TEKNIK PENYEMBUNYIAN PESAN PDF TERENKRIPSI MENGGUNAKAN ALGORITMA TWOFISH DAN STEGANOGRAFI END OF FILE DALAM MEDIA GAMBAR

SKRIPSI

Diajukan Untuk Memenuhi Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Pada Fakultas Teknik
Universitas Islam Riau

DISUSUN OLEH :

BAYU PURNOMO AJI

143510500

PROGRAM STUDITEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS ISLAM RIAU
PEKANBARU
2021

LEMBAR PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini:

Nama	:	Bayu Purnomo Aji
Tempat/Tgl Lahir	:	Lirik, 24 November 1996
Alamat	:	Jl. Kuansing, Kartama, Marpoyan

Adalah mahasiswa Universitas Islam Riau yang terdaftar pada:

Fakultas	:	Teknik
Jurusan	:	Teknik Informatika
Program Studi	:	Teknik Informatika
Jenjang Pendidikan	:	Strata-1 (S1)

Dengan ini menyatakan dengan sesungguhnya bahwa skripsi yang saya tulis adalah benar dan asli hasil dari penelitian yang telah saya lakukan dengan judul **"TEKNIK PENYEMBUNYIAN PESAN PDF TERENKRIPSI MENGGUNAKAN ALGORITMA TWOFISH DAN STEGANOGRAFI END OF FILE DALAM MEDIA GAMBAR"**

Apabila dikemudian hari ada yang merasa dirugikan dan atau menuntut karena penelitian ini menggunakan sebagian hasil tulisan atau karya orang lain tanpa mencantumkan nama penulis yang bersangkutan, atau terbukti karya ilmiah ini **bukan** karya saya sendiri atau **plagiat** hasil karya orang lain, maka saya bersedia menerima sanksi sesuai dengan peraturan perundangan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sesungguhnya untuk dapat digunakan sebagaimana mestinya.

Pekanbaru, 12 Juli 2021

Yang membuat pernyataan,

Bayu Purnomo Aji

TEKNIK PENYEMBUNYIAN PESAN PDF TERENKRIPSI MENGGUNAKAN ALGORITMA TWOFISH DAN STEGANOGRAFI END OF FILE DALAM MEDIA GAMBAR

Bayu Purnomo Aji
Fakultas Teknik
Program Studi Teknik Informatika
Universitas Islam Riau
Email : Bayupurnomoaji@student.uir.ac.id

ABSTRAK

Pesan adalah sebuah pernyataan rahasia yang dibuat oleh seseorang dan ditujukan untuk orang lain yang dikehendaki. Dalam proses pengiriman pesan keamanan pesan menjadi faktor yang sangat penting. Salah satu cara pengamanan pesan dapat dilakukan dengan mengkombinasikan kriptografi dan steganografi. Tujuannya adalah untuk merahasiakan pesan yang dikirim yang dapat dilakukan dengan proses kriptografi, serta sekaligus menghindarkan pesan tersebut dari kecurigaan yang dapat dilakukan dengan proses steganografi. Pesan yang digunakan dalam penelitian ini berupa file pdf. Pada proses kriptografi, pesan yang berupa file tersebut akan dienkrip dengan metode Twofish, dan selanjutnya pesan yang telah terenkrip tersebut akan dilakukan proses steganografi dengan cara menyisipkan informasi atau pesan ke dalam sebuah file, dimana informasi atau pesan disisipkan di akhir file tersebut, dengan metode End of File (EOF). Hasil dari Penelitian ini adalah bahwa dengan menggunakan sebuh metode kriptografi Twofish dan Steganografi dalam penyisipan, maka pesan yang telah dienkrip sulit untuk dikembalikan ke pesan aslinya oleh pihak yang tidak berwenang. Hasil dari aplikasi ini adalah dapat menyisipkan pesan tersembunyi berupa file pdf ke dalam berkas citra digital berformat PNG/JPG dan dapat mengekstraksi kembali pesan tersembunyi tersebut dari dalam citra (stego-image).

Kata Kunci : Kriptografi, Steganografi, Twofish, End of File (EOF).

END OF FILES HIDDING TECHNIQUES USING TWOFISH ALGORITHM AND END OF FILE STEGANOGRAPHY IN IMAGE MEDIA

Bayu Purnomo Aji
Faculty of Engineering
Informatics Engineering
Islamic University of Riau
Email: Bayupurnomoaji@student.uir.ac.id

ABSTRACT

Message is a secret statement made by someone and is intended for the desired person. In the process of sending messages, message security is a very important factor. One way of securing messages can be done by combining cryptography and steganography. The aim is to keep the messages sent that can be done by means of a cryptographic process secret, and at the same time to avoid the messages from suspicion which can be done by the steganography process. The message used in this study is a pdf file. In the cryptography process, the message in the form of a file will be encrypted using the Twofish method, and then the encrypted message will be steganographically processed by inserting information or messages into a file, where the information or message is inserted at the end of the file, using the End method of File (EOF). The result of this research is that by using a Twofish cryptographic method and Steganography in the insertion, the encrypted message is difficult to be returned to the original message by an unauthorized party. The result of this application is that it can insert a hidden message in the form of a pdf file into a digital image file in PNG / JPG format and can extract the hidden message from the image (stego-image).

Keywords: Cryptography, Steganography, Twofish, End of File (EOF).

KATA PENGANTAR

Dengan mengucapkan puji syukur kepada Allah Subhanahu Wa Ta'ala yang telah melimpahkan segala rahmat dan hidayah-Nya kepada penulis, sehingga penulis dapat menyelesaikan tugas akhir yang berjudul "Teknik Penyembunyian Pesan PDF Terenkripsi Menggunakan Algoritma *Twofish* Dan Steganografi *End of File* Dalam Media Gambar ".

Pada kesempatan ini penulis menyampaikan penghargaan yang setinggi-tingginya kepada semua pihak yang telah memberikan kontribusinya sebelum dan selama pengerjaan tugas akhir ini.Atas semua bantuan, bimbingan, arahan, dukungan dan fasilitas yang telah diberikan, penulis mengucapkan terima kasih.

Pengerjaan tugas akhir ini dilakukan dengan semaksimal mungkin oleh penulis, tetapi penulis menyadari bahwa hasil yang diperoleh masih jauh dari sempurna.Oleh karena itu, saran dan kritik yang membangun sangat penulis harapkan untuk kesempurnaan tugas akhir ini.Besar harapan penulis agar tugas akhir ini dapat bermanfaat bagi pendidikan, khususnya di Fakultas Teknik Universitas Islam Riau.

Akhirnya segala hal yang benar dan terealisasi pada tulisan ini semata-mata karena Allah Subhanahu Wa Ta'ala. Segala kesalahan yang ada semuanya karena kekurangan dan keterbatasan penulis.

Pekanbaru, 30 Januari 2021

Bayu Purnomo Aji

DAFTAR ISI

DAFTAR TABEL

Hal

DAFTAR GAMBAR

Hal

DAFTAR LAMPIRAN

Hal

BAB I
PENDAHULUAN

1.1 Latar Belakang Masalah

Pesan adalah sebuah pernyataan rahasia yang dibuat oleh seseorang dan ditujukan untuk orang lain yang dikehendaki. Sangat pentingnya informasi dari sebuah pesan rahasia menyebabkan pesan tersebut tidak sampai kepada penerimanya, melainkan pesan tersebut jatuh ke tangan orang lain yang tidak dikehendaki. Dalam proses pengiriman pesan keamanan pesan menjadi faktor yang sangat penting. Untuk menjaga keamanan informasi pesan, maka salah satu solusinya yaitu dengan ilmu kriptografi.

Menurut Schneier Kriptografi merupakan salah satu cara untuk mengamankan data, yaitu dengan menyandikan pesan asli (*plaintext*) ke dalam bentuk pesan rahasia (*ciphertext*). Proses pengamanan ini melibatkan algoritma dan kunci. Kunci enkripsi dapat dengan mudah mengembalikan *plaintext*dari *ciphertext*. Oleh karena itu diperlukan algoritma kriptografi yang kuat. Dengan berkembangnya ilmu penyandian, orang dapat dengan mudah memperoleh kunci penyandian lewat berbagai macam cara. (Ratna Wati Simbolon, 2016).

Oleh karena itu pengembangan metode kriptografi perlu diperluas penggunaannya yang tidak hanya terbatas untuk penyandian berupa teks, tetapi juga berupa gambar (Siang, 2002), audio maupun video (Soplanit, 2005). Terdapat dua teknik yang digunakan dalam menyandikan data/gambar yaitu pertama kriptografi

klasik (Kriptografi Simetri) dan kedua kriptografi modern (Asimetri). Menurut Stinson Penyandian menggunakan kriptografi klasik adalah metode untuk mengubah data asli (*plainteks*) ke bentuk sandi (*chiperteks*) dengan menggunakan kunci yang sama. Sedangkan kriptografi modern menggunakan dua buah kunci. Satu kunci yang disebut kunci (public key) yang dapat dipublikasikan, sedang kunci lain yang disebut kunci privat (private key) harus dirahasiakan. (Emi Styaningsih, 2009).

Twofish merupakan algoritma kriptografi yang beroperasi dalam mode blok cipher berukuran 128 bit dengan ukuran kunci sebesar 256 bit, ukuran kunci yang besar ditujukan untuk meniadakan kemungkinan kunci lemah (weak-key).

Algoritma twofish menggunakan struktur sejenis Feistel dalam 16 putaran dengan tambahan teknik whitening terhadap input dan output. Teknik whitening sendiri adalah teknik melakukan operasi XOR terhadap materi kunci sebelum putaran pertama dan sesudah putaran akhir. Elemen di luar jaringan feistel normal yang terdapat dalam algoritma twofish adalah rotasi 1 bit. Proses rotasi ini dapat dipindahkan ke dalam fungsi F untuk membentuk struktur jaringan Feistel yang murni, tetapi hal ini membutuhkan tambahan rotasi kata sebelum langkah output whitening.

Twofish memanfaatkan teknik pemanipulasian bit , kotak permutasi / pemutihan, jaringan feistel, pemutaran ulang dan pergiliran kunci dengan jumlah perputaran dan pergiliran kunci sebanyak 16 kali , tranformasi pseudo-Hadamard , ekspansi dan filter , dan kotak MDS (Most Distance Separable).Hasil enkripsi dari pesan rahasia yaitu berupa pesan acak dengan menggunakan salah satu metode dari kriptografi yaitu

Twofish, akan menimbulkan kecurigaan karena pesan acak sama sekali tidak mempunyai makna secara kasat mata. Untuk mengatasi permasalah ini agar terhindar dari kecurigaan maka digunakan sebuah teknik penyembunyian pesan rahasia yaitu steganografi.

Menurut Cahyadi Steganografi merupakan suatu ilmu seni dalam menyembunyikan informasi dengan memasukkan informasi atau pesan tersebut ke dalam media lain. Sehingga keberadaan informasi tersebut tidak diketahui oleh orang lain. Media yang dapat dimanfaatkan untuk steganografi yaitu gambar, teks, video, dan audio. (Nurul Ftriani Andi Mu'mi, 2017)

Gambar adalah salah satu media dari steganografi yang digunakan untuk menyisipkan pesan. Media gambar ini yang paling banyak digunakan karena gambar mempunyai makna sehingga tidak mudah dicurigai. menyisipkan pesan kedalam gambar dapat menggunakan sebuah metode dalam steganografi, yaitu metode *End of File* (EOF).

Berdasarkan latar belakang tersebut maka penggunaan metode *Twofish* akan diimplemetasikan untuk meyandikan sebuah pesan pdf ke dalam suatu bentuk gambar dengan format .png, yakni format gambar yang umum digunakan. Gambar sebagai media yang disisipkan sudah merupakan gambar berwarna (*true color*). Dalam studi kasus ini, peneliti akan mengkaji lebih lanjut tentang Kriptografi dan Steganografi dengan judul "Teknik Penyembunyian Pesan PDF Terenkripsi Menggunakan Algoritma *Twofish* Dan Steganografi *End of File* (EOF) Dalam Media Gambar".

Kriptografi berfungsi untuk mengenkripsikan data atau pesan, sedangkan steganografi berfungsi untuk menyisipkan pesan kedalam sebuah gambar.

1.2 Identifikasi Masalah

Adapun identifikasi masalah yang dapat diambil dari latar belakang tersebut adalah sebagi berikut :

1. Kurangnya keamanan pada data/pesan pdf sehingga pihak lain dapat melihat isi pesan rahasia.
2. Belum adanya proses enkripsi dengan menggunakan sebuah algoritma *Twofish* dan Steganografi untuk keamanan data.
3. Pihak lain dapat mengambil atau mencuri data/pesan pdf karena tidak adanya kemanan pada pesan pdf tersebut.

1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka yang menjadi rumusan masalah dalam penelitian ini adalah :

1. Bagaimana membuat suatu aplikasi untuk keamanan pesan dengan mengkombinasikan Kriptografi *Twofish* dan Steganografi dengan *End of File* (EOF) untuk keamanan data pdf.
2. Bagaimana mengamankan dan merahasiakan data atau pesan yang dianggap tidak layak diketahui oleh orang yang tidak berhak.

1.4 Batasan Masalah

Adapun batasan masalah dari penelitian ini adalah :

1. Wadah yang digunakan untuk menyisipkan pesan adalah media dalam bentuk gambar dengan format JPG dan PNG.
2. Hasil file output disimpan dengan format JPG/PNG.
3. Pesan yang disisipkan hanya dalam bentuk file pdf.
4. Aplikasi ini dibangun berbasis web.

