KRIPTOGRAFI DAN STEGANOGRAFI PADA CITRA *TRUE COLOR* DENGAN METODE BLOWFISH DAN LSB

Mahmud Yunus
Dosen STMIK Pradnya Paramita Malang
myoenoes@gmail.com

ABSTRAK

Kelemahan dari metode steganografi LSB adalah pesan rahasia yang disisipkan dapat dengan mudah diambil dengan menggunakan metode LSB pula. Sehingga perlu ada cara lain untuk mengamankan pesan rahasia agar tidak mudah dibaca atau dipahami oleh pihak yang tidak berhak. Pesan rahasia tersebut perlu disandikan (di-enkripsi) terlebih dahulu menggunakan teknik kriptografi sebelum disisipkan, agar pesan rahasia yang dimaksud tidak mudah dibaca atau dipahami oleh pihak yang tidak berhak. Salah satu algoritma kriptografi yang dapat digunakan adalah algoritma Blowfish. Algoritma Blowfish merupakan metode enkripsi dalam golongan symmetric cryptosystem, yaitu algoritma kriptografi yang menggunakan kunci (key) yang sama untuk proses enkripsi dan dekripsi pesan

Fokus dari penelitian ini adalah bagaimana menerapkan teknik kriptografi pesan digital menggunakan metode kombinasi Blowfish dan MD5, serta penyisipan dan pengambilan pesan terenkripsi (ciphertext) pada media citra true color 24 bit dengan teknik steganografi LSB (Least Significant Bit).Hasil akhir yang ingin dicapai dari penelitian ini adalah terciptanya sebuah aplikasi perangkat lunak yang dapat digunakan sebagai alat untuk menyebunyikan pesan terenkripsi pada LSB (Least Significant Bit) dari berkas citra digital true color 24 bit, dengan metode kriptografi Blowfish dan MD5 untuk mengamankan data dan informasi penting dari tidakan ilegal pihak yang tidak berhak

Berdasarkan hasil pengujian yang telah dilakukan terhadap aplikasi perangkat lunak yang menerapkan teknik kriptografi pesan digital menggunakan metode kombinasi Blowfish dan MD5, serta penyisipan dan pengambilan pesan terenkripsi pada media citra true color 24 bit dengan teknik steganografi LSB (Least Significant Bit), diperoleh hasil; (1) enkripsi terhadap seluruh teks pesan rahasia dengan metode MD5 dan Blowfish, diperoleh tingkat keberhasilan sebesar 100% untuk menghasilkan teks pesan rahasia terenkripsi (ciphertext); (2) penyisipan dan pengambilan teks pesan rahasia terenkripsi (ciphertext) ke/dari dalam stego object berupa citra digital true color, 100% berhasil dilakukan dengan tampilan gambar yang mirip (tidak nampak perubahan yang signifikan) dengan tampilan citra digital sebelum dilakukan penyisipan, sehingga keberadan teks pesan rahasia di dalam stego object tidak mudah disadari keberadaannya oleh pihak lain; dan (3) proses dekripsi terhadap teks pesan rahasia terenkripsi yang diambil dari stego object, 100% berhasil dilakukan untuk menghasilkan kembali teks pesan rahasia asli (plaintext).

Kata kunci: kriptografi, steganografi, MD5 dan Blowfish.

I. PENDAHULUAN

Berbagai cara perlindungan dari tindakan pencurian data dan informasi digital yang tersimpan dan ditransmisikan dalam jaringan komputer, telah banyak dikenalkan pada masyarakat luas. Salah satu metode pengamanan data dan infromasi (pesan) digital adalah dengan menggunakan teknik steganografi. Steganografi adalah ilmu dan seni menyembunyikan pesan rahasia di dalam pesan lain sehingga keberadaan pesan rahasia tersebut tidak dapat diketahui (Munir, 2006). Metode steganografi yang sederhana dan mudah diimplementasikan adalah metode *Least Significant Bit* (LSB), yaitu penggunaan bit-bit yang tidak terlalu berpengaruh (penting) pada berkas (*file*) tertentu, seperti berkas suara, citra dan video digital, untuk tempat menyembunyikan pesan.

Kelemahan dari metode LSB adalah pesan rahasia yang disisipkan dapat dengan mudah diambil dengan menggunakan metode LSB pula. Sehingga perlu ada cara lain untuk mengamankan pesan rahasia agar tidak mudah dibaca atau dipahami oleh pihak yang tidak berhak. Pesan rahasia tersebut perlu disandikan (di-enkripsi) menggunakan teknik kriptografi sebelum disisipkan, agar pesan rahasia yang dimaksud tidak mudah dibaca atau dipahami

oleh pihak yang tidak berhak. kriptografi merupakan ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas data serta otentikasi (Menezez, 1997).

Algoritma kriptografi yang dapat digunakan adalah algoritma Blowfish. Blowfish merupakan algoritma kriptografi kunci simetrik *cipher* blok dengan panjang blok tetap sepanjang 64 bit. Blowfish memanfaatkan teknik manipulasi bit dan teknik pemutaran ulang dan pergiliran kunci yang dilakukan sebanyak 16 kali putaran. Algoritma utama terbagi menjadi dua sub-algoritma utama, yaitu bagian ekspansi kunci dan bagian enkripsi/dekripsi data. Pengekspansian kunci dilakukan pada saat awal dengan masukan sebuah kunci dengan panjang 32 hingga 448 bit, dan keluaran adalah sebuah larik sub-kunci dengan total 4.168 bit. Bagian enkripsi-dekripsi data terjadi dengan memanfaatkan perulangan 16 kali terhadap jaringan feistel.

Keamanan data atau informasi (pesan) yang dienkripsi menggunakan algoritma Blowfish dapat ditingkatkan lagi dengan mengkombinasikannya bersama metode MD5. MD5 digunakan untuk meng-*hash* kunci (*key*) dari proses enkripsi/dekripsi pesan berdasarkan algoritma Blowfish.

Salah satu berkas (*file*) yang dapat digunakan untuk media steganografi adalah berkas citra digital *true color* 24 bit. Setiap pixel pada citra true color 24 bits, memiliki 3 lapis (*layer*) intensitas warna berukuran 8 bit yang merepresentasikan level warna Merah (*Red*), Hijau (*Green*) dan Biru (*Blue*). Sebagian dari 8 bit intensitas warna tersebut, terdapat bit-bit paling kanan yang bernilai kecil dan dianggap tidak terlalu perpengaruh pada nilai keseluruhan bit (*Least Significant Bit/LSB*). Pada bit-bit LSB dari berkas citra true color 24 bit inilah nantinya data atau informasi digital (pesan) yang telah dikriptografi dapat disisipkan, sebagai implementasi dari teknik steganografi.

