.
2. Memori RAM yang digunakan minimal 128 MB.
3. VGA Card 32 MB.
4. Kapasitas *Harddisk* minimal 10 MB
5. Monitor SVGA dengan Resolusi 1024 × 768.
6. *Keyboard* dan *Mouse*

4.2.2 Persiapan Perangkat Lunak (*Software*)

Sedangkan perangkat lunak yang digunakan sebagai pendukung untuk perangkat lunak pembelajaran CAST–128 membutuhkan:

1. Sistem Operasi *Windows 98/2000/XP*
2. *Active Skin* dari *SoftShape*, Inc.

4.2.3 Cara Menjalankan Program

Untuk menggunakan perangkat lunak ini, jalankan *file* CAST128.EXE, maka akan ditampilkan *form menu*. Untuk mengakses ke bagian tertentu dari program ini, lakukan klik pada tombol yang diinginkan misalnya menu:

1. Operasi pada CAST–128: bagian pembelajaran operasi pada CAST–128
2. Algoritma Enkripsi dan Dekripsi CAST-128: bagian algoritma pembentukan kunci, enkripsi, dan dekripsi CAST-128.
3. Generasi Kunci: untuk melihat pembelajaran algoritma pembentuk sub kunci CAST-128.
4. Enkripsi: untuk melihat pembelajaran algoritma enkripsi CAST-128.
5. Dekripsi: untuk melihat pembelajaran algoritma dekripsi CAST-128.
6. Keluar: untuk keluar dari program ini.

4.2.4 Hasil Program

Hasil Pembentukan Kunci:

```
Kriptografi - Metoda CAST-128 Bit - Key Generation

Key Yang Diinput adalah: 0123456789abcdef
Bentuk dalam Biner:
00110000001100010011001000110011001101000011010100110110000110111001110
00001110010110000101100010011000110110010001100101011001 10
Bentuk dalam Hexadesimal: 30 31 32 33 34 35 36 37 38 39 61 62 63 64 65
66


Proses Round - 1
Nilai dari Z0Z1Z2Z3 Z4Z5Z6Z7 Z8Z9ZAZB ZCZDZEZF (dlm Hexa): 644FC17C
9411AEA3 53FEFA2D 665CB08C

Kunci I       (K1): 74E267A4
```

```
Kunci II      (K2): 3916769E
Kunci III     (K3): 0E0B9F48
Kunci IV      (K4): 6E4DBFDB

Proses Round - 2
Nilai dari X0X1X2X3 X4X5X6X7 X8X9XAXB XCXDXEXF (dlm Hexa): D91E2B5B
317D525D AA87C304 2F707DF2

Kunci V       (K5): 9604A9A9
Kunci VI      (K6): 7FB9F820
Kunci VII     (K7): C2AE11F8
Kunci VIII    (K8): 4A3DF70A

Proses Round - 3

Nilai dari Z0Z1Z2Z3 Z4Z5Z6Z7 Z8Z9ZAZB ZCZDZEZF (dlm Hexa): F7DEAC35
0B18C4BB 5E325C2D 6C7E91F5

Kunci IX      (K9): 30CFA82B
Kunci X      (K10): A423DC8D
Kunci XI     (K11): 83DC03D5
Kunci XII    (K12): 90FBB374

Proses Round - 4

Nilai dari X0X1X2X3 X4X5X6X7 X8X9XAXB XCXDXEXF (dlm Hexa): 7B8F4DB4
206B01CB A5BA49D9 033BE63E

Kunci XIII   (K13): 088E3A5B
Kunci XIV    (K14): DDADA8C4
Kunci XV     (K15): 7026D3A0
Kunci XVI    (K16): 3FE09872

Proses Round - 5

Nilai dari Z0Z1Z2Z3 Z4Z5Z6Z7 Z8Z9ZAZB ZCZDZEZF (dlm Hexa): 249B982E
C9B29A36 63C5C3BE 326462D8
```

```
Kunci XVII   (K17): D685D0D1
Kunci XVIII  (K18): 251E9B8B
Kunci XIX    (K19): A361C65F
Kunci XX     (K20): 3E67DE79

Proses Round - 6

Nilai dari X0X1X2X3 X4X5X6X7 X8X9XAXB XCXDXEXF (dlm Hexa): CC9E838A
8A88068A B17FF7EF BC427F60

Kunci XXI    (K21): 60E29386
Kunci XXII   (K22): AB14330A
Kunci XXIII  (K23): CF8A3C69
Kunci XXIV   (K24): DE963700

Proses Round - 7

Nilai dari Z0Z1Z2Z3 Z4Z5Z6Z7 Z8Z9ZAZB ZCZDZEZF (dlm Hexa): 07B20BEB
2F622D10 6F4D3301 6AC1B7BD

Kunci XXV    (K25): 3DC076B3
Kunci XXVI   (K26): EC86789C
```

```
Kunci XXVII  (K27): FEA07D6E
Kunci XXVIII (K28): C2302A8F

Proses Round - 8

Nilai dari X0X1X2X3 X4X5X6X7 X8X9XAXB XCXDXEXF (dlm Hexa): F20FA25A
346B2D37 71A03BEF 754B1431

Kunci XXIX   (K29): E8F081FE
Kunci XXX    (K30): E9E7E2B9
Kunci XXXI   (K31): B2BC3F95
Kunci XXXII  (K32): BF76FA57
```