1.5 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

1. Mengamankan pesan dengan metode *Twofish* dan metode *End of File* (EOF).
2. Membantu pengguna untuk mengamankan pesan pdf yang akan dikirim dengan menyisipkan di dalam sebuah media berupa gambar.

1.6 Manfaat penelitian

Adapun manfaat dari penelitian ini yaitu diharapkan dapat membantu penggunanya untuk mengirim informasi atau pesan yang bersifat rahasia agar sampai ke tangan penerima tanpa menimbulkan kecurigaan pada pihak lain.

BAB II

LANDASAN TEORI

2.1 Studi Kepustakaan

Dalam penelitian ini, penulis mengambil beberapa referensi studi kepustakaan yang bersumber pada penelitian-penelitian sebelumnya. Hal ini berguna sebagai perbandingan bahan referensi dalam menyelesaikan penelitian ini.

Penelitian yang dilakukan oleh (Sandro Sembiring,2013) dengan judul "Perancangan Aplikasi Steganografi Untuk Menyisipkan Pesan Teks Pada Gambar Dengan Metode *End of File*". Permasalahan pada penelitian ini adalah adalah seseorang yang mengirimkan pesan tentunya menginginkan pesan tersebut hanya dapat dibaca oleh orang yang diinginkannya. Untuk menyelesaikan masalah tersebut dibutuhkan suatu sistem keamanan yang mampu menjaga kerahasiaan pesan yaitu menggunakan ilmu steganografi dengan teknik *End of File* (EOF).

Perbedaan nya adalah penelitian tersebut hanya menggunakan steganografi dengan teknik *End of File* (EOF) sedangkan penulis menggunakan kriptografi metode twofish dan steganografi *End of File* (EOF)

Penelitian lainnya yang juga menjadi acuan dalam penelitian ini adalah penelitian yang dilakukan oleh (Marsela Sutikno Dibiyo,Dkk 2013) dengan judul "Teknik Penyembunyian Pesan PDF Terenkripsi Menggunakan Algoritma Kriptografi Vernam Cipher Dan Steganografi *End of File*. Permasalahan pada penelitian ini adalah seseorang yang mengirim pesan tentunya menginginkan pesan

tersebut aman dan sampai kepada orang yang tepat. Untuk menyelesaikan masalah tersebut dibutuhkan suatu sistem keamanan yang mampu menjaga kerahasiaan pesan yaitu menggunakan ilmu kriptografi vernam cipher dan steganografi dengan teknik *End of File* (EOF).

Perbedaan nya adalah penelitian tersebut menggunakan vernam cipher sedangkan penulis menggunakan metode twofish, hanya saja pada kasus permasalahan nya hampir sedikit sama dengan penulis yaitu pada penyisipan pesan pdf ke dalam suatu bentuk gambar.

Berdasarkan penelitian terdahulu yang menjadi acuan, maka akan dilakukan penelitian kriptorafi yang menggunakan algoritma *Twofish* dan menggunakan steganografi *End of File* untuk proses enkripsi pada pesan berformat pdf dan akan disisipkan dimedia yang digunakan yaitu *file gambar* berformat *.jpg/png.

2.2 Dasar Teori

2.2.1 Kriptografi

Kriptografi berasal dari bahasa Yunani, yaitu *crypto* dan *graphia*. *Crypto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Menurut terminologinya, kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat lain. Kriptografi pada awalnya dijabarkan sebagai ilmu yang mempelajari bagaimana menyembunyikan pesan. Namun pada pengertian modern kriptografi adalah ilmu yang bersandarkan pada teknik matematika untuk berurusan dengan keamanan informasi seperti kerahasiaan, keutuhan data dan

otentikasi entitas. Jadi pengertian kriptografi modern adalah tidak saja berurusan hanya dengan penyembunyian pesan namun lebih pada sekumpulan teknik yang menyediakan keamanan informasi.

Berikut ini adalah beberapa sistem kriptografi yaitu, (Dony Ariyus, 2008):

1. ***Plainteks :*** Pesan atau data dalam bentuk aslinya yang dapat terbaca. Plainteks adalah masukan bagi algoritma enkripsi. Untuk selanjutnya digunakan istilah teks asli sebagai padanan kata *plainteks*.

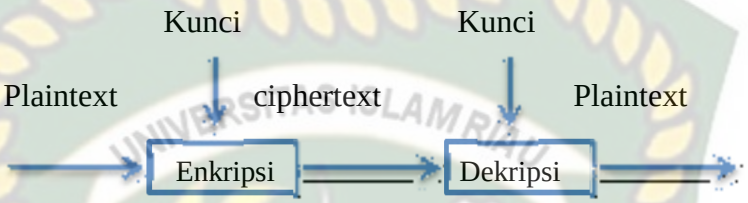
2. ***Secret Key :*** secret key yang juga masukan bagi algoritma enkripsi merupakan nilai yang bebas terhadap teks asli dan menentukan hasil keluaran algoritma enkripsi. Untuk selanjutnya digunakan istilah kunci rahasia sebagai padanan kata *secret key*.

3. ***Chiper Teks :*** *chiper teks* adalah keluaran algoritma enkripsi. *Chiper text* dapat dianggap sebagai pesan dalam bentuk tersembunyi. Algoritma enkripsi yang baik akan menghasilkan *chipertext* yang terlihat acak

4. **Algoritma Enkripsi :** Algoritma enkripsi memiliki 2 masukan teks asli dan kunci rahasia. Algoritma enkripsi melakukan transformasi terhadap teks asli sehingga menghasilkan teks sandi.

5. **Algoritma Dekripsi:** Algoritma dekripsi memiliki 2 masukan yaitu teks sandi dan kunci rahasia. Algoritma dekripsi memulihkan kembali teks sandi menjadi teks asli bila kunci rahasia yang dipakai algoritma dekripsi sama dengan kunci rahasia yang diapakai algoritma enkripsi.

Berikut ini adalah gambaran mekanisme atau cara kerja dalam kriptografi. Perhatikan pada Gambar 2.1



Gambar 2.1 Mekanisme Kriptografi

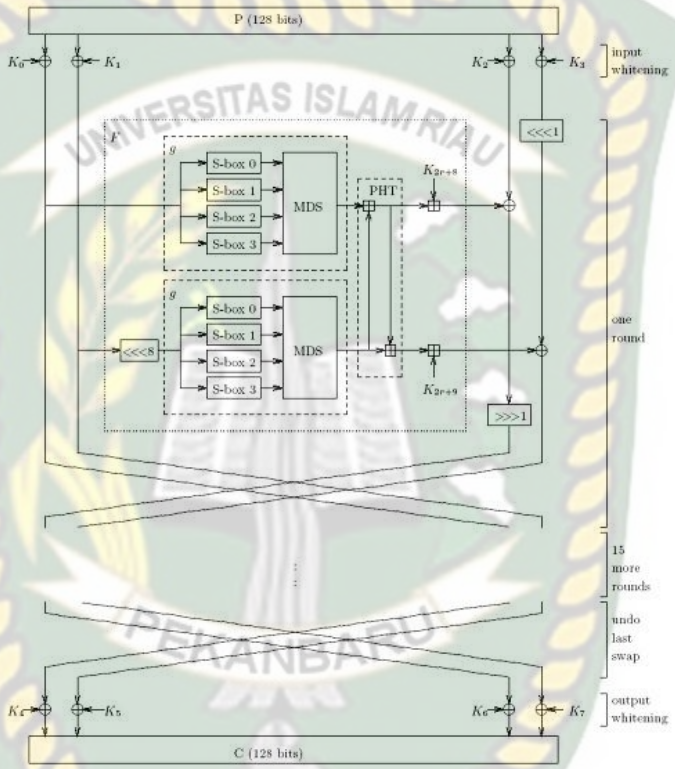
2.2.2 Twofish

Twofish merupakan algoritma yang beroperasi dalam mode block. Algoritma Twofish sendiri merupakan pengembangan dari algoritma Blowfish. Perancangan Twofish dilakukan dengan memperhatikan kriteria-kriteria yang diajukan National Institute of Standards and Technology (NIST) untuk kompetisi Advanced Encryption Standard (AES).

Algoritma Twofish diciptakan oleh Bruce Schneier, sebelumnya ia menciptakan algoritma blowfish dengan 64 bit block chipper dan kunci 128 bit. Twofish merupakan algoritma kunci simetris blok cipher dengan blok masukan 128 bit dan kunci 128 bit,192 bit dan 256 bit.

Pada implementasi algoritma Twofish, terdapat beberapa hal yang harus diperhatikan, antara lain :

1. Bit masukan sebanyak 128 bit akan dibagi menjadi empat bagian masing – masing 32 bit menggunakan konvensi little-Endian. Dua bagian bit akan menjadi bagian kanan, dan dua lainnya



Gambar 2.2 Struktur Algoritma Twofish

2. Bit input akan di-XOR terlebih dahulu dengan empat bagian kunci, atau dengan kata lain mengalami proses whitening.

R0,i = Pi ⊕_Ki i = 0, ..., 3 Dimana K adalah kunci, Ki berarti sub kunci yang ke-i.

3. Algoritma Twofish menggunakan struktur jaringan Feistel. Jaringan Feistel yang digunakan oleh Twofish terdiri atas 16 perulangan. Fungsi f pada algoritma Twofish terdiri atas beberapa tahap yaitu :

a. Fungsi g, yang terdiri dari 4 s-box dan matriks MDS

b. PHT (Pseudo-Hadamard Transformation) atau Transformasi Pseudo-Hadamard

c. Penambahan hasil PHT dengan kunci

Algoritma twofish memiliki beberapa blok pembangun, yaitu :

1. Jaringan Feistel, merupakan model komputasi berulang yang digunakan oleh banyak cipher blok. Fungsi dari model jaringan feistel adalah untuk memetakan suatu fungsi enkripsi yang sederhana menjadi fungsi enkripsi yang rumit dan kuat.
2. S-Box, merupakan matriks substitusi yang digunakan untuk memetakan bit-bit. S-box dapat bervariasi bentuk dan ukuran input output-nya. Twofish menggunakan empat S-box yang berbeda, dengan ukuran 8x8 bit.
3. Matriks MDS (Maximum Distance Separable) adalah matriks transformasi dari sebuah kode linear. Selain blok-blok pembangunan, algoritmactwofish terdiri dari beberapa proses, yaitu:

   1.Perubahan Pseudo-Hadamard, merupakan transformasi dua arah yang menghasilkan difusi. Difusi yang dimaksudkan disini adalah property dari operasi cipher yang dikatakan aman. Bit masukan dari Pseudo-Hadamard harus memiliki panjang yang genap, karena akan dibagi menjadi dua bagian yang sama panjang, masing-masing sepanjang n/2 yang dilambangkan dengan a dan b. Persamaan pseudo-hadamard adalah sebagai berikut :

a` = a + b (mod 2n) b` = a + 2b (mod 2n) untuk membalikkan persamaan di atas, persamaanya adalah : b = b` - a` (mod 2n) a = 2a` - b` (mod 2n) dimana n adalah jumlah bit yang digunakan.

2. Whitening, merupakan teknik untuk meningkatkan keamanan dari cipher blok yang menggunakan iterasi, tujuannya adalah agar input dan output dari fungsi F tidak diketahui. Whitening dilakukan dengan cara mengubah data dengan meng-XOR data dengan sebagian dari kunci sebelum iterasi pertama dan setelah iterasi terakhir dari enkripsi.
3. Penjadwalan kunci adalah proses mengubah kunci menjadi beberapa subkunci yang akan digunakan pada iterasi-iterasi.

Tahapan-tahapan pada algoritma twofish lebih jelasnya adalah sebagai berikut :

1. Bit masukan disebut sebagai P0, P1, P2, dan P3. P0 dan P1 akan menjadi bagian kiri, dua lainnya akan menjadi masukan pada bagian kanan.
2. Kemudian akan melalui proses whitening.
3. Bagian kiri akan menjadi masukan untuk fungsi f, P0 akan langsung menjadi masukan bagi fungsi g, sementara P1 akan di-rotate 8 bit sebelum diproses oleh fungsi g.
4. Didalam fungsi g, bit-bit tersebut akan melalui Sbox dan matriks MDS kemudian kedua keluaran akan digabungkan oleh PHT.
5. Setelah melalui PHT, kedua bagian tersebut akan ditambah dengan bagian dari kunci sesuai dengan iterasi yang telah dilewati. Untuk

keluaran dari fungsi f dengan input P1 akan ditambah dengan K2r+8. Untuk keluaran dari fungsi dengan input P1 akan ditambah dengan K2r+9, dimana r adalah jumlah iterasi yang telah dilewati. Masing-masing ditambah delapan dan sembilan karena delapan urutan awal sudah digunakan untuk whitening input dan output.

6. Keluaran dari fungsi f dengan input P0 akan diXOR dengan P2, kemudian hasil XOR tersebut akan di-rotate 1 bit.
7. Keluaran dari fungsi f dengan input P1 akan diXOR dengan P3, namun P sebelumnya di-rotate 1 bit terlebih dahulu.
8. Setelah perhitungan bit selesai, bagian kanan yangtelah dihitung tadi akan menjadi bagian kiri dan bagian kiri yang belum dihitung akan menjadi bagian kanan.
9. Kemudian setelah 16 iterasi, akan dilakukan whitening terhadap keluarannya. Whitening pada output akan meng-undo pertukaran bagian kanan dan bagian kiri pada iterasi terakhir, dan melakukan XOR data dengan 4 bagian kunci, Ci = R16,(i+2)mod 4 Ki+4 i = 0, ..., 3 Bagian kunci yang digunakan disini berbeda dengan bagian kunci yang akan digunakan saat whitening pada input. Oleh karena itu urutan bagian kunci yang dipakai ditambah empat, karena empat urutan bagian kunci satu sampai empat sudah terlebih dahulu digunakan untuk whitening pada input.