II. LANDASAN TEORI

2.1 Konsep Dasar Kriptografi

Kriptografi merupakan ilmu sekaligus seni untuk menjaga keamanan pesan (Schneier, 1996). Sedangkan menurut Menezez (1997), kriptografi merupakan ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas data serta otentikasi. Terdapat beberapa terminologi istilah yang penting dalam kriptografi, diantaranya adalah :

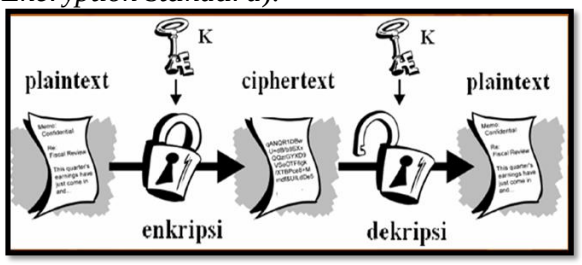
1. Pesan *Plaintext* dan *Ciphertext*. Pesan (*message*) adalah data atau informasi yang dapat dibaca dan dimengerti maknanya. Pesan asli disebut *plaintext* atau teks-jelas (*cleartext*). Sedangkan pesan yang sudah disandikan disebut *ciphertext*
2. Pengirim dan Penerima. Komunikasi data melibatkan pertukaran pesan antara dua entitas. Pengirim (*sender*) adalah entitas yang mengirim pesan kepada entitas lainnya. Penerima (*receiver*) adalah entitas yang menerima pesan.
3. Penyadap (*eavesdropper*) merupakan orang yang mencoba menangkap pesan selama ditransmisikan.
4. Kriptanalisis dan Kriptologi. Kriptanalisis (*cryptanalysis*) adalah ilmu dan seni untuk memecahkan *chipertext* menjadi *plaintext* tanpa mengetahui kunci yang digunakan. Pelakunya disebut kriptanalis. Kriptologi (*cryptology*) adalah studi mengenai kriptografi dan kriptanalisis.
5. Enkripsi dan Dekripsi. Enkripsi (*encryption/enciphering*) merupakan proses menyandikan *plaintext* menjadi *ciphertext*, sedangkan Dekripsi (*decryption/ deciphering*) merupakan proses merubah *chipertext* menjadi *plaintext* kembali.
6. *Cipher* dan Kunci. Algoritma kriptografi disebut juga *cipher* yaitu aturan untuk *enchipering* dan *dechipering*, atau fungsi matematika yang digunakan untuk proses enkripsi dan dekripsi. Kunci (*key*) adalah parameter yang digunakan untuk transformasi *enciphering* dan *dechipering*. Kunci biasanya berupa *string* atau deretan bilangan.

Semua fungsi kriptografi harus memiliki sifat *reversibility* (balik ke asal), yaitu mampu mengembalikan *ciphertext* hasil enkripsi kembali ke *plaintext* melalui proses dekripsi. Kemampuan *reversibility* pada hampir semua metode pada algoritma kunci simetrik mengandalkan kemampuan operasi kebalikan (*reverse operation*). Metode ini berintikan membalik semua operasi yang ada, yaitu dengan melakukan operasi yang berlawanan. Terdapat pula beberapa algoritma kunci simetrik blok cipher memiliki sub metode yang tidak bersifat

reversible jika berdiri sendiri seperti metode Expand dan Filter. Metode tersebut akan bersifat *reversible* jika ditanamkan pada metode Jaringan Feistel.

2.2 Kriptografi Kunci Simetri

Kriptografi kunci simetri menggunakan kunci yang sama untuk proses enkripsi dan dekripsi. Keamanan sistem kriptografi simetri terletak pada kerahasiaan kuncinya. Kriptografi simetri ini, diantaranya adalah DES (*Data Encryption Standard*), Blowfish, Twofish, Triple-DES, IDEA, Serpent, AES (*Advanced Encryption Standard*).



Gambar 2.1: Penggunaan kunci pada algoritma kriptografi simetris

Algoritma kriptografi (*cipher*) simetri terbagi dalam menjadi dua bagian, yaitu: (1) *Cipher* aliran (*stream cipher*). Algoritma kriptografi beroperasi pada *plaintext* dalam bentuk bit tunggal. Cipher aliran memproses satu bit pesan sekali dalam satu waktu, sehingga rangkaian bit dienkripsikan/didekripsikan bit per bit; dan (2) *Cipher* blok (*block cipher*).

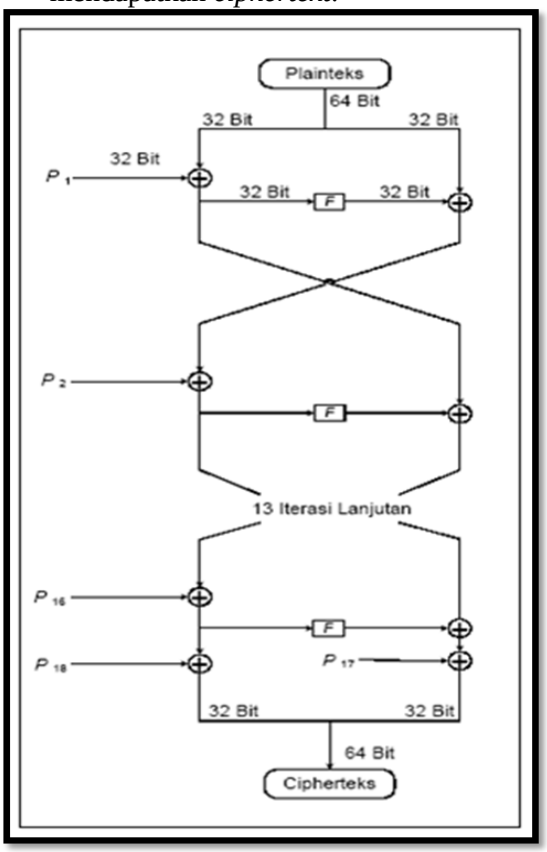
Algoritma kriptografi beroperasi pada *plaintext* dalam bentuk blok bit. Rangkaian bit dibagi menjadi blok-blok bit yang panjangnya sudah ditentukan sebelumnya. Ukuran blok yang umum dipakai adalah 64 bit.

2.3 Algoritma Kriptografi Blowfish

Blowfish adalah algoritma kriptografi yang menggunakan blok *cipher* 64-bit dan memiliki sebuah kunci yang panjangnya bervariasi antara 32-bit sampai 448 bit. Algoritma Blowfish terdiri dari dua bagian yaitu pembangkitan sub-kunci (*key expansion*) dan blok data yang akan dienkripsi. Enkripsi Data terdiri dari iterasi fungsi sederhana (*feistel network*) sebanyak 16 kali putaran dengan data masukannya adalah 64 bit elemen data (blok *cipher*).

Semua operasi yang dilakukan adalah operasi penambahan (*addition*) dan XOR pada variabel 32 bit. Sub kunci pada algoritma terdiri dari 18 sub kunci (*sub key*) berukuran 32 bit yang tersusun dalam array (misalkan P → P_1, P_2, P_3,... P_{18}). Kunci-kunci ini harus dihitung atau dibangkitkan terlebih dahulu sebelum dilakukan enkripsi atau dekripsi data. Jika dimisalkan blok data (*cipher*) sebagai variabel X dan sub key sebagai variabel array P, maka langkah-langkah enkripsi sebagai berikut:

1. Bagi X berukuran 64 bit menjadi dua bagian misalkan X_L dan X_R yang masing-masing berukuran 32 bit.
2. Lakukan perulangan (iterasi) yang melibatkan 16 sub kunci pertama
   For i := 1 to 16 do begin
   $X_L := X_L$ XOR P_i
   $X_R := F(X_L)$ XOR X_R
   Tukarkan nilai X_L dan X_R
   End of For
3. Setelah iterasi ke-16, tukarkan nilai X_L dan X_R lagi untuk membatalkan pertukaran terakhir.
4. Lakukan operasi berikut:
   $X_R := X_R$ XOR P_{17}
   $X_L := X_L$ XOR P_{18}
5. Gabungkan kembali X_L dan X_R untuk mendapatkan *ciphertext*.