Hasil Proses Enkripsi:

```
CAST-128 Bits Encryption Algorithms

Plain Text yang Diinput Adalah: 12345678
Plain Text Dalam Bentuk Biner:
 00110001001100100011001100110100001101010011011000110111001110000

Plain Text Dalam Bentuk Hexadesimal: 3132333435363738
```

```
Key yang Diinput Adalah: 0123456789abcdef

Key Dalam Bentuk Biner:
00110000001100010011001000110011001101000011010100110110000110111001110
00001110010110000101100010011000110110010001100101011001l0

Key Dalam Bentuk Hexadesimal: 30313233343536373839616263646566

Hasil Dari Key (K1 s/d K32):
K(1): 74E267A4
K(2): 3916769E
K(3): 0E0B9F48
K(4): 6E4DBFDB
K(5): 9604A9A9
K(6): 7FB9F820
K(7): C2AE11F8
K(8): 4A3DF70A
K(9): 30CFA82B
K(10): A423DC8D
K(11): 83DC03D5
K(12): 90FBB374
K(13): 088E3A5B
```

```
K(14): DDADA8C4
K(15): 7026D3A0
K(16): 3FE09872
K(17): D685D0D1
K(18): 251E9B8B
K(19): A361C65F
K(20): 3E67DE79
K(21): 60E29386
K(22): AB14330A
K(23): CF8A3C69
K(24): DE963700
K(25): 3DC076B3
K(26): EC86789C
K(27): FEA07D6E
K(28): C2302A8F
K(29): E8F081FE
K(30): E9E7E2B9
K(31): B2BC3F95
K(32): BF76FA57

Tabel Hasil Enkripsi:
```

Rounds	L(i)	R(i)	F(i)
0	31323334	35363738	-
1	35363738	E443424C	D5717178
2	E443424C	EB93CA22	DEA5FD1A
3	EB93CA22	43E81B0E	A7AB5942
4	43E81B0E	B12C1906	5ABFD324
5	B12C1906	C3C62A2A	802E3124
6	C3C62A2A	93E9594E	22C54048
7	93E9594E	CC32C9C6	0FF4E3EC
8	CC32C9C6	4F4DB7E8	DCA4EEA6
9	4F4DB7E8	C2FFCAC2	0ECD0304
10	C2FFCAC2	92984F89	DDD5F861
11	92984F89	94D44254	562B8896
12	94D44254	01250C45	93BD43CC
13	01250C45	43F68E13	D722CC47
14	43F68E13	D6464E0F	D763424A
15	D6464E0F	18FE11B4	5B089FA7
16	18FE11B4	B0B856D3	66FE18DC

```
------------------------------------------------------------------------
CipherText (Dalam Hexa): B0B856D318FE11B4
CipherText (Dalam Bentuk ASCII): °¸VÓþ´
```

Hasil Proses Dekripsi:

```
CAST-128 Bits Decryption Algorithms

CipherText yang Diinput Adalah: °¸VÓ_þ_´
CipherText Dalam Bentuk Biner:
10110000101110000101011011010011000110001111111000010001101101000
CipherText Dalam Bentuk Hexadesimal: B0B856D318FE11B4
Key yang Diinput Adalah: 0123456789abcdef
Key Dalam Bentuk Biner:
00110000001100010011001000110011001101000011010100110110001101111001110
00001110010110000101100010011000110110010001100101011001100110
Key Dalam Bentuk Hexadesimal: 30313233343536373839616263646566

Hasil Dari Key (K1 s/d K32):
K(1): 74E267A4
K(2): 3916769E
K(3): 0E0B9F48
K(4): 6E4DBFDB
K(5): 9604A9A9
K(6): 7FB9F820
```

```
K(7): C2AE11F8
K(8): 4A3DF70A
K(9): 30CFA82B
K(10): A423DC8D
K(11): 83DC03D5
K(12): 90FBB374
K(13): 088E3A5B
K(14): DDADA8C4
K(15): 7026D3A0
K(16): 3FE09872
K(17): D685D0D1
K(18): 251E9B8B
K(19): A361C65F
K(20): 3E67DE79
K(21): 60E29386
K(22): AB14330A
K(23): CF8A3C69
K(24): DE963700
K(25): 3DC076B3
K(26): EC86789C
K(27): FEA07D6E
K(28): C2302A8F
K(29): E8F081FE
K(30): E9E7E2B9
K(31): B2BC3F95
K(32): BF76FA57

Tabel Hasil Dekripsi:
```