10. Keempat bagian cipherteks tersebut kemudian ditulis menjadi 16 byte C0, ..., C15 menggunakan konversi little-endian seperti pada plainteks. Ci = mod 28 i = 0, ..., 15 Implementasi algoritma twofish harus memperhatikan kecepatan komputasi yang diinginkan.

2.2.3 Steganografi

Steganografi adalah ilmu dan seni untuk menyembunyikan pesan rahasia (*hidding message*) sedemikian sehingga keberadaan (*eksistensi*) pesan tidak terdeteksi oleh indera manusia. Media yang digunakan umumnya merupakan suatu media yang berbeda dengan media pembawa informasi rahasia, disinilah fungsi dari teknik *steganography* yaitu sebagai teknik penyamaran menggunakan media lain yang berbeda sehingga informasi rahasia dalam media awal tidak terlihat secara jelas (Munir,2004).

Menurut Alatas (2009), bahwa ada beberapa contoh media penyisipan pesan rahasia yang digunakan dalam teknik *steganography* antara lain:

1. Teks

   Dalam algoritma *steganography* yang menggunakan teks sebagai media penyisipan biasanya digunakan teknik NLP (*Natural Language Processing*) sehingga teks yang telah disisipi pesan rahasia tidak akan mencurigakan untuk orang yang melihatnya.

2. Gambar

   Format gambar paling sering digunakan, karena format ini merupakan salah satu format *file* yang sering dipertukarkan dalam dunia internet.

Alasan lainnya adalah banyaknya tersedia algoritma *Steganography* untuk media penampung yang berupa citra.

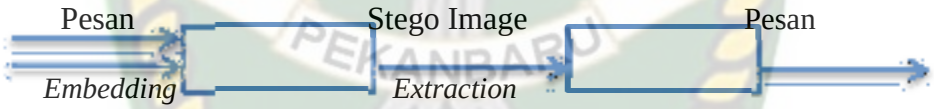
3. Audio

   Format ini pun sering dipilih karena biasanya berkas dengan format ini berukuran relatif besar, sehingga dapat menampung pesan rahasia dalam jumlah yang besar pula Format suara sering dipilih karena biasanya berkas dengan format ini berukuran relatif besar. Sehingga dapat menampung pesan rahasia dalam jumlah yang besar.

5. Video

   Format ini memang merupakan format dengan ukuran *file* yang relatif sangat besar namun jarang digunakan karena ukurannya yang terlalu besar sehingga mengurangi kepraktisannya dan juga kurangnya algoritma yang mendukung format ini.

Proses steganografi secara umum dengan media citra dapat dilihat pada Gambar 2.3 yaitu sebagai berikut.



Gambar 2.3 Proses *Embedding* dan Ekstraksi

Berdasarkan Gambar 2.3 proses steganografi untuk menyisipkan data yang ingin disembunyikan membutuhkan dua unsur yaitu media penampung seperti citra dan pesan yang ingin disembunyikan. Langkah pertama dengan menginputkan pesan dan *cover image* (citra yang digunakan sebagai media penyisipan). Kemudian dilakukan proses *embedding* untuk menyisipkan pesan ke dalam *cover image* maka diperoleh hasilnya berupa *stego image.* kemudian *Stego image* ini yang akan dikirim ke penerima pesan. Selanjutnya Penerima pesan melakukan proses *extraction* untuk

ekstraksi pesan dari *stego image*. Setelah melakukan proses *extraction*, maka pesan yang telah dikirim dapat dibaca.

Sebuah steganografi memiliki tiga aspek yang dapat menentukan berhasil tidaknya sebuah steganografi dalam melakukan pekerjaannya (Emardi dkk, 2004):

1. Kapasitas (*capacity*)

Kapasitas merujuk pada jumlah informasi yang bisa disembunyikan dalam medium cover. Keamanan adalah ketidakmampuan pengamat untuk mendeteksi pesan tersembunyi dan ketahanan dalam jumlah modifikasi medium stego yang bisa bertahan sebelum musuh merusak pesan rahasia tersembunyi tersebut.

2. Keamanan (*security*)

Keamanan dari sistem steganografi klasik mewujudkan kerahasiaan sistem *encodingnya.*

3. Ketahanan (*robustness*)

Ketahanan mangacu pada data citra penampung seperti pengubah kontras, penajaman, rotasi, perbesar gambar, pemotongan dan lain-lainnya.Bila pada citra dilakukan operasi pengolahan citra, maka data yang disembunyikan tidak rusak.

2.2.4 *End of File*

Metode *End of File* (EOF) merupakan salah satu metode yang digunakan dalam steganografi. Metode ini menyembunyikan pesan rahasia dengan cara menambahkan bit-bit pesan yang akan disembunyikan ke akhir file citra penampung. Teknik ini menggunakan cara dengan menyisipkan data pada akhir file. Sehingga, tidak akan mengganggu kualitas data awal yang akan disisipkan pesan. Namun, ukuran file

setelah disisipi pesan rahasia akan bertambah. Sebab, ukuran file yang telah disisipkan pesan rahasia sama dengan ukuran file sebelum disisipkan pesan rahasia ditambah dengan ukuran pesan rahasia yang disisipkan. Untuk mengenal data yang disisipkan pada akhir file, diperlukan suatu tanda pengenal atau simbol pada awal dan akhir data yang akan disisipkan. Proses penyisipan pesan dengan metode EOF dapat dituliskan dalam algoritma sebagai berikut :

1. Inputkan cipherfile yang akan disisipkan.
2. Inputkan citra yang akan menjadi media penyisipan cipherfile.
3. Baca nilai setiap pixel citra.
4. Tambahkan cipherteks sebagai nilai akhir pixel citra dengan diberi karakter penanda sebagai penanda akhir cipherfile.
5. Petakan menjadi citra baru.

Proses pengambilan cipherteks dari media menggunakan metode *End of File* adalah sebagai berikut :

1. Inputkan citra yang telah disisipkan cipherfile (stego image).
2. Baca nilai pixel stego image yang terdapat pada baris terakhir matriks pixel citra.
3. Ambil cipherfile yang terdapat pada stego image, yaitu nilai pixel awal yang terdapat pada baris terakhir matriks pixel citra sampai nilai desimal karakter penanda.

Berikut ini merupakan contoh penyisipan pesan menggunakan metode End of File terdapat suatu citra warna RGB 7x7 yang memiliki nilai setiap pixel seperti pada Tabel 2.1.

Tabel 2.1 Matriks *Pixel* Citra RGB

R=77 G=82 B=43	R=65 G=65 B=89	R=167 G=168 B=70	R=171 G=147 B=153	R=160 G=130 B=141	R=60 G=30 B=41	R=60 G=70 B=141
R=92 G=55 B=35	R=76 G=76 B=90	R=64 G=57 B=158	R=181 G=70 B=179	R=204 G=203 B=199	R=24 G=93 B=109	R=204 G=95 B=99
R=76 G=90 B=82	R=69 G=73 B=83	R=163 G=63 B=171	R=186 G=190 B=189	R=177 G=184 B=162	R=17 G=101 B=56	R=77 G=84 B=162
R=95 G=102 B=112	R=86 G=98 B=114	R=157 G=172 B=182	R=167 G=182 B=184	R=174 G=186 B=187	R=74 G=86 B=18	R=174 G=186 B=187
R=74 G=84 B=99	R=121 G=14 B=152	R=173 G=87 B=194	R=184 G=96 B=21	R=202 G=14 B=71	R=22 G=46 B=73	R=100 G=214 B=227
R=74 G=84 B=99	R=21 G=134 B=100	R=173 G=96 B=94	R=184 G=196 B=107	R=22 G=46 B=79	R=104 G=77 B=99	R=202 G=89 B=99
R=74 G=94 B=93	R=121 G=134 B=92	R=173 G=90 B=194	R=84 G=96 B=79	R=92 G=100 B=81	R=52 G=76 B=91	R=202 G=214 B=227

Citra RGB tersebut akan disisipkan pesan “saya arfiyah” yang memiliki nilai decimal dalam kode ASCII yaitu “115 97 121 97 32 97 114 102 105 121 97 104”. Pesan akan ditambahkan sebagai nilai akhir secara vertikal pada pixel citra RGB. Pada awal dan akhir pesan diberi karakter penanda “zx” yang memiliki nilai desimal dalam kode ASCII yaitu “122 120”. Maka didapatlah matriks pixel seperti pada Tabel 2.2.

Tabel 2.2 Matriks Pixel Citra RGB yang Telah Disisipkan Pesan dengan metode *End of File* (EOF)

R=77 G=82 B=43	R=65 G=65 B=89	R=167 G=168 B=70	R=171 G=147 B=153	R=160 G=130 B=141	R=60 G=30 B=41	R=60 G=70 B=141	**R=122** **G=120** **B=115**
R=92 G=55 B=35	R=76 G=76 B=90	R=64 G=57 B=158	R=181 G=70 B=179	R=204 G=203 B=199	R=24 G=93 B=109	R=204 G=95 B=99	**R=97** **G=121** **B=97**
R=76 G=90 B=82	R=69 G=73 B=83	R=163 G=63 B=171	R=186 G=190 B=189	R=177 G=184 B=162	R=17 G=101 B=56	R=77 G=84 B=162	**R=32** **G=97** **B=114**
R=95 G=102 B=112	R=86 G=98 B=114	R=157 G=172 B=182	R=167 G=182 B=184	R=174 G=186 B=187	R=74 G=86 B=18	R=174 G=186 B=187	**R=102** **G=105** **B=121**
R=74 G=84 B=99	R=121 G=14 B=152	R=173 G=87 B=194	R=184 G=96 B=21	R=202 G=14 B=71	R=22 G=46 B=73	R=100 G=214 B=227	**R=97** **G=104** **B=122**
R=74 G=84	R=21 G=134	R=173 G=96	R=184 G=196	R=22 G=46	R=104 G=77	R=202 G=89	**R=120** **G=0**

B=99	B=100	B=94	B=107	B=79	B=99	B=99	**B=0**
R=74 G=94 B=93	R=121 G=134 B=92	R=173 G=90 B=194	R=84 G=96 B=79	R=92 G=100 B=81	R=52 G=76 B=91	R=202 G=214 B=227	**R=0** **G=0** **B=0**

2.2.5 Citra Digital

Citra (*Image)* adalah sebuah istilah lain untuk gambar, sebagai salah satu komponen multimedia yang memegang peranan penting sebagai bentuk informasi visual. Citra mempunyai karateristik yang tidak dimiliki oleh data teks, yaitu citra kaya dengan informasi. Secara harfiah citra *(Image)* adalah gambar pada bidang dwimatra *(Dua dimensi)* ditinjau dari sudut pandang matematis, citra merupakan fungsi menerus*(Continue)* dari intensitas cahaya pada dwimatra. Sumber cahaya menerangi objek dan objek memantulkan kembali sebagaian dari berkas cahaya pantulan cahaya tersebut ditangkap oleh alat - alat optik, misalnya mata pada manusia, kamera, pemindai (*Scanner*) sehingga objek yang disebut citra tersebut terekam (Munir, 2004).

2.2.6 PHP

Pengertian PHP menurut (Anhar, 2010:3), PHP merupakan singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada *server* (*server side HTML embedded scripting*). PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client.* Mekanisme ini menyebabkan

informasi yang diterima *client* selalu yang terbaru/*up to date*. Semua *script* PHP dieksekusi pada *server* di mana *script* tersebut dijalankan.

2.2.7 *Unified Modelling Language* (UML)

Unified Modeling Language (UML) adalah sebuah bahasa pemrograman yang telah menjadi standard untuk merancang dan mendokumentasikan sistem perangkat lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem dan sudah digunakan secara luas dan menggunakan notasi yang sudah dikenal untuk analisa dan desain berorientasi objek. Sesuai dengan pengertian dan konsep UML, maka ada beberapa diagram yang dapat digunakan untuk memperjelas penggunaan UML dalam pemrograman berorientasi objek, diantaranya: *Use Case Diagram. Class Diagram*, *Component Diagram* dan *Physical Diagram*.

2.2.8 *Use Case Diagram*

Use Case Diagram menggambarkan sebuah fungsi yang dibutuhkan oleh sebuah sistem. Dalam hal ini ada kondisi yang agak beda, yaitu disini sistem dituntut untuk berbuat. Sebuah use case mempresentasikan sebuah interaksi antara pengguna dengan sebuah sistem. *Use Case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, membuat sebuah daftar aktivitas dan sebagainya

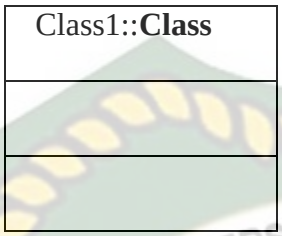
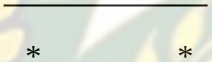
Tabel 2.3 Simbol pada *Use Case*

No	Nama	*Symbols*	Keterangan
1	*Use case*		Abstraksi dari interaksi antara *system* dan *actor*.
2	*Actor*		Mewakili peran orang , system yang lain atau alat ketika berkomunikasi dengan *use case*
3	*Relationship*		Penghubung antara objek satu dengan yang lain.