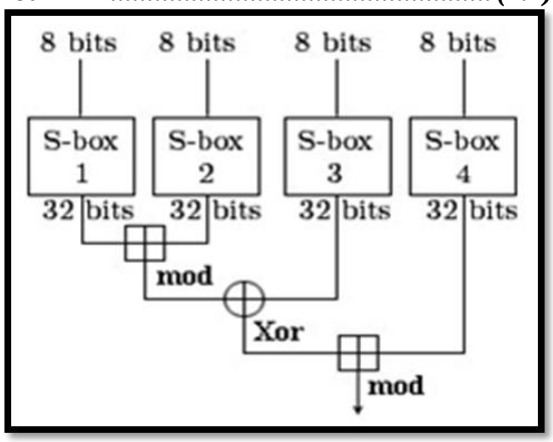


Gambar 2.2: Jaringan feistel untuk enkripsi

pada algoritma Blowfish

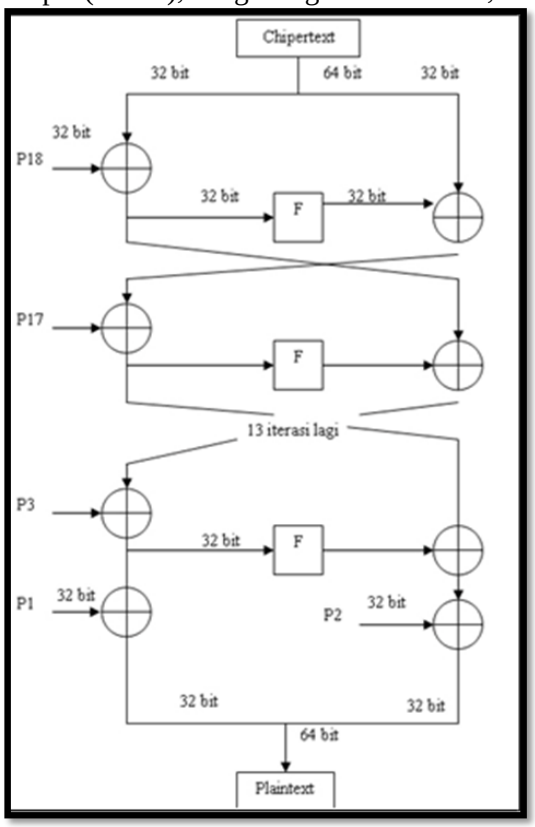
Fungsi F merupakan fungsi yang membagi X_L menjadi empat bagian misalkan a, b, c dan d, masing-masing berukuran 8-bit. Persamaan fungsi F adalah sebagai berikut;

$$F(X_L) = ((S_{1,a} + S_{2,b} \bmod 2^{32}) \text{ xor } S_{3,c}) + S_{4,c} \bmod 2^{32} \quad \text{..... (2.1)}$$



Gambar 2.3: Fungsi F pada algoritma Blowfish

Proses dekripsi pada algoritma Blowfish dilakukan dengan cara membalik urutan proses enkripsi (*revers*), dengan algoritma berikut;



Gambar 2.4: Jaringan feistel untuk dekripsi pada algoritma Blowfish

Cara menghitung atau membangkitkan sub-key pada algortima Blowfish adalah sebagai berikut:

1. Inisialisasi P-array yang pertama dan juga empat S-box, berurutan, dengan string yang telah pasti. String tersebut terdiri dari digit-digit hexadesimal.
2. XOR P1 dengan 32-bit pertama dari kunci, XOR P2 dengan 32-bit kedua dari kunci, dan seterus- nya untuk seluruh bit dari kunci (sampai P18). Ulangi siklus seluruh bit kunci secara berurutan sampai seluruh P-array telah di-XOR-kan dengan bit-bit kunci
3. Enkripsikan string yang seluruhnya nol sebanyak 64 bit dengan algoritma Blowfish menggunakan sub key baru dari langkah 2, gantikan seluruh elemen dari P-Box dan kemudian keempat S-box secara berurutan dengan hasil keluaran algoritma Blowfish yang terus menerus berubah.

2.4 Algoritma MD5

MD5 adalah fungsi *hash* satu-arah yang dibuat oleh Ron Rivest. Algoritma MD5 menerima masukan berupa pesan dengan ukuran sembarang dan menghasilkan *message digest* yang panjangnya 128 bit (Munir, 2006). Urutan langkah pembuatan *message digest* secara garis besar adalah sebagai berikut:

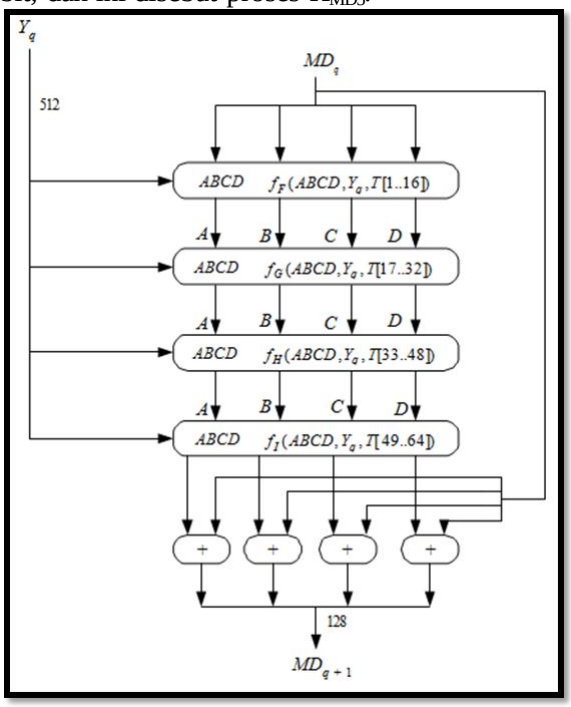
1. Penambahan bit-bit pengganjal (*padding bits*). Pesan ditambah dengan se jumlah bit pengganjal sedemikian sehingga panjang pesan (dalam satuan bit) kongruen dengan 448 modulo 512. Ini berarti panjang pesan setelah ditambahi bit-bit pengganjal adalah 64 bit kurang dari kelipatan 512. Angka 512 ini muncul karena *MD5* memperoses pesan dalam blok-blok yang berukuran 512 bit. Panjang bit-bit pengganjal adalah antara 1 sampai 512. Bit-bit pengganjal terdiri dari sebuah bit 1 diikuti dengan sisanya bit 0.
2. Penambahan nilai panjang pesan semula. Pesan yang telah diberi bit-bit pengganjal selanjutnya ditambah lagi dengan 64 bit yang menyatakan panjang pesan semula.Jika panjang pesan > 2^{64} maka yang diambil adalah panjangnya dalam modulo 2^{64}. Jika panjang pesan semula adalah *K* bit, maka 64 bit yang ditambahkan menyatakan *K* modulo 2^{64}.
3. Inisialisasi penyangga (*buffer*) *MD. MD5*

membutuhkan 4 buah penyangga (*buffer*) yang masing- masing panjangnya 32 bit. Total panjang penyangga adalah 4 x 32 = 128 bit. Keempat penyangga ini diberi nama *A*, *B*, *C*, dan D untuk menampung hasil antara dan hasil akhir. Setiap penyangga diinisialisasi dengan nilai-nilai (dalam notasi HEX) sebagai berikut:

A = 01234567
B = 89ABCDEF
C = FEDCBA98
D = 76543210

4. Pengolahan pesan dalam blok berukuran 512 bit.

Pesan dibagi menjadi *L* buah blok yang masing-masing panjangnya 512 bit (Y_0 sampai Y_{L-1}). Setiap blok 512-bit diproses bersama dengan penyangga *MD* menjadi keluaran 128-bit, dan ini disebut proses H_{MD5}.