Rounds	L(i)	R(i)	F(i)
0	B0B856D3	18FE11B4	-
1	18FE11B4	D6464E0F	66FE18DC
2	D6464E0F	43F68E13	5B089FA7
3	43F68E13	01250C45	D763424A
4	01250C45	94D44254	D722CC47
5	94D44254	92984F89	93BD43CC
6	92984F89	C2FFCAC2	562B8896
7	C2FFCAC2	4F4DB7E8	DDD5F861
8	4F4DB7E8	CC32C9C6	0ECD0304
9	CC32C9C6	93E9594E	DCA4EEA6
10	93E9594E	C3C62A2A	0FF4E3EC
11	C3C62A2A	B12C1906	22C54048
12	B12C1906	43E81B0E	802E3124
13	43E81B0E	EB93CA22	5ABFD324
14	EB93CA22	E443424C	A7AB5942
15	E443424C	35363738	DEA5FD1A
16	35363738	31323334	D5717178

```
PlainText (Dalam Hexa): 3132333435363738
PlainText (Dalam Bentuk ASCII): 12345678
```

[5] Kesimpulan

Dari hasil penelitian yang dibuat, maka peneliti dapat mengambil beberapa kesimpulan diantaranya dengan adanya perangkat lunak pembelajaran CAST−128 yang menampilkan cara perhitungan proses pembentukan kunci, proses enkripsi, dan dekripsi CAST-128 secara bertahap maka algoritma CAST−128 lebih mudah dipahami,. Proses enkripsi dan dekripsi menggunakan bentuk *form* yang sama karena enkripsi dan dekripsi CAST-128 sama hanya berbeda dalam penggunaaan fungsi *f* dan variabel. Hasil *output* dari program ini dapat dipergunakan untuk menganalisa algoritma CAST−128.

Untuk pengembangan lebih lanjut dari program pembelajaran CAST-128 ini maka dapat diberikan saran agar program dapat ditambahkan pembelajaran untuk metode CAST-256. Perangkat lunak pembelajaran ini sebaiknya ditambahkan dengan fasilitas suara yang akan menerangkan semua proses yang berlangsung. Menambahkan efek animasi yang lebih banyak pada perangkat lunak pembelajaran ini sehingga tampilannya lebih menarik dan mudah dipahami. Penambahan *video* dari seorang narator layaknya program pembelajaran umumnya membuat proses pemahaman algoritma CAST-128 lebih mudah.

Daftar Pustaka:

[1] Adams, C. M., ***A Formal and Practical Design Procedure for Substitution – Permutation Network Cryptosystems***, Ph.D. Thesis, *Department of Electrical Engineering*, Queen's University, 1990.

[2] Adams, C. M. and S. E. Tavares**,** ***Designing S-Boxes for Ciphers Resistant to Differential Cryptanalysis***, Proceedings of the 3rd Symposium on the State dan Progress of Research in Cryptography, Rome, Italy, Feb., 1993, pp. 181-190.

[3] Adams, C. M., ***Simple and Effective Key Scheduling for Symetric Ciphers***, in Workshop Record of the Workshop on Selected Areas in Cryptography (SAC 94), May 5–6, 1994, pp. 129–133.

[4] Adams, C. M., ***Designing DES-Like Ciphers with Guaranteed Resistant to Differential and Linear Attacks***, in Workshop Record of the Workshop on Selected Areas in Cryptography (SAC 95), May 18–19, 1995, pp. 133–144.

[5] Leman, **Metodologi Pengembangan Sistem**, PT. Elex Media Komputindo, Jakarta, 1998.

[6] Murach, Mike & Associate, Inc, 1999, **Murach Visual Basic 6**, Tech Publications Pte Ltd, Singapore.

[7] Cryptography, **http://en.wikipedia.org/cryptography**, tanggal akses 22 Juli 2011.

[8] Sudirman, I., 2003, **Perkembangan Software Komputer**, Kuliah Pengantar Ilmu Komputer.com.