2.2.9 Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). Class diagram memberikan pandangan secara luas dari seuatu sistem denganmenunjukkan kelas-kelasnya dan hubungan mereka. Diagram Class bersifat statis, menggambarkan hubungan apa yang terjadi bukan apa yang terjadi jika mereka berhubungan. Dalam *class diagram* terdapat beberapa simbol, beberapa simbol tersebut dapat dilihat pada tabel 2.4 dibawah ini:

Tabel 2.4 Simbol pada *Class Diagram*

No	SIMBOL	PENJELASAN
1		*Class*, digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari *class*. Bagian tengah mendefinisikan property/atribut *class*. Bagian akhir mendefinisikan *method-method* dari sebuah *class*.
2		*Assosiation,* digunakan sebagai relasi antar dua kelas atau lebih.
3		*Composition,* jika sebuah *class* tidak bisa berdiri sendiri dan harus merupakan bagian dari *class* yang lain, maka *class* tersebut memiliki relasi *composition* terhadap *class* tempat dia bergantung tersebut. Sebuah *relationship composition* digambarkan sebagai garis dengan ujung berbentuk jajaran genjang berisi/solid.
4		*Dependency*, digunakan untuk menunjukan operasi pada suatu *class* yang menggunakan *class* yang lain. Sebuah *dependency* dilambangkan sebagai sebuah panah bertitik-titik.

2.2.10 *Activity Diagram*

Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses parelel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan *state diagram* khusus, dimana sebagian besar *sate* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). *Activity diagram* dapat digunakan untuk menjelaskan bisnis dan alur kerja operasional secara tahap demi tahap dari komponen suatu sistem. *Activity diagram* menunjukkan keseluruhan dari aliran *control*. Berikut ini ada beberapa simbol yang terdapat pada *activity diagram.*

perhatikan pada Tabel 2.5 dibawah ini:

Tabel 2.5 Simbol pada *Activity Diagram*

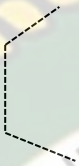
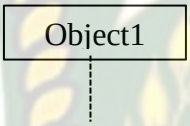
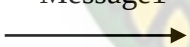
No	SIMBOL	PENJELASAN
1		*Activity*, memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		*Action*, state dari sistem yang mencerminkan eksekusi dari suatu aksi.
3		*Initial state*, bagaimana objek di bentuk atau diawali.
4		*Final state*, bagaimana objek dibentuk dan diakhiri .
5		*Decision*, digunakan untuk menggambarkan suatu keputusan atau tindakan yang harus diambil pada kondisi tertentu.
6		*Control Flow*, menunjukan bagaimana kendali suatu aktivitas terjadi pada aliran kerja dalam tindakan tertentu.

2.2.11 *Sequence Diagram*

Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence* diagram adalah gambaran tahap demi

tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case* diagram.

Tabel 2.6 Simbol *Sequence Diagram*

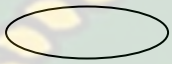
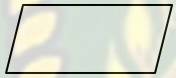
No	SIMBOL	PENJELASAN
1		*Lifeline* mengindikasikan keberadaan sebuah *object* dalam basis waktu. Notasi untuk *Lifeline* adalah garis putus-putus *vertikal* yang ditarik dari sebuah *object.*
2		*Object* merupakan *instance* dari sebuah *class* digambarkan sebagai sebuah *class* (kotak) dengan nama *object* didalamnya yang diawali dengan sebuah titik koma.
3		*Actor* juga dapat berkomunikasi dengan *object*, maka *actor* juga dapat diurutkan sebagai kolom. Simbol *actor* sama dengan simbol pada *Actor Case Diagram*.
4		*Activation* dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah *lifeline*. Mengindikasikan sebuah objek yang akan melakukan sebuah aksi.
5		*Message*, digambarkan dengan anak panah horizontal antara *Activation Message* mengindikaskan komunikasi antara objek-objek.

2.2.12 Diagran Alir (Flowchart)

Penggunaan diagram alir ini adalah untuk menggambarkan alur logika dari sebuah program. Penggambaran alur logika digambarkan secara grafis menggunakan *flowchart*. Urutan-urutan proses yang sangat rumit yang tidak bias dibuat dengan

pseudocode akan mampu digambarkan oleh diagram alir ini. Simbol-simbol yang digunakan dalam diagram alir dapat dilihat pada tabel 2.7 berikut.

Tabel 2.7 Simbol *Flowchart*

No	Simbol	Nama	Fungsi
1		Memulai/Selesai	Memulai proses
2		Proses	Menyatakan operasi yang dilakukan oleh sebuah sistem.
3		Input / Output	Menunjukan data masukkan atau keluaran.
4		Kondisi	Menentukan kondisi yang diambil oleh sistem.
5		Dokumen	Menyatakan cetak
6		Penghubung	Menyatakan titik temu aliran diagram alur
7		Tanda Prosedur	Menyatakan prosedur algoritma

2.3 Hipotesis

Kesimpulan sementara dari penelitian ini adalah:

1.Aplikasi ini dapat membantu dan memudahkan pengguna dalam menjaga keamanan data/pesan pdf.

2.Dapat memberikan tingkat keamanan data/pesan pdf dengan jauh lebih baik karena menggunakan twofish dan steganografi EOF.

BAB III
METODOLOGI PENELITIAN

3.1 Alat dan Bahan Penelitian Yang Digunakan

Alat dan bahan penelitian ini adalah sebuah pendukung baik perangkat keras dan perangkat lunak sehingga penelitian ini sesuai dengan tujuan dan manfaat. Berikut ini adalah alat dan bahan penelitian digunakan penulis untuk menganalisa dan merancang sistem.

3.1.1 Spesifikasi Perangkat Keras (*Hardware*)

Spesifikasi Perangkat Keras (*hardware*) pada laptop yang digunakan dalam melakukan penelitian ini adalah sebagai berikut :

1. Laptop Intel Inside Core i3
2. RAM 2 GB
3. *Hardisk* 500 GB

3.1.2 Spesifikasi Perangkat Lunak (*Software*)

1. *Microsoft Windows* 7 *Ultimate*.
2. PHP.
3. CSS.
4. *Java Script*.
5. *Microsoft Visio 2007.*
6. *Sublime Text* 3.

7. XAMPP V3.2.1.

8. Web *browser* : *Mozilla Firefox* dan *Google Chrome*.

3.1.3 Metode Penelitian

Metode penelitian adalah ilmu yang mempelajari cara-cara melakukan pengamatan dengan pemikiran yang tepat secara terpadu melalui tahapan-tahapan yang disusun secara ilmiah mencari, menyusun serta menganalisis dan menyimpulkan data-data, sehingga dapat dipergunakan untuk menemukan, mengembangkan dan menguji kebenaran sesuatu pengetahuan berdasarkan bimbingan Tuhan. Metode analisa dalam skripsi ini adalah metode eksperimen, adapun tahap-tahapan dalam penulisan ini adalah :

1. Pengumpulan data

Data yang digunakan pada penelitian ini bersumber dari berbagai informasi yang membahas tentang permasalahan yang penulis teliti yaitu, pertama studi kepustakaan adalah mengumpulkan data dengan cara mencari dan mempelajari dari berbagai sumber yang berkaitan dengan masalah yang diteliti, baik dari internet, buku, jurnal ilmiah dan bacaan lainnya yang dapat dipertanggung jawabkan. Kedua kuesioner adalah mengumpulkan data dengan cara memberikan kuesioner dengan beberapa pertanyaan kepada masyarakat dan mahasiswa.

2. Analisis Masalah

Setelah pengumpulan data dan membaca literatur penelitian sebelumnya maka penulis mendapatkan masalah. Dari masalah tersebut maka penulis menjadikan nya sebuah penelitian yang baru dan berbeda.

3. Pengembangan dan Perancangan sistem.

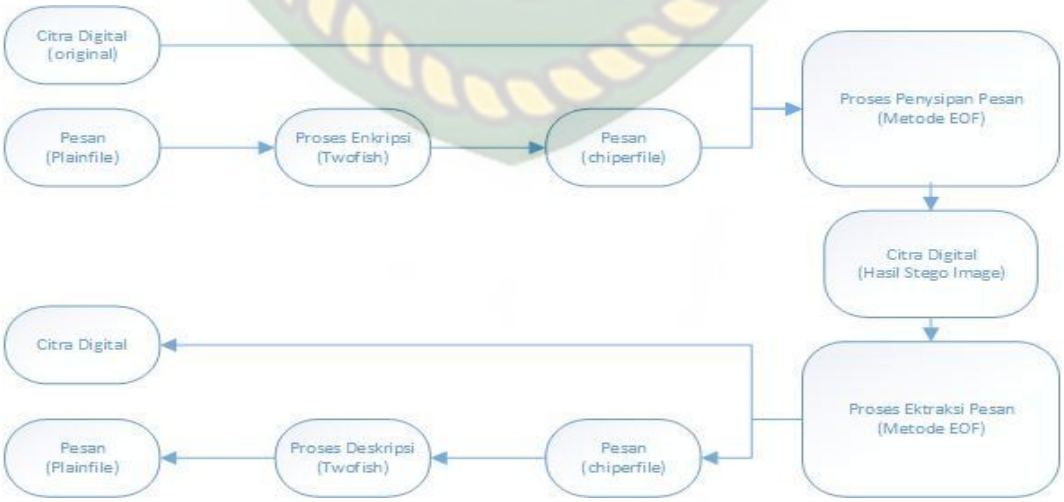
Pengembangan yang dilakukan penulis adalah dengan mengkombinasikan Kriptografi *Twofish* dan Steganografi dengan EOF yang digunakan dalam penelitian ini.

4. Pengujian

Pengujian yang akan digunakan adalah pengujian White Box atau Black Box.

3.2 Sistem Yang Akan Dibangun

Pada perancangan aplikasi kombinasi kriptografi *Twofish* dan steganografi dengan EOF ini terdiri dari dua proses utama yang dilakukan oleh pengirim pesan yaitu, proses Embedding pesan dan Enkripsi pesan *Twofish* dan steganografi dengan EOF, dan dua proses utama yang dilakukan oleh penerima pesan yaitu, Extraction pesan dan Dekripsi pesan dengan menggunakan metode *Twofish* dan steganografi dengan EOF. Alur sistem yang akan dibangun dapat dilihat pada gambar 3.1.



Gambar 3.1 Alur Sistem yang akan dibangun

3.3 Analisa Prosedural

Analisa prosedural merupakan analisa implementasi pada pemrograman sistem.Sistem yang dimaksud disini adalah analisa sistem pemograman enkripsi dengan menggunakan algoritma twofish dan steganografi *End of File*.

Pada proses enkripsi pesan terdapat tugas-tugas seperti berikut :

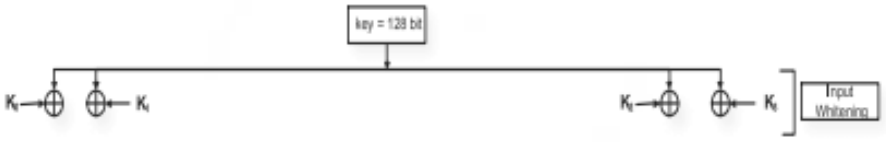
1. Load file yang akan dienkripsi. Format file yang digunakan adalah .pdf. Sedangkan untuk ukuran filenya tidak ada batasannya, semakin besar ukuran filenya maka semakin banyak pula pesan yang dapat dienkripsi.
2. Input plainteks sebesar 128 bit akan dibagi menjadi empat word yaitu P0, P1, P2, P3 yang masing-masing sebesar 32 bit. P0 dan P1 akan menjadi bagian kiri, sedangkan P2 dan P3 akan menjadi bagian kanan. Plainteks tersebut dipecah menjadi empat kata sebanyak 32-bit dengan menggunakan konversi little endian.

$$\text{Pi}=\sum_{j}^{3}=\text{P}(4_i+j)2^{8j} \tag{3.1}$$

   dimana i=0,…,3.
3. Plainteks akan melalui proses input whitening yaitu input akan di-XOR dengan empat word kunci yang telah terjadwal yaitu K0, K1, K2, dan K3. Secara formalnya adalah sebagai berikut :