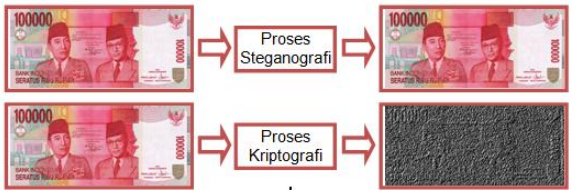
Gambar 2.5: Pengolahan pesan dalam blok 512 bit pada proses H_{MD5}

Proses H_{MD5} terdiri dari 4 buah putaran, dan masing-masing putaran melakukan operasi dasar MD5 sebanyak 16 kali dan setiap operasi dasar memakai sebuah elemen T. Jadi setiap putaran memakai 16 elemen Tabel *T*. Pada Gambar 5, Y_q menyatakan blok 512-bit ke-*q* dari pesan yang telah ditambah bit-bit pengganjal dan tambahan 64 bit nilai panjang pesan semula. MD_q adalah nilai *message digest* 128-bit dari proses H_{MD5} ke-q. Pada awal proses, MD_q berisi nilai inisialisasi penyangga MD.

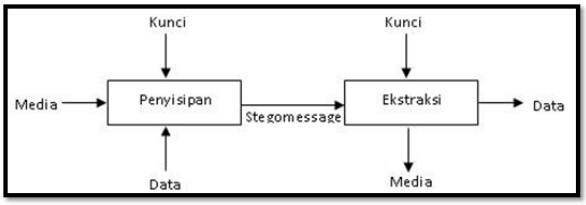
2.5 Steganografi

Steganografi merupakan seni dan ilmu untuk berkomunikasi sedemikian rupa bahwa kehadiran pesan tidak dapat dideteksi (Cachin, 1998). Steganografi merupakan seni penyembunyian pesan ke dalam pesan lainnya sedemikian rupa sehingga orang lain tidak menyadari ada sesuatu di dalam pesan tersebut.

Steganografi dan kriptografi dapat digunakan untuk menjamin kerahasiaan data. Perbedaannya adalah dengan kriptografi, siapa pun dapat melihat bahwa kedua belah pihak berkomunikasi secara rahasia. Pada steganografi menyembunyikan keberadaan pesan rahasia pada media lain dan tidak ada yang bisa melihat bahwa kedua belah pihak berkomunikasi secara rahasia. Hal ini membuat steganografi cocok untuk beberapa tugas yang tidak bisa dilakukan dengan enkripsi (Cummins, 2004).



Gambar 2.6: Perbedaan proses Steganografi dengan Kriptografi

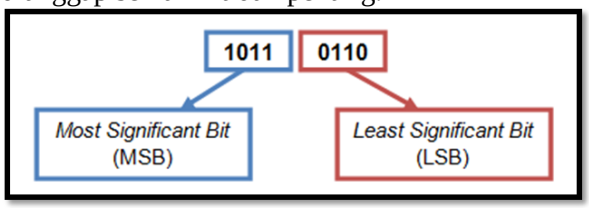


Gambar 2.7: Penyisipan & ekstraksi data dalam media penampung pada sistem Steganografi

2.6 Metode Least-Significant Bit

Steganografi dengan metode LSB (*Least Significant Bit*) dapat diartikan sebagai proses penyembunyian data dilakukan dengan mengganti bit-bit data yang tidak terlalu berpengaruh di dalam segmen citra dengan bit-bit data rahasia (Cummins, 2004). Pada susunan bit di dalam ukuran 1 Byte (8 bit), ada bit yang paling berarti (*most significant bit* atau *MSB*) dan bit yang kurang berarti (*least significant bit* atau *LSB*). Semakin ke kiri posisi bit dalam urutan 8 bit (1 Byte), maka semakin penting nilainya (nilainya semakin besar). Sebaliknya, semakin ke kanan posisi bit dalam

urutan 8 bit, maka nilainya semakin kecil dan dianggap semakin tidak penting.



Gambar 2.8: Pembagian bit-bit MSB dan LSB untuk Steganografi

Bit-bit dari rangkaian 8 bit (1 Byte) data penampung yang cocok untuk diganti dengan bit-bit pesan rahasia adalah bit LSB, sebab perubahan tersebut hanya mengubah nilai Byte data penampung maksimal sebesar +/- 15 poin. Besar kecilnya perubahan nilai Byte data penampung tergantung pada 2 hal yaitu; (1) berapa banyak bit LSB yang digunakan, dan (2) nilai bit-bit pesan rahasia yang menggantikan nilai bit-bit data penampung. Bila 4 bit LSB yang digunakan, maka perubahan nilainya maksimal +/- 15 poin (antara 0000 s/d 1111). Semakin sedikit bit LSB yang digunakan, maka semakin kecil perubahan nilai byte data penampung steganografi.

Misalkan huruf "A" dengan kode ASCII **65** (biner: **0100 0001**), akan disisipkan kedalam sebuah pixel pada citra digital 24 bit (citra RGB *true color*) dengan komposisi nilai derajat warna merah (*Red*) sebesar **213** (biner: **1101 0101**), hijau (*Green*) sebesar **155** (biner: **1001 1011**) dan biru (*lue*) sebesar **195** (biner: **1100 0011**), maka skenario penyisipannya dapat dilakukan dengan cara sebagai berikut;

1. Ambil 4 bit pertama dari huruf "A" (**0100**) untuk menggantikan 4 bit LSB layer warna merah, sehingga nilai derajat warna merah berubah -1 poin dari **213** (biner: **1101 0101**) menjadi **212** (**1101 0100**).
2. Ambil 2 bit berikutnya (bit 5 dan 6) dari huruf "A" (**00**) untuk menggantikan 2 bit LSB layer warna hijau yang bersesuaian (bit 5 dan 6), sehingga nilai derajat warna hijau berubah -8 poin dari **155** (biner: **1001 1011**) menjadi **147** (**1001 0011**).
3. Ambil 2 bit terakhir (bit 7 dan 8) dari huruf "A" (**01**) untuk menggantikan 2 bit LSB terakhir layer warna biru (bit 7 dan 8) , sehingga nilai derajat warna biru berubah -2 poin dari **195** (biner: **1100 0011**) menjadi **193** (biner: **1100 0001**).

Berdasarkan skenario penyisipan tersebut, perubahan nilai derajat warna pada setiap layernya tidak lebih dari +/- 15 yang tidak akan nampak perubahannya oleh penglihatan mata manusia.

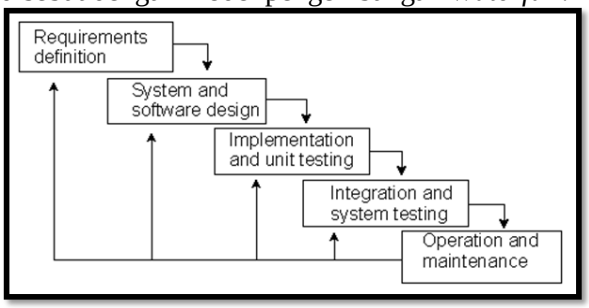
III. METODE PENELITIAN

3.1 Jenis dan Tahapan Penelitian

Hasil akhir dari penelitian ini adalah terciptanya suatu produk berupa software aplikasi komputer yang dapat digunakan untuk mengamankan data dan informasi penting dari tidakan ilegal pihak yang tidak berhak. Software aplikasi komputer yang dibangun menerapkan teknik kriptografi pesan digital menggunakan metode kombinasi Blowfish dan MD5. Pesan yang telah terinkripsi disisipkan (disembunyikan) pada media citra *true color* 24 bit dengan teknik steganografi LSB (*Least Significant Bit*).

Pendekatan yang sesuai dengan tuntutan hasil akhir tersebut, adalah model pendekatan penelitian dan pengembangan (*research and development*). Penelitian dan pengembangan ini merupakan suatu siklus yang berlapis, berulang, dan berkesinambungan, mulai dari penelitian deskriptif, pengembangan model awal (*prototype)* sebagai produk pendahuluan (*preliminari form*), pengujian kelayakan model oleh pakar, pengembangan model untuk menjadi produk, pengujian produk, hingga dihasilkannya suatu produk yang dapat digunakan secara layak.

Suatu model pengembangan perangkat lunak yang dilakukan secara sistematis dan sekuensial linier mulai dari tingkat dan kemajuan pada tahap analisis, desain, pengkodean program, pengujian (*testing*) sistem hingga pengoperasian dan pemeliharaan sistem disebut dengan model pengembangan *Waterfall*.



Gambar 3.1: Pengembangan perangkat lunak dengan model Waterfall

Rancang strategi penelitian meliputi;

1. Melakukan penelitian pendahuluan untuk mengetahui kelayakan dan relevansi masalah penelitian dengan kondisi dan kekinian perkembangan ilmu pengetahuan.
2. Membangun konsep penelitian dengan mendefinisikan kebutuhan penelitian, merumuskan masalah dan tujuan penelitian.
3. Mengumpulkan data dan teori tentang kriptografi khususnya mengenai algoritma MD5 dan Blowfish, serta teori steganografi pada citra digital *true color* 24 bit.
4. Melakukan analisa dan pemecahan masalah terhadap data yang terkumpul.
5. Merancang antar muka perangkat lunak.
6. Merancang algoritma enkripsi dan dekripsi kunci dengan metode kriptografi MD5
7. Merancang algoritma kriptografi pesan (*message*) dengan metode Blowfish.
8. Merancang algoritma penyisipan dan pembacaan data citra digital *true color* 24 bit dengan format BMP (*Bitmap Image*) dengan metode LSB (*Least Significant Bit*).
9. Melakukan pengkodean program untuk membangun mesin kriptografi Blowfish, serta mesin penyisipan dan pengambilan data ke dalam/dari data citra digital dengan metode LSB
10. Melakukan pengujian dan evaluasi kelayakan terhadap aplikasi perangkat lunak atau sistem yang telah dibangun hingga pada level kesalahan (*error*) sistem di bawah 3%.

3.2 Data Penelitian

Data yang digunakan dalam penelitian ini adalah data primer yang terdiri dari

1. Sejumlah data pesan digital berupa berkas dalam format teks (*text file*) berserta data kunci kriptografi yang akan dienkripsi/didekripsi dengan metode Blowfish dan MD5, yang kemudian disisipkan ke dalam media berkas citra digital *true color* 24 bit sebagai pesan rahasia yang tersembunyi.
2. Sejumlah berkas (*file*) citra digital *true color* 24 bit dalam format Bitmap (BMP) sebagai media penampung (*stego object*) pesan rahasia/tersembunyi (*hidden message*) dalam teknik steganografi.

3.3 Rancangan Algoritma Perangkat Lunak

Secara garis besar rancangan algoritma yang dibagun terdiri dari langkah-langkah;

1. Inisialisasi metode dan panjang blok *cipher*, *hash* serta kunci kriptografi.
2. Pemilihan berkas teks pesan rahasia yang akan dienkripsi/didekripsi dan disisipkan/diurai ke file citra digital 24 bit.
3. Pemilihan berkas citra digital 24 bit sebagai media penyembunyian pesan (*stego object*)
4. Pengisian kunci kriptografi untuk menghasilkan *ciphertext* dari *plaintext* atau sebaliknya.
5. Pengenkripsian *plaintext* menjadi *ciphertext* dan sebaliknya
6. Penyisipan *ciphertext* ke dalam *stego object* dengan metode steganografi LSB dan sebaliknya, yaitu penguraian/pengambilan *ciphertext* dari *stego object*.
7. Proses selesai

IV. HASIL DAN PEMBAHASAN

4.1 Pengujian Sistem

Data yang digunakan dalam pengujian sistem ini terdiri dari 7 *file* citra digital 24 bit dengan format BMP (*Bitmap Image*) sebagai media penampung (*stego object*) untuk menyembunyikan pesan rahasia, dan 5 pesan rahasia dalam format *text file*. Skenario pengujian dengan cara mencoba semua pesan rahasia (*text file*) untuk di-enkripsi/di-dekripsi dan disisipkan/diambil satu per satu ke/dari dalam *file* citra digital dengan kunci kriptografi sepanjang 10 karakter.

Tabel 4.1: Data file citra digital untuk pengujian sistem

No.	Nama File, Ukuran dan Dimensi	Tampilan Citra
1	1Ribu.bmp 1,390 kb 1024 x 463 Pixel	
2	2Ribu.bmp 1,384 kb 1024 x 461 Pixel	
3	5Ribu.bmp 1,381 kb 1024 x 460 Pixel	
4	10Ribu.bmp1 ,363 kb 1024 x 454 Pixel	