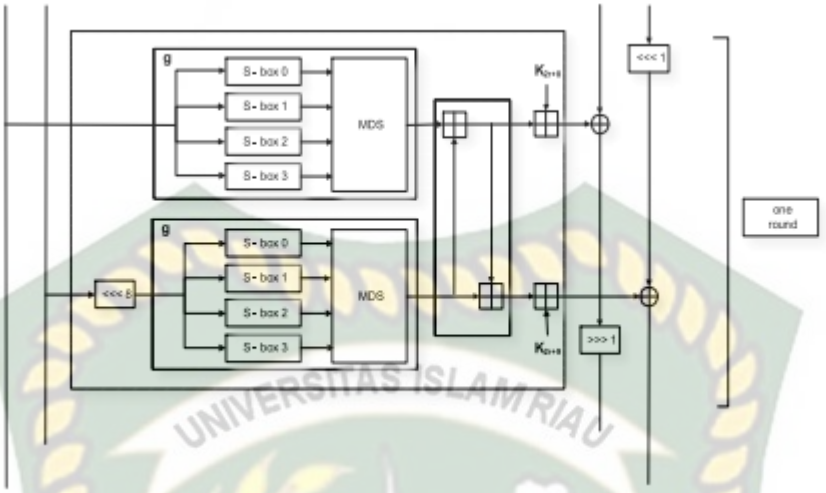
$$\text{Ro,i} = \text{Pi xor Ki} \tag{3.2}$$

   dimana i=0,…,3



Gambar 3.2 Proses whitening

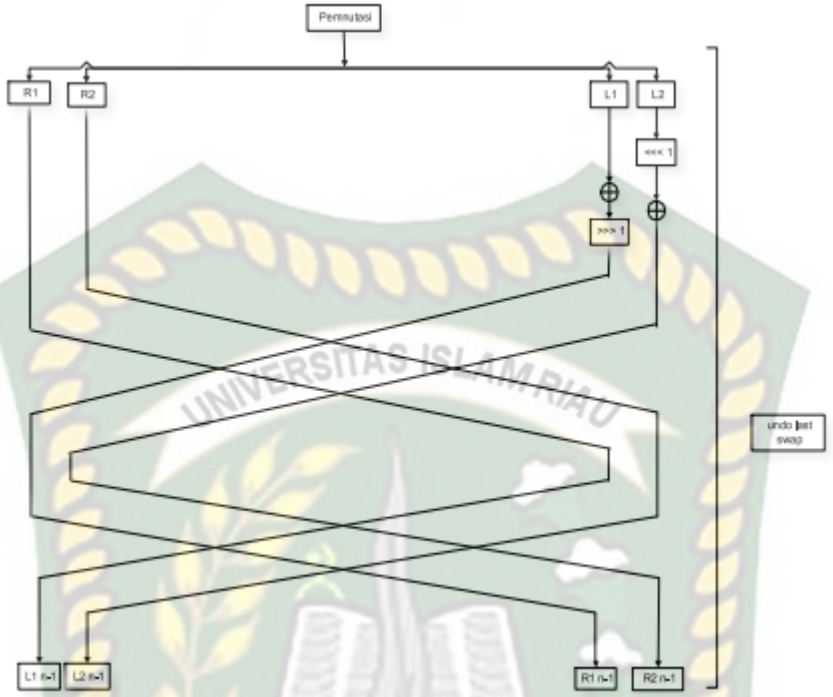
4. Proses berikutnya input akan melalui proses pada fungsi F yang meliputi didalamnya adalah fungsi g dan dilanjutkan dengan PHT (pseudo hadamard transform), dan dilakukan penambahan hasil PHT dengan kunci. Proses fungsi F tersebut dilakukan secara bertahap. R0 dan R1 yang merupakan hasil whitening akan menjadi input untuk fungsi F.
5. R0 dan R1 akan dimasukkan ke dalam fungsi g yang merupakan bagian awal dari fungsi F. Untuk R1 sebelum dimasukkan ke dalam fungsi g akan dirotasi ke kiri sejauh 8 bit. R0 dan R1 melalui S-box dan selanjutnya akan dikalikan dengan matriks MDS. Hasil dari fungsi g ini masing-masing menjadi T0 dan T1.
6. T0 dan T1 akan melalui proses PHT yang merupakan penggabungan T0 dan T1 dimana T0 + T1 dan T0 + 2T1. Setelah itu hasil dari PHT tersebut masing-masing akan ditambahkan dengan kunci yang sudah terjadwal yaitu K2r+8 dan K2r+9. Hasil dari fungsi F adalah F0 dan F1, maka dengan demikian fungsi F telah terpenuhi.



Gambar 3.3 Fungsi *F*

Dari Gambar 3.3 Fungsi F dapat dilihat bahwa untuk mengambil tiga argumentasi, dua kata masukan R0 dan R1, dan angka bulat r digunakan untuk memilih subkey yang sesuai. Hasil whitening akan melalui fungsi F yang mempunyai outputF0 dan F1 dan masing-masing di-XORkan dengan R2 dan R3 (melalui rotasi ke kiri 1 bit).

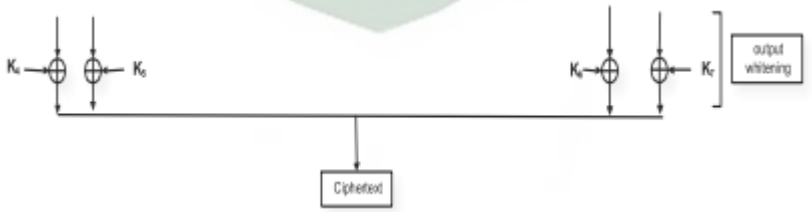
7. Setelah itu F0 dan F1 masing-masing di-XOR dengan R2 dan R3. Hasil dari R2 XOR F0 dirotasi ke kanan sejauh 1 bit. Sedangkan R3 XOR F1 , sebelumnya R3 dirotasi ke kiri sejauh 1 bit.
8. Setelah itu, maka akan dilakukan iterasi sebanyak 16 kali. Setiap iterasi sama dengan proses sebelumnya.
9. Hasil dari swap blok terakhir adalah penukaran bagian kanan dan kiri yang di-undo.



Gambar 3.4 *Swap* Blok Terakhir

Dari Gambar 3.4 dapat dilihat bahwa ada penukaran pada bagian kanan dan kiri yang di argumen pertama (32 bit word) disebelah kiri atau kanan dengan jumlah bit yang diindikasikan dengan argumen kedua.

10. Hasil dari 16 round enkripsi akan melalui output whitening yaitu proses peng-XORan 16 round enkripsi dengan K4, K5, K6, dan K7.



Gambar 3.5 *Output Whitening*

Dari Gambar 3.5 dapat dilihat bahwa setelah proses pertukaran pada bagian kanan dan kiri yang di undo, lalu memasuki tahap output whitening yang merupakan proses peng-XORan 16 round enkripsi.

Berikut pseudocode untuk pengujian kunci :

```
switch (true) {
        case $length <= 128:
          $this->key_length = 16;
          break;
        case $length <= 192:
          $this->key_length = 24;
          break;
        default:
          $this->key_length = 32;
```

Berikut *pseudocode* untuk proses enkripsi pada algoritma *Twofish* akan tampak seperti berikut ini :

```
$S0,S1,S2,S3,K
$in = unpack("V4", $in);
$R0 = $K[0] ^ $in[1];
$R1 = $K[1] ^ $in[2]
$R2 = $K[2] ^ $in[3]
$R3 = $K[3] ^ $in[4]
$ki = 7
```

```
while ($ki < 39)
    $t0 = $S0[ $R0        & 0xff] ^
         $S1[($R0 >>  8) & 0xff] ^
         $S2[($R0 >> 16) & 0xff] ^
         $S3[($R0 >> 24) & 0xff]
    $t1 = $S0[($R1 >> 24) & 0xff] ^
         $S1[ $R1        & 0xff] ^
         $S2[($R1 >>  8) & 0xff] ^
         $S3[($R1 >> 16) & 0xff];
    $R2^= $t0 + $t1 + $K[++$ki]
    $R2 = ($R2 >> 1 & 0x7fffffff) | ($R2 << 31)
    $R3 = (($R3 >> 31) & 1) | ($R3 << 1)) ^ ($t0 + ($t1 << 1) +
    $K[++$ki])
    $t0 = $S0[ $R2        & 0xff] ^
         $S1[($R2 >>  8) & 0xff] ^
         $S2[($R2 >> 16) & 0xff] ^
         $S3[($R2 >> 24) & 0xff];
    $t1 = $S0[($R3 >> 24) & 0xff] ^
         $S1[ $R3        & 0xff] ^
         $S2[($R3 >>  8) & 0xff] ^
         $S3[($R3 >> 16) & 0xff];
    $R0^= ($t0 + $t1 + $K[++$ki]);
    $R0 = ($R0 >> 1 & 0x7fffffff) | ($R0 << 31);
```

```
$R1 = (((($R1 >> 31) & 1) | ($R1 << 1)) ^ ($t0 + ($t1 << 1) +
$K[++$ki])
Return ("V4", $K[4] ^ $R2
$K[5] ^ $R3
$K[6] ^ $R0
$K[7] ^ $R1)
```

Berikut *pseudocode* untuk proses penyisipan pesan ke dalam gambar dengan metode *End of File* (*embedding):*

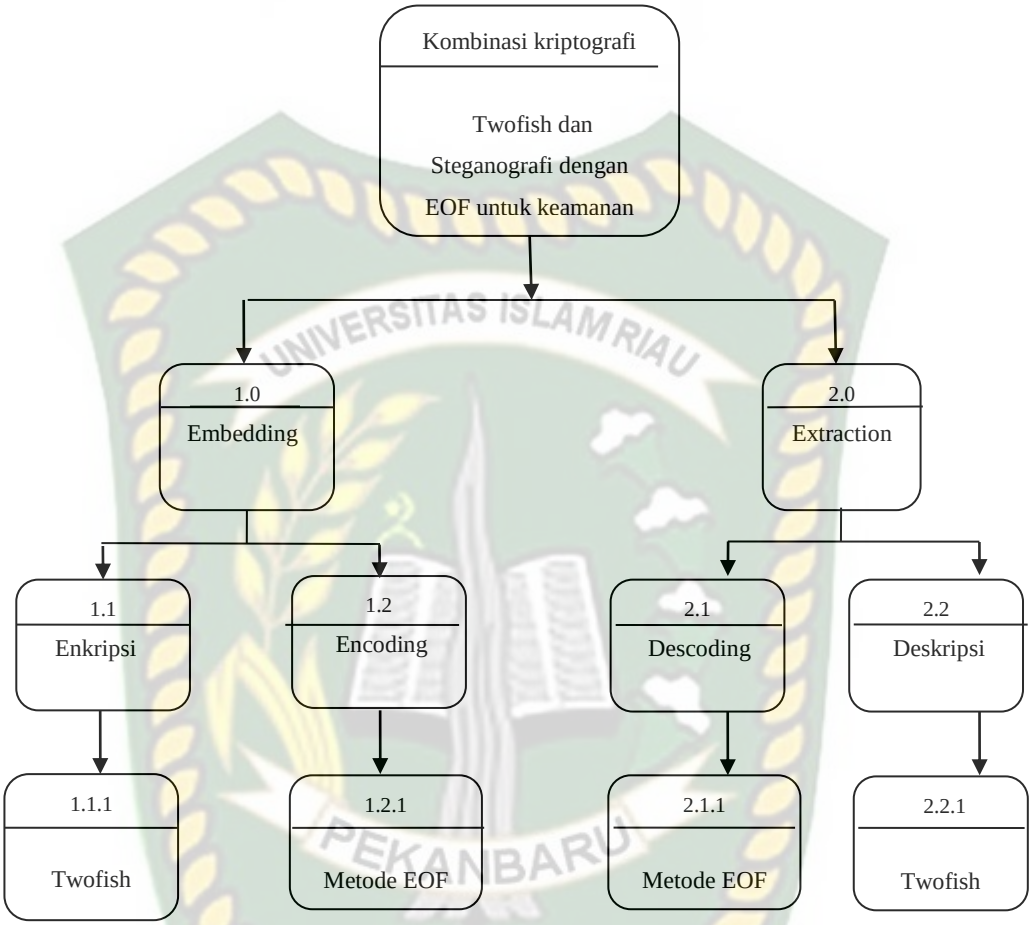
```
$x=0
for($iw=0;$iw<$width;$iw++)
for($ih=0;$ih<$height;$ih++)
$rgb = imagecolorat($im,$iw,$ih)
$r = ($rgb >>16) & 0xFF
$g = ($rgb >>8) & 0xFF
$b = $rgb & 0xFF
if($x<$message_length)
$newB = toBin($b)
$newB[strlen($newB)-1] = $binary_message[$x]
$newB = toString($newB)
$new_color = imagecolorallocate($im,$newR,$newG,$newB)
 imagesetpixel($im,$iw,$ih,$new_color)
$x++
```

3.4 Pengembangan dan Perancangan Sistem

Pada perancangan kombinasi kriptografi *Twofish* dan steganografi dengan EOF ini terdiri dari 2 proses utama. Pertama proses penyandian dan penyembunyian pesan pada gambar dan ekripsi pesan. Kedua ekstraksi atau pembacaan pesan yang telah disisipkan pada gambar dan dekripsi pesan.

3.4.1 *Hirarchy Chart*

Hirarchy chart adalah diagram yang menggambarkan permasalahan kompleks yang kemudian diuraikan dalam beberapa elemen, berikut gambaran *hirarchy chart* pada kombinasi kriptografi *Twofish* dengan EOF untuk keamanan data teks dapat dilihat pada gambar 3.6.