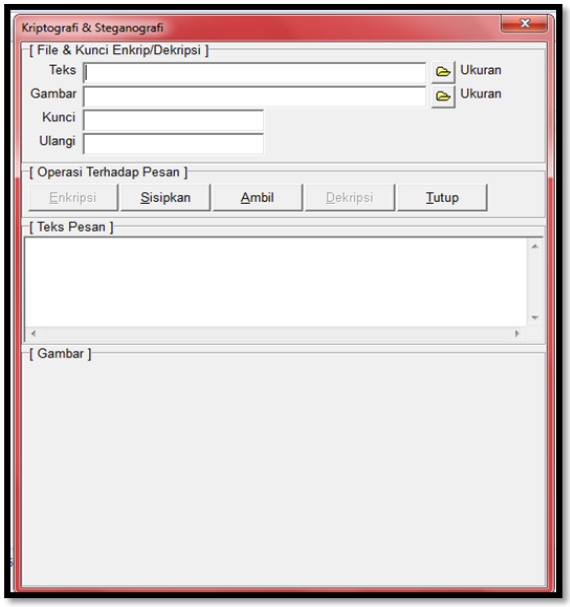
5	20Ribu.bmp1 ,321 kb 1024 x 440 Pixel	
6	50Ribu.bmp1 ,312 kb 1024 x 437 Pixel	
7	100Ribu.bmp 1,345 kb 1024 x 448 Pixel	

Tabel 4.2: Data file teks untuk pengujian sistem

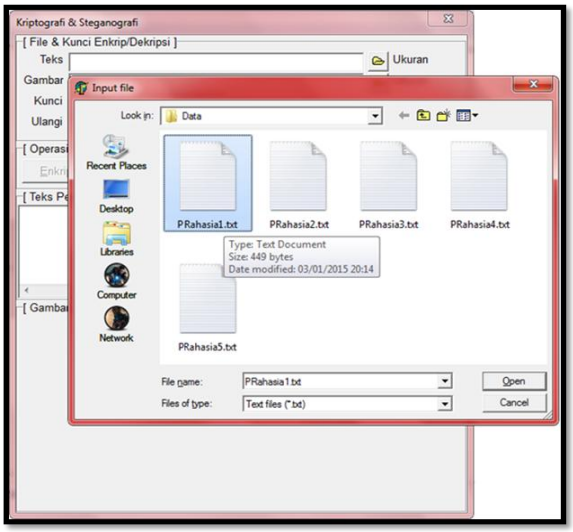
No.	Nama File	Ukuran	Jumlah Karakter
1	Prahasia1.txt	1 kb	449
2	Prahasia2.txt	11 kb	10.830
3	Prahasia3.txt	5 kb	4.496
4	Prahasia4.txt	42 kb	42.860
5	Prahasia5.txt	14 kb	13.668

4.1.1 Pemilihan Berkas Pesan Rahasia

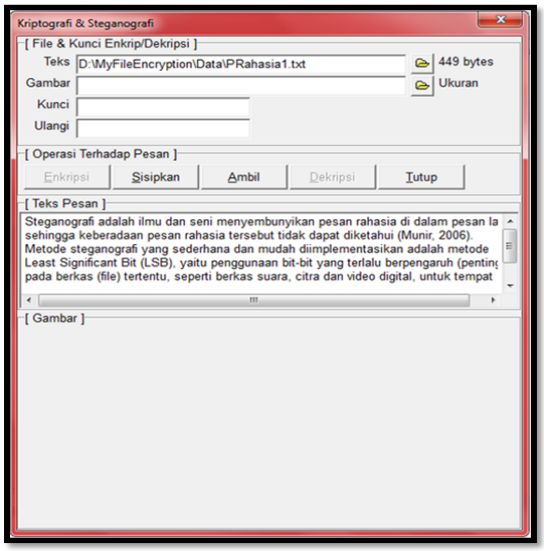
Proses pengujian sistem diawali dengan melakukan pemilihan *file* teks sebagai berkas pesan rahasia yang akan dienkripsi dengan metode MD5 dan Blowfish. Pemilihan *file* teks pesan rahasia dilakukan dengan cara menuliskan alamat dan nama file teks yang dimaksud pada bagian masukan (*input*) Teks.



Gambar 4.1: Antar muka aplikasi kriptografi & steganografi



Gambar 4.2: Pemilihan file teks dengan fasilitas open file dialog

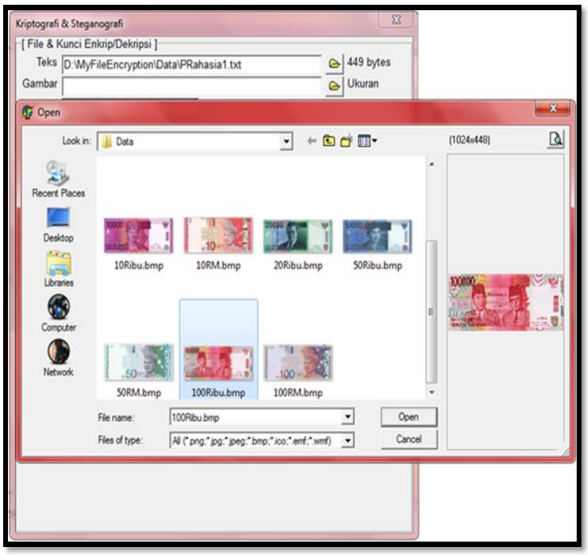


Gambar 4.3: Isi pesan rahasia dari file teks yang dipilih

Gambar 4.5: Tampilan gambar dari file citra digital yang dipilih

4.1.2 Pemilihan Berkas Citra Digital

Pemilihan *file* citra digital sebagai media penampung steganografi (*stego object*) dari teks pesan rahasia, dilakukan dengan cara menuliskan alamat dan nama *file* citra digital yang dimaksud pada bagian masukan (*input*) Gambar.



Gambar 4.4: Pemilihan file citra digital dengan fasilitas open file dialog

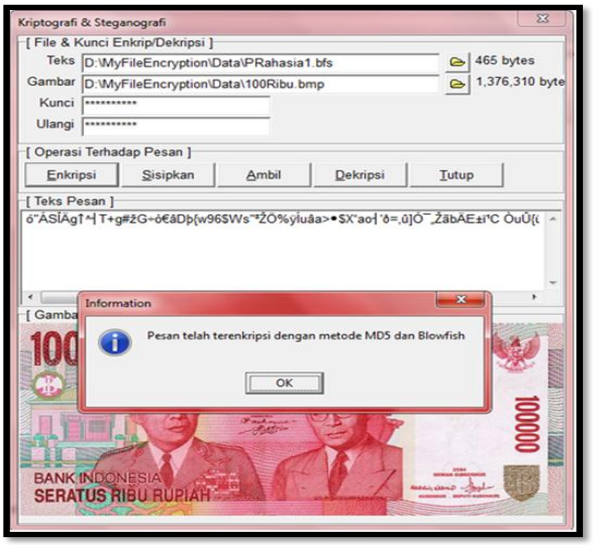
4.1.3 Pengisian Kunci Kriptografi

Pengisian kunci kriptografi dilakukan dengan cara mengisi sebuah teks sembarang pada bagian "Kunci". Selanjutnya ulangi pengisian teks kunci yang dimaksud pada bagian "Ulangi".

Gambar 4.6: Pengisian kunci kriptografi pesan rahasia

4.1.4 Proses Enkripsi Pesan Rahasia

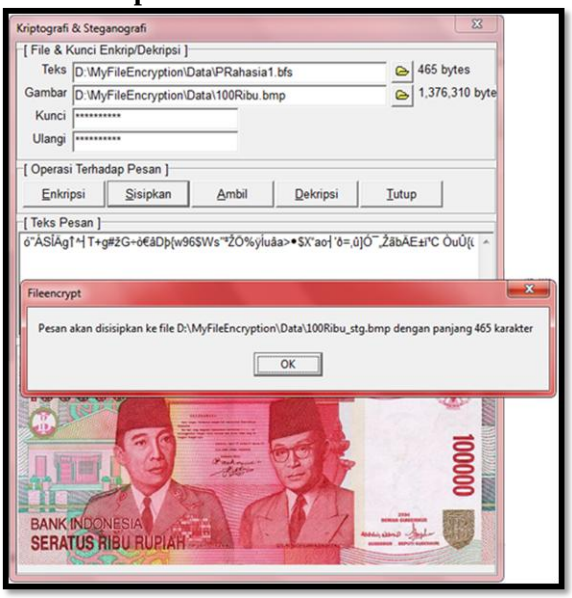
Proses selanjutnya adalah melakukan proses enkripsi teks pesan rahasia dengan cara menekan tombol **Enkripsi**. Proses enkripsi dilakukan dengan metode kriptografi MD5 dan Blowfish agar *plaintext* menjadi *ciphertext*.



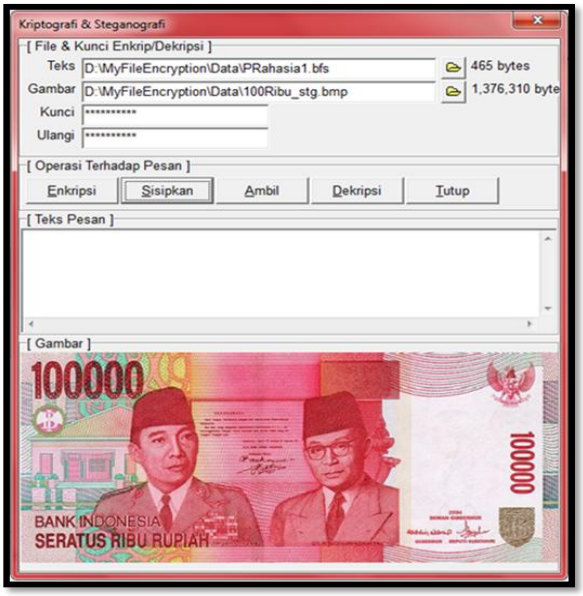
Gambar 4.7: Pengenkripsian pesan rahasia

4.1.5 Penyisipan Pesan Rahasia Ke Dalam *Stego Object*

Proses penyisipan pesan rahasia terenkripsi dengan teknik LSB (*Least Significant Bit*) ke *stego object* berupa citra digital *true color* yang telah dipilih, dengan menekan tombol **Sisipkan**.



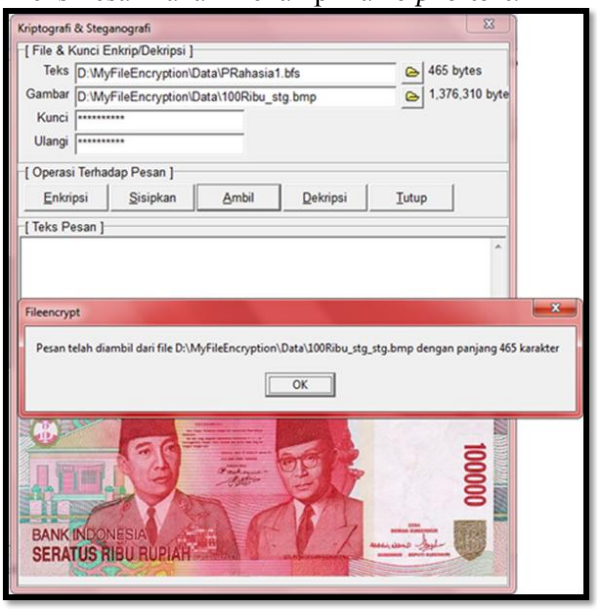
Gambar 4.8: Penyisipan pesan rahasia terenkripsi ke dalam *stego object*



Gambar 4.9: *Stego object* yang tersisipi pesan rahasia terenkripsi

4.1.6 Pengambilan *Ciphertext* Dari *Stego Object*

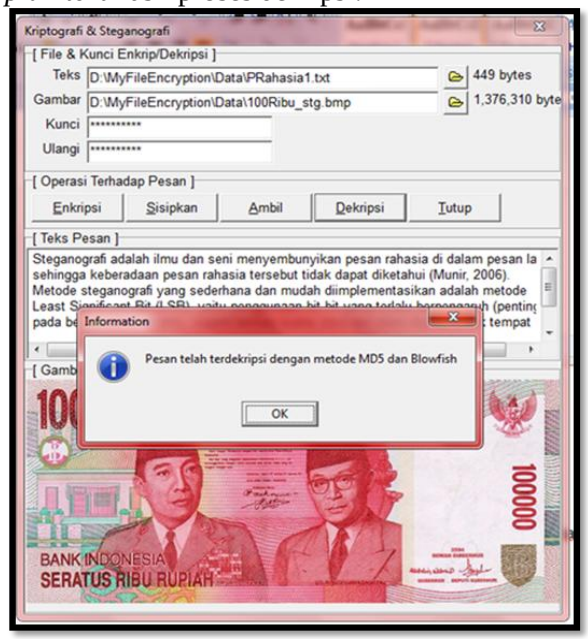
Pengambilan *ciphertext* dari dalam *stego object*, dilakukan dengan cara memilih file *stego object* yang berisi pesan rahasia terenkripsi pada bagian masukan **Gambar**. Selanjutnya menekan tombol **Ambil**. Jika berhasil, maka pada bagian "Teks Pesan" akan menampilkan *ciphertext*.



Gambar 4.10: Pengambilan *ciphertext* dari *stego object*

4.1.7 Proses Dekripsi *Ciphertext* Dari *Stego Object*

Proses dekripsi *ciphertext* untuk menjadi *plaintext*, dilakukan dengan cara memasukkan kunci dekripsi sesuai kunci yang digunakan pada saat enkripsi. Langakah selanjutnya adalah menekan tombol **Dekripsi**. Jika berhasil, maka pada bagian "Teks Pesan" akan menampilkan *plaintext* hasil proses dekripsi.



Gambar 4.11: Hasil proses dekripsi pesan rahasia

4.2 Hasil Pengujian

Hasil pengujian sistem untuk mengenkripsi *plaintext* dari kelima file pesan rahasia menjadi *ciphertext* dengan metode Blowfish dan *password* '**1234567890**', diperoleh hasil sebagai berikut;