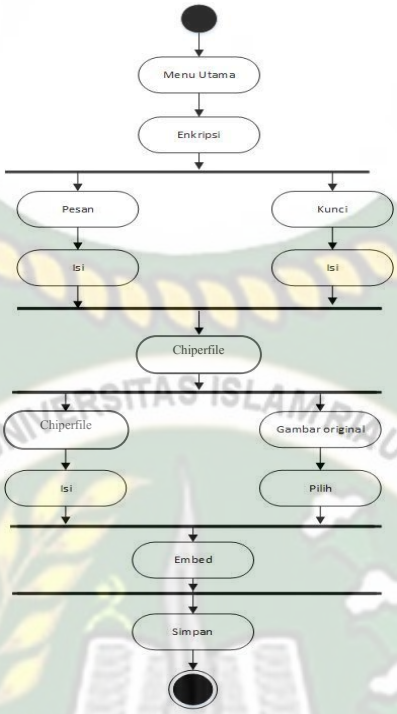


Gambar 3.6 Hirarchy Chart

Berdasarkan *Hirarchy Chart* pada gambar 3.6, terdapat 2 proses utama yang terdiri dari proses *Embedding* dan proses *Extraction.* Pada proses *Embedding* akan dilakukan enkripsi pesan dengan menggunakan algoritma kriptografi yaitu *Twofish*. Kemudian setelah pesan dienkripsi, pesan tersebut akan disembunyikan (*Encoding*) pada gambar (citra digital) dengan menggunakan metode *End of File.* Selanjutnya pada proses *Extraction*, stego *image* (gambar yang telah disisipi pesan teks) akan dilakukan *Descoding* dengan menggunakan metode *End of File*. Kemudian dilakukan deskripsi pada pesan dengan menggunakan algoritma kriptografi yaitu *Twofish*.

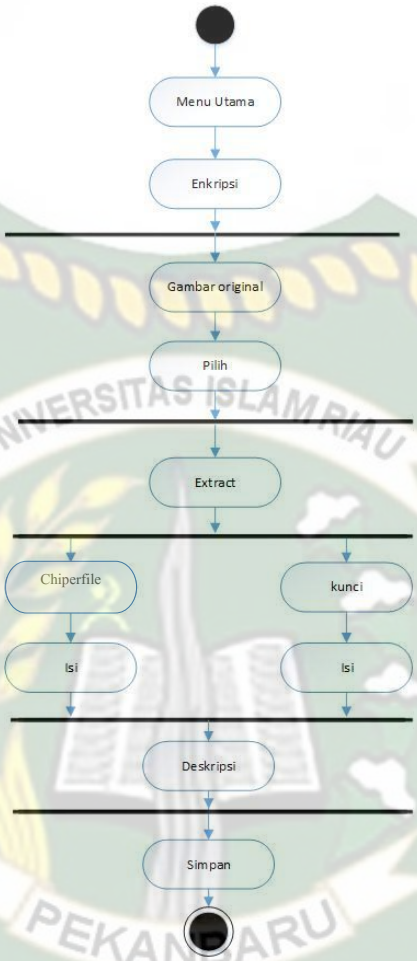
3.4.2 *Activity Diagram*

Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity Diagram* pada sistem ini terdiri atas 2 bagian, yaitu *Activity Diagram Embedding* dan *Activity Diagram Extraction. Activity Diagram Embedding* dapat dilihat pada gambar 3.7.



Gambar 3.7 *Activity Diagram Embedding*

Pada gambar 3.7 dapat dijelaskan bahwa proses *Embedding* terdiri dari beberapa tahap. Tahap pertama dimulai dari implementasi algoritma kriptografi *Twofish*. Dimana pada tahap ini akan dilakukan enkripsi terhadap pesan pdf. Untuk melakukan enkripsi, pengirim pesan terlebih dahulu harus menginputkan file yang akan dienkripsi beserta kata kuncinya. Kemudian, setelah dihasilkan *Chiperfile* dari hasil enkripsi tersebut, akan dilanjutkan dengan tahap kedua, yaitu tahap steganografi dengan menggunakan metode *End of File*. *Chiperfile* yang dihasilkan oleh algoritma *Twofish* akan dijadikan sebagai inputan pada tahap selanjutnya. Untuk melakukan penyisipan pesan pada gambar, *Chiperfile* diinputkan kembali bersama dengan gambar original (citra digital). Gambar ini digunakan sebagai media untuk menyisipkan pesan tersebut. Selanjutnya *Activity Diagram Extraction* dapat dilihat pada gambar 3.8



\Gambar 3.8 *Activity Diagram Extraction*

Pada gambar 3.8 dapat dijelaskan bahwa proses *extraction* terdiri dari beberapa tahap, yaitu tahap pertama gambar stegofile (gambar yang telah disisipkan pesan) yang telah dipilih akan dilakukan ekstraksi dengan menggunakan metode *End of File*. Dari hasil ekstraksi tersebut akan dihasilkan berupa file yang masih dalam bentuk *Chiperfile*. Kemudian selanjutnya akan dilakukan dekripsi dengan menggunakan algoritma *Twofish* untuk mengubah *Chiperfile* tersebut menjadi *Plainfile* agar pesan dapat dibaca oleh penerima pesan.

3.4.3 *Use Case Diagram*

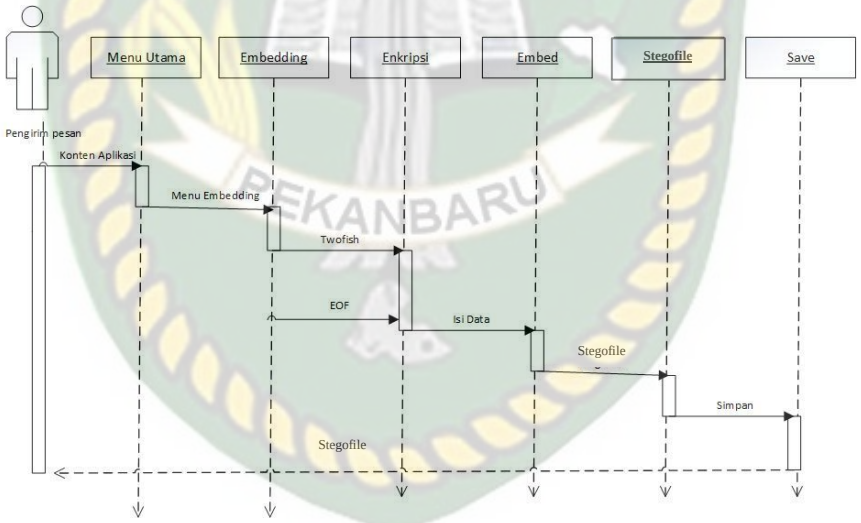
Pada perancangan aplikasi kombinasi kriptografi *Twofish* dan steganografi dengan EOF ini terdiri dari dua proses utama yang dilakukan oleh pengirim pesan yaitu, proses Embedding pesan dan Enkripsi pesan, dan dua proses utama yang dilakukan oleh penerima pesan yaitu, Extraction pesan dan Dekripsi pesan dengan menggunakan metode *Twofish* dan *End of File. User Case* aplikasi yang akan dibangun dapat dilihat pada gambar 3.9 dibawah ini.

Gambar 3.9 *Use Case Sistem Yang Dibangun*

Pada gambar 3.9 diatas dapat dilihat pada aplikasi yang akan dibangun terdiri dari 2 aktor, pertama pengirim pesan dan kedua penerima pesan. Pada aplikasi ini terdiri dari 5 case, *embedding*, *extraction*, *user guide* dan *about*.

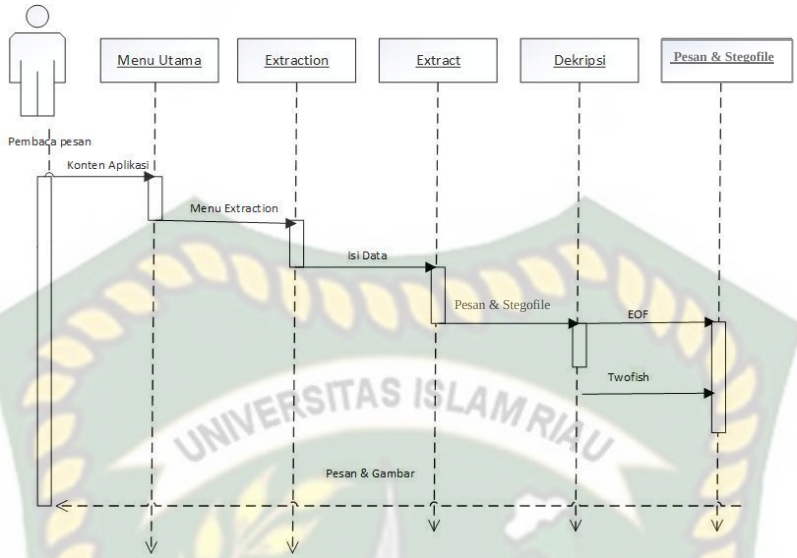
3.4.4 *Sequence* Diagram

Sequence diagram digunakan untuk mengetahui tentang alur proses dan interaksi antara objek pada aplikasi yang akan dibangun. Dengan menggunakan *sequence diagram* kita dapat melihat bagaimana objek-objek bekerja. *Sequence diagram* dapat menampilkan bagaimana sistem merespon setiap kejadian atau permintaan dari *user*, dapat mempertahankan integritas internal, bagaimana data dipindahkan ke *user interface* dan bagaimana objek-objek diciptakan dan dimanipulasi, *Sequence Diagram* pada proses pengirim pesan dapat dilihat pada gambar 3.10 dibawah ini.



Gambar 3.10 *Sequence Diagram* Pengirim Pesan

Berdasarkan gambar 3.10 diatas dengan actor membuat pesan dapat dilihat langkah-langkah yang dilakukan actor pengirim pesan mulai dari menjalankan aplikasi sampai dengan melakukan proses penyandian, penyembunyian dan mengenkripsi pesan pada gambar, kemudian menyimpan gambar stegofile.



Gambar 3.11 *Sequence Diagram* Penerima Pesan

Berdasarkan gambar 3.11 diatas dengan actor pembaca pesan dapat dilihat langkah-langkah yang dilakukan actor pembaca pesan mulai dari menjalankan aplikasi sampai dengan melakukan proses pembacaan pesan yang ada pada gambar stegofile.

3.4.5 *Class Diagram*

Class Diagram menggambarkan struktur dan dekripsi *class, package,* dan objek yang saling terhubung. *Class Diagram* yang dijelaskan pada analisa ini adalah *class Diagram* pada aplikasi yang akan dibangun, seperti gambar 3.12 dibawah ini

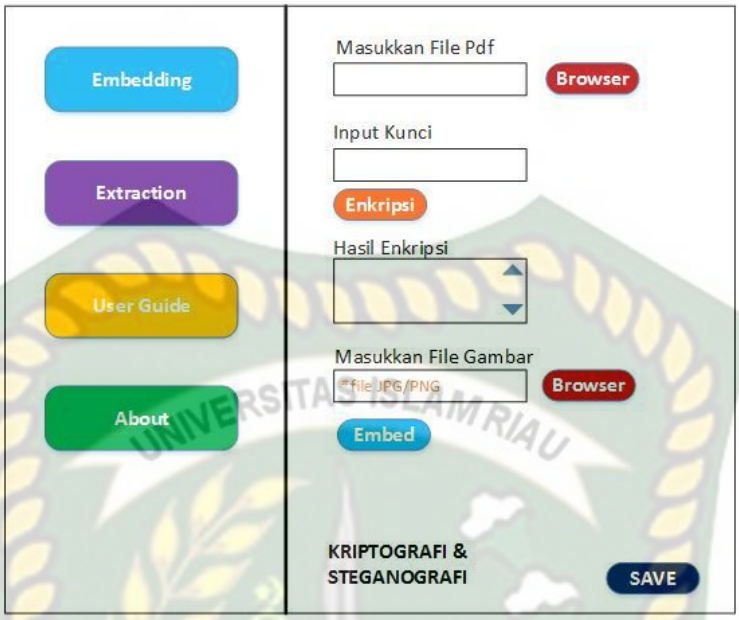
Gambar 3.12 *Class Diagram*

3.5 Perancangan Input

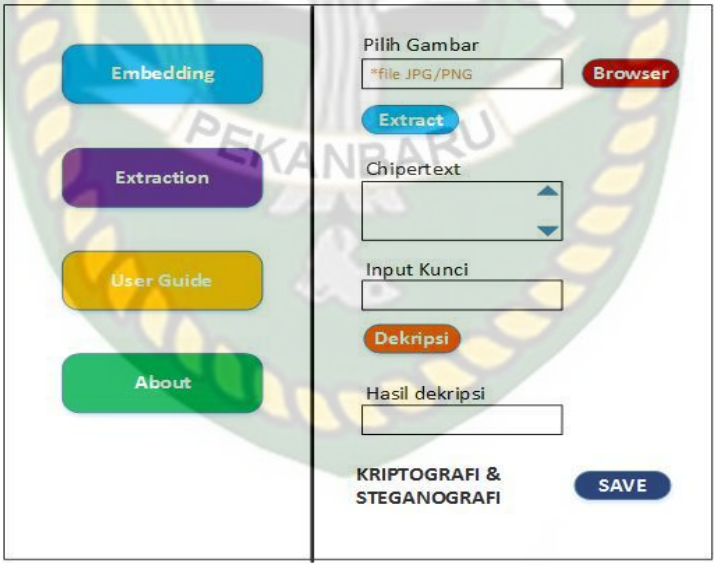
Desain input merupakan perancangan desain masukan dari pengguna kepada sistem.

3.5.1 Desain Input Embedding

Desain input embed ini merupakan bentuk tampilan yang digunakan untuk melakukan proses input pesan, input kunci dan pilih gambar. Tampilan input *embedding* dan *extraction* dapat dilihat pada gambar 3.13 dan 3.14 dibawah ini.