Tabel 4.3: Hasil Proses Enkripsi *Plaintext*

Uji Ke	Nama File	*Plaintext*	*Ciphertext*	Kesimpulan
1	Prahasia1.txt	Steganografi adalah ilmu dan seni menyembunyikan pesan rahasia di dalam pesan lain sehingga keberadaan	Óê3§_tl{ ý^Ò__€7 ô±¹_·5ÒÀ É°□þ„‡3 Ø&"§î0·_ VE- ýèY"Ù_¢ _3¯å^Ïþ¿ TMK_\ò 1K<uN_,° ...	*Plaintext* pesan rahasia berhasil terenkripsi
2	Prahasia2.txt	4.1 Hasil Studi Literatur, Observasi dan Wawancara Kegiatan studi literature, observasi dan wawancara ...	°5¡TÚ _&¥_}2Y ¹ª7[<e~:?j _¬»%(D□ Ý¦~__qÑ· □ò€r_- @__– q¶¨ì_;%c Tž ...	*Plaintext* pesan rahasia berhasil terenkripsi
3	Prahasia3.txt	Kriptografi dalam sejarahnya tercatat dipergunakan secara terbatas oleh bangsa Mesir 4000 ..	·É@ŒB5 qº|&_6¯^ _ÐÍ_g…8 FßËdgCt; ¤– æ_MÐùM _tø___□– X- _l!Œ8É_ ®Ä□— W- O"ø$)·Û ...	*Plaintext* pesan rahasia berhasil terenkripsi
4	Prahasia4.txt	Untuk menjaga keamanan data ataupun informasi yang tersimpan dalam bentuk file, salah satu caranya ...	ï6ßqwÓÐ .žf2œDË MæR=(è• ïØ_S_ÏË" „_ Þ:WÜ,ã,ä X:_‹h[X @ðœ}RŸ Cj(k,6÷¬ °_2¹Œ— |TÀÌó H8»□¸ ...	*Plaintext* pesan rahasia berhasil terenkripsi
5	Prahasia5.txt	Cipher blok merupakan salah satu pendekatan dalam algoritma kriptografi kunci simetrik.	Ú__>Rw ƒ Ê__£î: Ãý□_a„i_ ÜEç2üï« Dá»(q£_† □…ÄSgë Q_Šµä_ ‰ŽV_×9 …3_T§Ú	*Plaintext* pesan rahasia berhasil terenkripsi

Uji Ke	Nama File	*Plaintext*	*Ciphertext*	Kesimpulan
		Pendekatan ...	_wOôS-ˆ3hI>¿þÁ ...	

Tabel 4.4: Hasil Proses Dekripsi *Ciphertext*

Uji Ke	Nama File	*Ciphertext*	*Plaintext*	Kesimpulan
1	Prahasia1.bfs	Óê3§_tl{ ý^Ò__€7 ô±¹_·5ÒÀ Éº□þ„‡3 Ø&“§î0·_ VE-ýèY"Ù_¢ _3¯å^Ïþ¿ TMK_\ò 1K<uN_,° ...	Steganografi adalah ilmu dan seni menyembunyikan pesan rahasia di dalam pesan lain sehingga keberadaan ...	*Ciphertext* berhasil didekripsi
2	Prahasia2.bfs	°5¡TÚ _&¥_}2Y ¹ª7[<e~:?j _¬»%(D□ Ý¦~__qÑ· □ò€r_- @__– q¶¨ì_;%c Tž ...	4.1 Hasil Studi Literatur, Observasi dan Wawancara Kegiatan studi literature, observasi dan wawancara ...	*Ciphertext* berhasil didekripsi
3	Prahasia3.bfs	·É@ŒB5 qº|&_6¯^ _ĐÍ_g…8 FßËdgCt; ¤– æ_MĐùM _tø___□– X-_l!Œ8É_ ®Ä□— W-O”ø$)·Û ...	Kriptografi dalam sejarahnya tercatat dipergunakan secara terbatas oleh bangsa Mesir 4000 ..	*Ciphertext* berhasil didekripsi
4	Prahasia4.bfs	ï6ßqwÓÐ .žf2œDË MæR=(è• ïØ_S_ÏË" „_ Þ:WÜ,ã,ä X:_‹h[X @ðœ}RŸ Cj(k,6÷¬ °_2¹Œ— |TÀÌó H8»□¸ ...	Untuk menjaga keamanan data ataupun informasi yang tersimpan dalam bentuk file, salah satu caranya ...	*Ciphertext* berhasil didekripsi
5	Prahasia5.bfs	Ú__>Rw *f* Ê__£î: Ãý□_a„i_ ÜEç2üï« Dá»(q£_† □…ÄSgë Q_Šµä_ ‰ŽV_×9 …3_T§Ú _wOôS-ˆ3hI>¿þÁ ...	Cipher blok merupakan salah satu pendekatan dalam algoritma kriptografi kunci simetrik. Pendekatan ...	*Ciphertext* berhasil didekripsi

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan seluruh pengujian yang telah dilakukan terhadap aplikasi perangkat lunak yang menerapkan teknik kriptografi pesan digital menggunakan metode kombinasi Blowfish dan MD5, serta penyisipan dan pengambilan pesan terenkripsi pada media citra true color 24 bit dengan teknik steganografi LSB (*Least Significant Bit*), diperoleh hasil sebagai berikut;

1. Proses enkripsi terhadap seluruh teks pesan rahasia dengan metode MD5 dan Blowfish, diperoleh hasil 100% sukses dilakukan untuk menghasilkan teks pesan rahasia terenkripsi (*ciphertext*).
2. Proses penyisipan dan pengambilan teks pesan rahasia terenkripsi (*ciphertext*) ke/dari dalam *stego object* berupa citra digital *true color*, 100% berhasil dilakukan dengan tampilan gambar yang mirip (tidak nampak perubahan yang signifikan) dengan

tampilan citra digital sebelum dilakukan penyisipan, sehingga keberadan teks pesan rahasia di dalam *stego object* tidak mudah disadari keberadaannya oleh pihak lain.

3. Proses dekripsi terhadap teks pesan rahasia terenkripsi yang diambil dari *stego object*, 100% berhasil dilakukan dengan metode MD5 dan Blowfish, untuk menghasilkan teks pesan rahasia asli (*plaintext*).

5.2 Saran

Pada penelitian selanjutnya, dapat dilakukan pengembangan dan implementasi pada metode-metode kriptografi lainnya dengan cara mengkombinasikan dua atau lebih teknik kriptografi untuk *ciphertext* yang lebih sulit dipecahkan oleh pihak yang tidak berhak. Teknik steganografi untuk menyembunyikan keberadaan pesan rahasia dapat juga diterapkan pada *stego object* lainnya (selain *object* citra digital *true color*) seperti media file suara dalam format mp3 dan wav, oleh karena itu perlu dilakukan penelitian lebih lanjut untuk mengetahui tingkat keberhasilannya.

DAFTAR PUSTAKA

Borg, Walter R. dan Meredith Damien Gall. 1979. *Educational Research, An Introduction*. third edition. New York: Longman.

C. Cachin, An Information-Theoretic Model for Steganography, *Proceedings of 2nd Workshop on Information Hiding*, MIT Laboratory for Computer Science, May 1998.

Jonathan Cummins, Patrick Diskin, Samuel Lau and Robert Parlett, Steganography And Digital Watermarking. 2004, School of Computer Science, The University of Birmingham.

Gonzales, Rafael C., Woods Richard E. 2002. *Digital Image Processing*.

Munir, Rinaldi. 2004. Kriptografi. (online), (http://ebookbrowse.com/algoritmarsa-pdf-d309993420, diakses tanggal 11 Maret 2013).

Munir, Rinaldi. 2006. Kriptografi. Bandung: Penerbit Informatika.

Menezes, A, VanOorschot, P, Vanstone, S. 1997. Handbook of Applied Cryptography. CRC Press, Inc.

Schneier, Bruce. 1996. Applied Cryptography. Wiley Publisher.