Gambar 3.13 Desain Input *Embedding*



Gambar 3.14 Desain Input *Extraction*

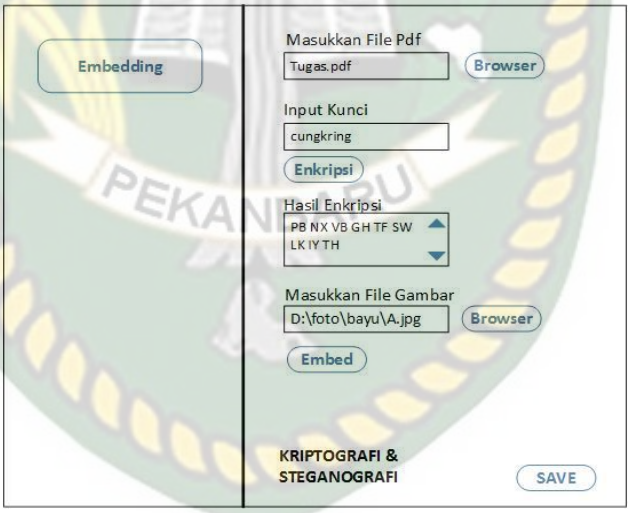
Pada gambar 3.14 diatas dapat dilihat, yang menjadi input pada proses ini yaitu stegofile, kunci dekripsi yang akan menampilkan pesan asli.

3.6 Perancangan *Output*

Desian *output* merupakan rancangan tampilan *output* atau hasil dari sistem setelah melakukan proses yang terdiri dari *embedding* dan *extraction.*

3.6.1 Desain Output Ekstraksi

Hasil output merupakan rancangan bentuk tampilan output dari sistem setelah melakukan proses *embedding* dan *extraction* berupa pesan rahasia. Adapun hasil dari output dapat dilihat pada gambar 3.15 dan 3.16 dibawah ini.



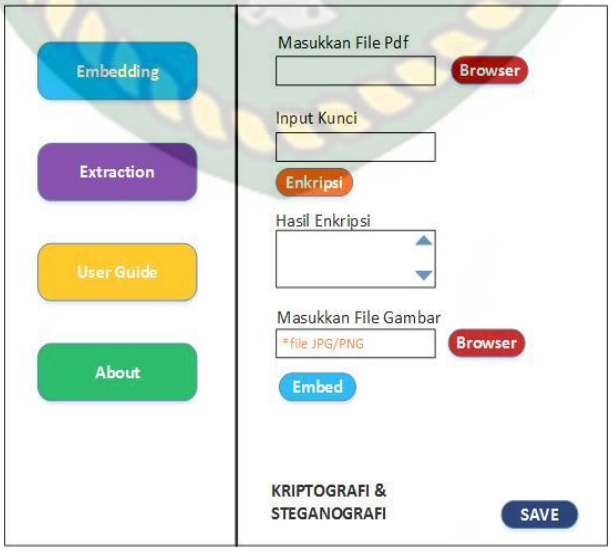
Gambar 3.15 Desain Output *Embedding*

Gambar 3.16 Desain Output *Extraction*

3.7 Desain *Interface*

3.7.1 Halaman Utama

Pada halaman utama akan menampilkan halaman awal pada sistem pertama kali dijalankan. Adapun tampilan halaman utama dapat dilihat pada Gambar 3.17.



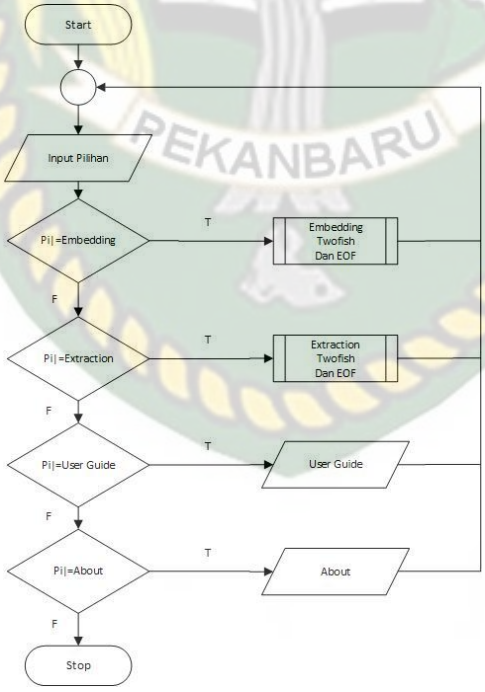
Gambar 3.17 Halaman Utama

Pada gambar 3.17 diatas terdapat 5 menu pilihan yaitu pertama *embedding*, merupakan menu untuk menyembunyikan, penyandian pesan dan mengenkripsi pesan ke gambar. Kedua *extraction*, merupakan membaca pesan yang disisipkan. Ketiga menu *user guide* atau tata cara menggunakan aplikasi yang dibuat, keempat about berisi tentang aplikasi yang dibuat.

3.8 Perancangan Logika Program

Perancangan logika program akan memberikan gambaran bagaimana sistem bekerja mulai dari proses *input* sampai dengan proses *output*. Dan memberikan gambaran kinerja sistem yang terstruktur dan sistematis.

3.8.1 *Flowchart* Menu Utama

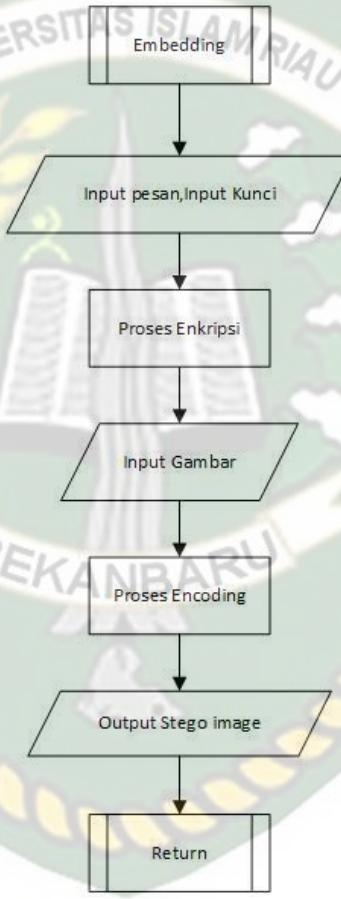


Gambar 3.18 *Flowchart* Menu Utama

Pada gambar 3.18 diatas dapat dilihat, pada saat program dijalankan atau *start* akan ditampilkan menu utama dan terdapat menu pilihan yaitu *Embedding*,

Extraction, *User Guide*, dan *About*. Ketika salah satu menu dipilih, akan menampilkan tampilan menu tersebut dan ketika selesai akan kembali ketampilan menu utama, setelah *stop* program akan keluar.

3.8.2 *Flowchart Embedding*

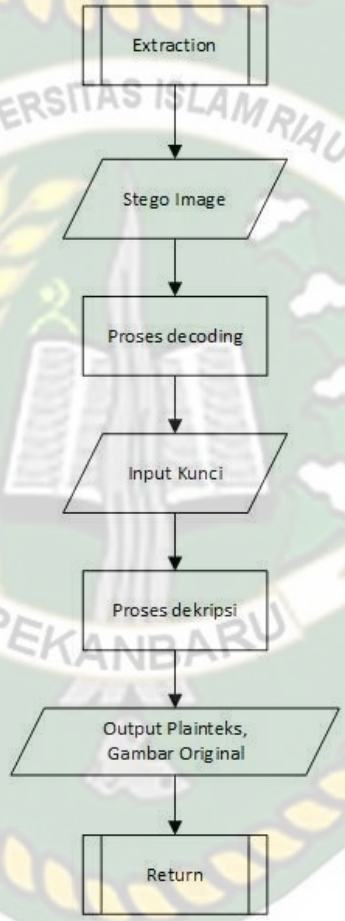


Gambar 3.19 *Flowchart* Menu *Embedding*

Pada gambar 3.19 diatas dapat dilihat, pada saat menu embedding maka akan tampil menu embedding maka pilih embedding. Pada menu embedding terdapat beberapa inputan yaitu, pesan, kunci dan gambar. Pada menu embedding

terdapat 2 proses yaitu enkripsi dan encoding, yang menjadi outputnya *stegofile* dan setelah di return pada program akan kembali ke menu utama.

3.8.3 *Flowchart Extraction*



Gambar 3.20 *Flowchart* Menu *Extraction*

Pada gambar 3.20 diatas dapat dilihat, pada saat menu *extraction* dipilih akan menampilkan menu *extraction.* Pada menu ini terdapat menu extraction dan terdapat beberapa inputan decoding dan dekripsi, yang menjadi outputnya pesan rahasia dan setelah direturn program akan kembali ke menu utama.

3.9 Evaluasi

Pengembangan sistem keamanan file dengan menggunakan kriptografi *Twofish* dan steganografi EOF dapat meningkatkan keamanan suatu pesan yang kompleks. Hal ini disimpulkan berdasarkan kombinasi dari dua jenis keamanan tersebut (kriptografi *Twofish* dan steganografi EOF).

Pada dasarnya masing-masing teknik pengamanan memiliki kekuatan masing-masing. Pertama pada teknik kriptografi menggunakan *Twofish* yang merupakan teknik kriptografi yang memiliki tingkat kekuatan dalam mengenkripsi pesan. Sementara pada teknik steganografi EOF yang merupakan teknik pengamanan yang menggunakan media citra digital sebagai basis untuk menyembunyikan pesan yaitu pesan akan disisipkan pada sebuah gambar.

Berdasarkan uraian tersebut kombinasi dari kriptografi dan steganografi ini memiliki level kekuatan pengamanan yang cukup kuat karena enkripsi dilakukan dengan 2 tahap. Pertama menyandikan pesan, selanjutnya pesan sandi tersebut disembunyikan dalam media citra digital berbentuk gambar dengan format .JPG dan .PNG.

Selanjutnya, pertahanan yang dimiliki pada sistem ini untuk menghadapi serangan kriptanalis yang menggunakan metode *Brute Force Attack* atau serangan bertubi-tubi yang dilakukan dengan menggunakan prinsip *trial* dan *error* termasuk cukup rumit. Apabila dengan asumsi kriptanalis mengetahui algoritma kriptografi yang digunakan pada sistem ini, kriptanalis akan membutuhkan waktu yang cukup lama untuk mencoba setiap kemungkinan kunci yang digunakan.

Selain itu, jika kriptanalis mampu menemukan kunci kriptografi yang digunakan, tidak dapat langsung membuka pesan yang telah dienkripsi dengan algoritma kriptografi tersebut, melainkan untuk mengubah pesan keseluruhan, harus melewati tahapan enkripsi berikutnya yaitu steganografi.

BAB IV

HASIL DAN PEMBAHASAN

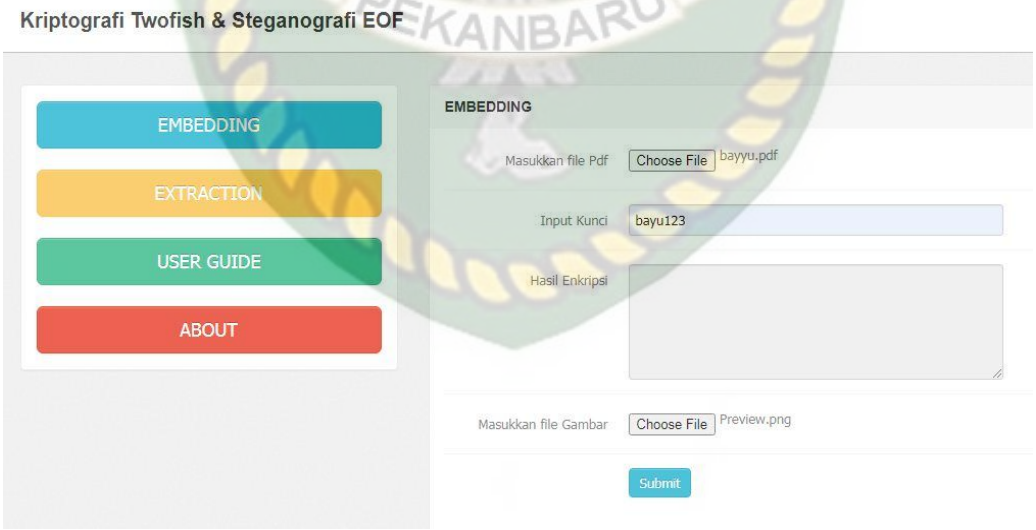
4.1 Hasil Penelitian

Sebelum aplikasi yang dibangun dipublikasikan, ada beberapa tahapan yang harus dilakukan, hal ini dimaksudkan agar sewaktu aplikasi benar-benar sudah dipublikasikan tidak terjadi lagi kesalahan. Dalam pengujian sistem ini dilakukan dengan metode *black box*.

4.2. Pengujian *Black Box*

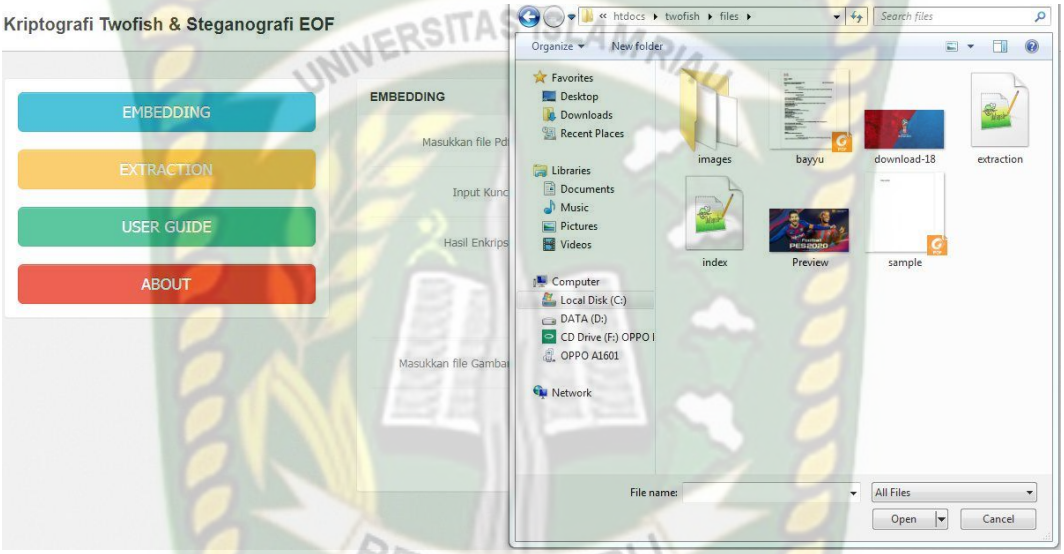
Pengujian *black box* (*black box testing*) adalah salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas, khususnya pada *input* dan *output* aplikasi.

4.2.1 Pengujian Black Box Form Embedding



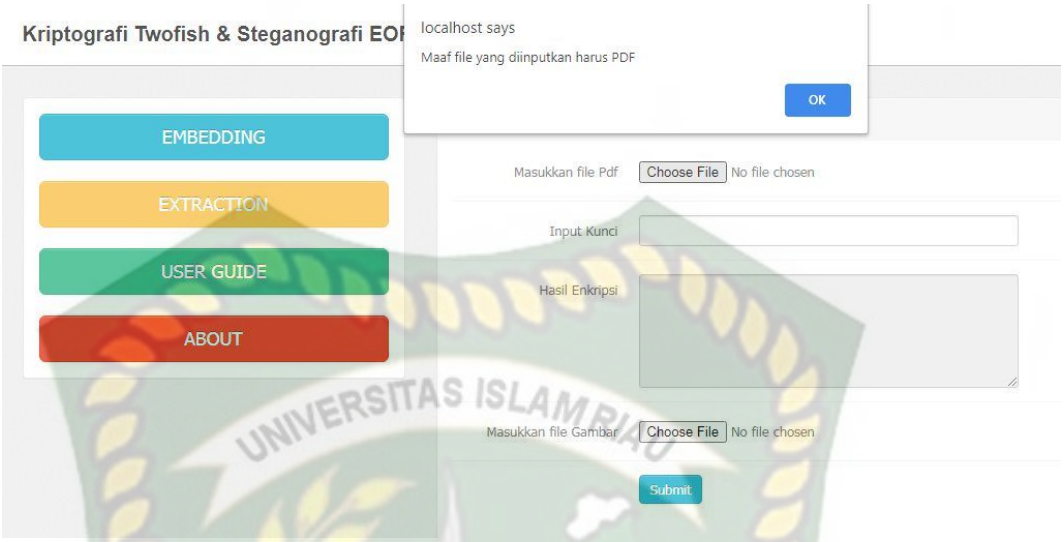
Gambar 4.1 Form Menu Embedding

Pada gambar 4.1 dijelaskan bahwa ketika pengguna membuka menu embedding maka terdapat 3 field yang harus diisi. Field pertama adalah pilih file pdf, file yang kedua adalah inputkan kunci dan file yang ketiga adalah pilih file gambar. File pdf adalah file yang akan disembunyikan. Sedangkan file gambar adalah file tempat file pdf akan disembunyikan.



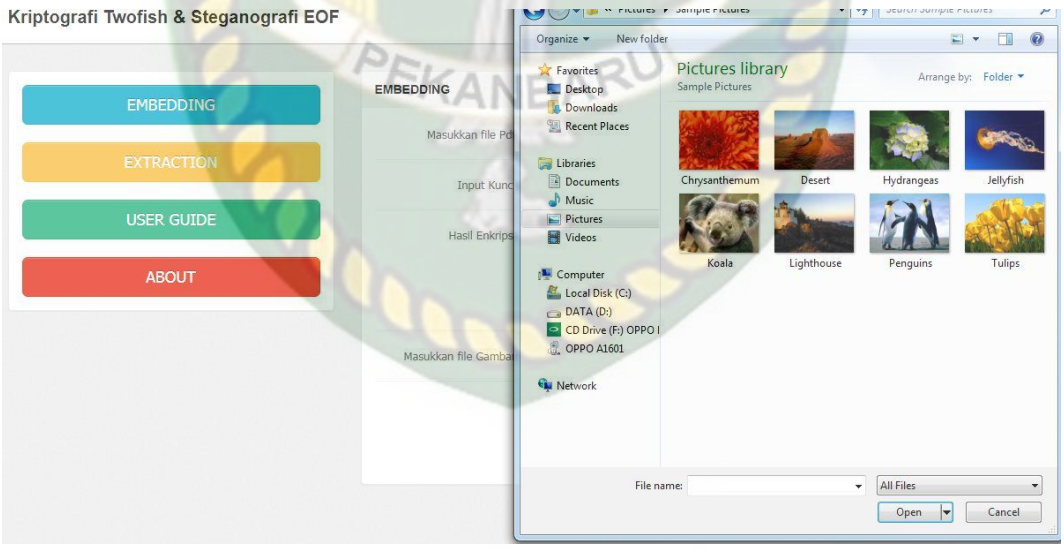
Gambar 4.2 Pengujian Tombol Choose File Pdf

Pada gambar 4.2 dijelaskan bahwa ketika pengguna menekan tombol Choose file maka sistem akan membuka kotak dialog *mycomputer* dan menampilkan file untuk dapat dipilih sebagai file pdf.



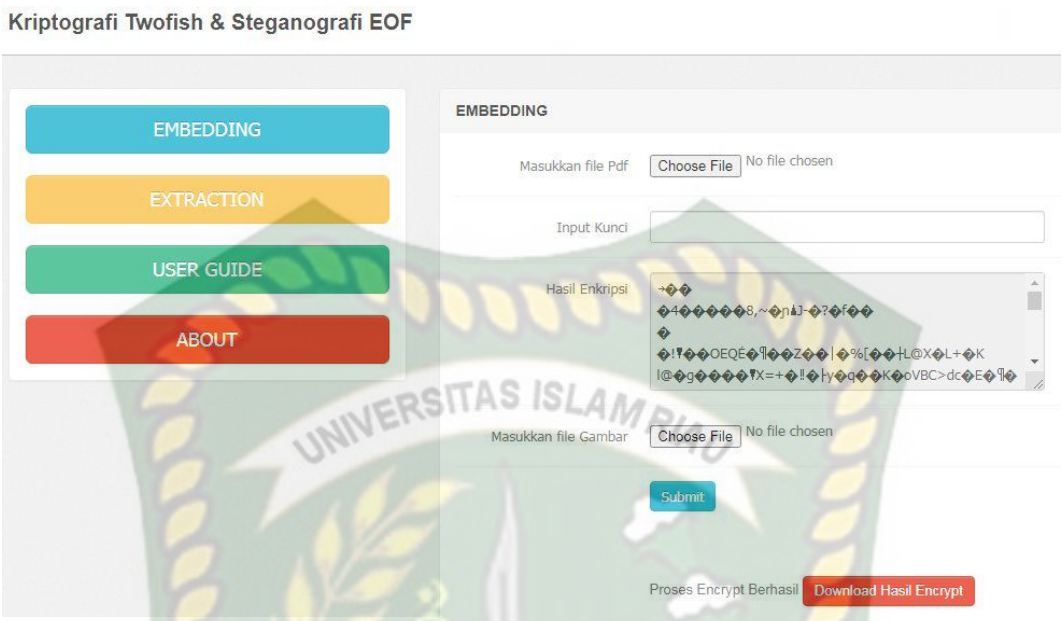
Gambar 4.3 *Message dialog* peringatan file yang dipilih harus pdf.

Pada gambar 4.3 dijelaskan bahwa apabila format file yang dipilih bukan pdf maka sistem akan membuka kotak message dialog.

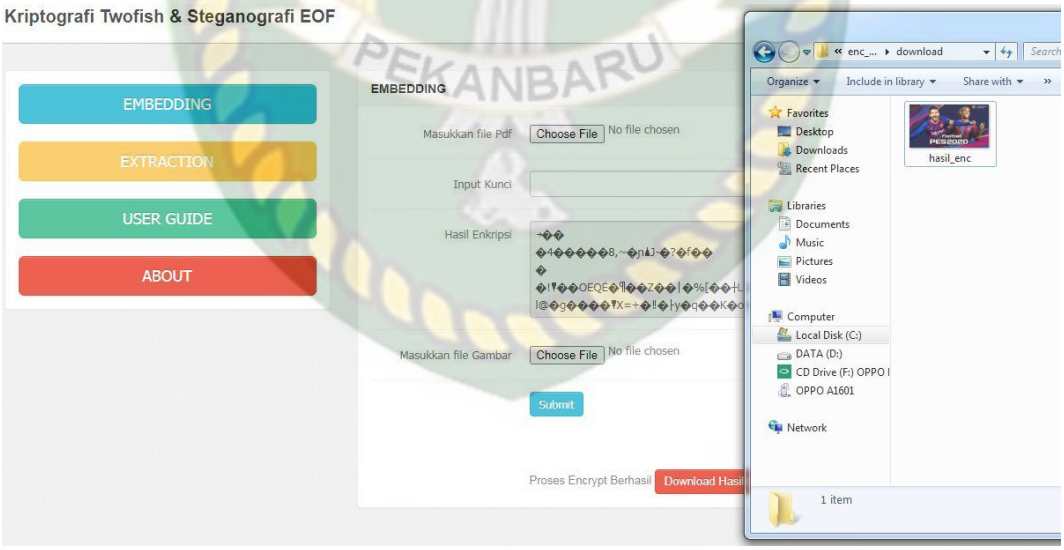


Gambar 4.4 Pengujian Tombol Choose File Gambar

Pada gambar 4.4 dijelaskan bahwa ketika pengguna menekan tombol Choose file maka sistem akan membuka kotak dialog *mycomputer* dan menampilkan file untuk dapat dipilih sebagai file gambar.



Gambar 4.5 Pengujian Tombol *Submit*



Gambar 4.6 Pengujian Tombol *Submit(2)*

Pada gambar 4.5 dijelaskan bahwa ketika pengguna menekan tombol *Submit* maka sistem akan memproses enkripsi file apabila enkripsi berhasil file

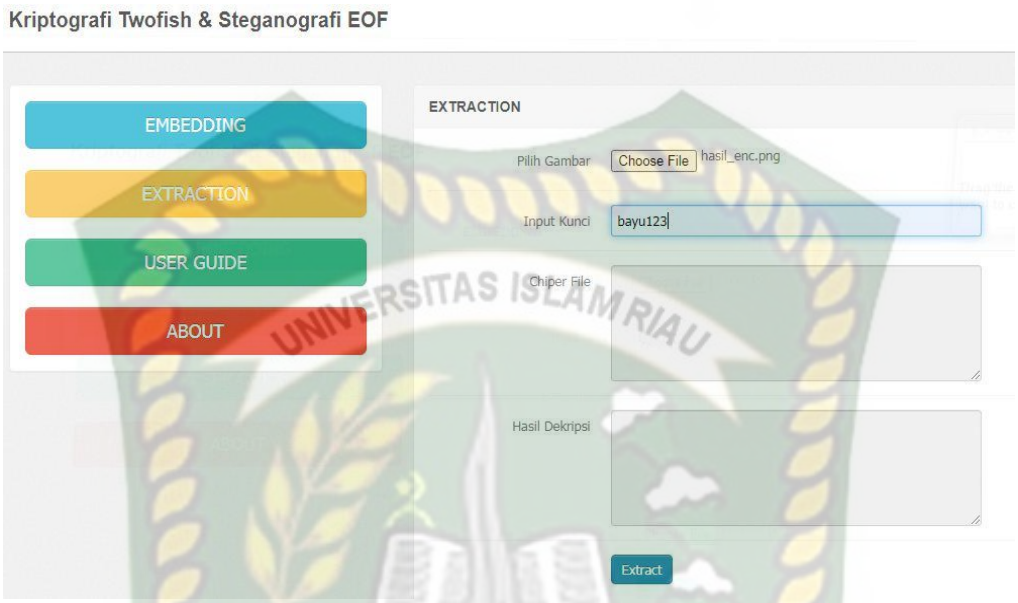
bisa didownload dapat dilihat pada gambar 4.5. Sistem akan menyembunyikan file hasil enkripsi kedalam file gambar yang dapat dilihat pada gambar 4.6.

Tabel 4.1 Pengujian *Black Box Form Embedding*

Komponen yang diuji	Skenario Penguji	Hasil yang diharapkan	Hasil
Tombol *choose* file pdf	Menekan Tombol *choose* file	Sistem akan menampilkan kotak dialog baru untuk memilih file yang diinginkan	[✓]Sesuai Harapan [ ] Tidak Sesuai Harapan
Input kunci	Mengisi kata kunci	-	[✓]Sesuai Harapan [ ] Tidak Sesuai Harapan
Tombol *choose* file gambar	Menekan Tombol *choose* file	Sistem akan menampilkan kotak dialog baru untuk memilih file yang diinginkan	[✓]Sesuai Harapan [ ] Tidak Sesuai Harapan
Proses Enkripsi	Menekan Tombol submit	Sistem akan memproses enkripsi, lalu stegano. Memunculkan hasil enkripsi dan memunculkan tombol download hasil enkripsi.	[✓]Sesuai Harapan [ ] Tidak Sesuai Harapan

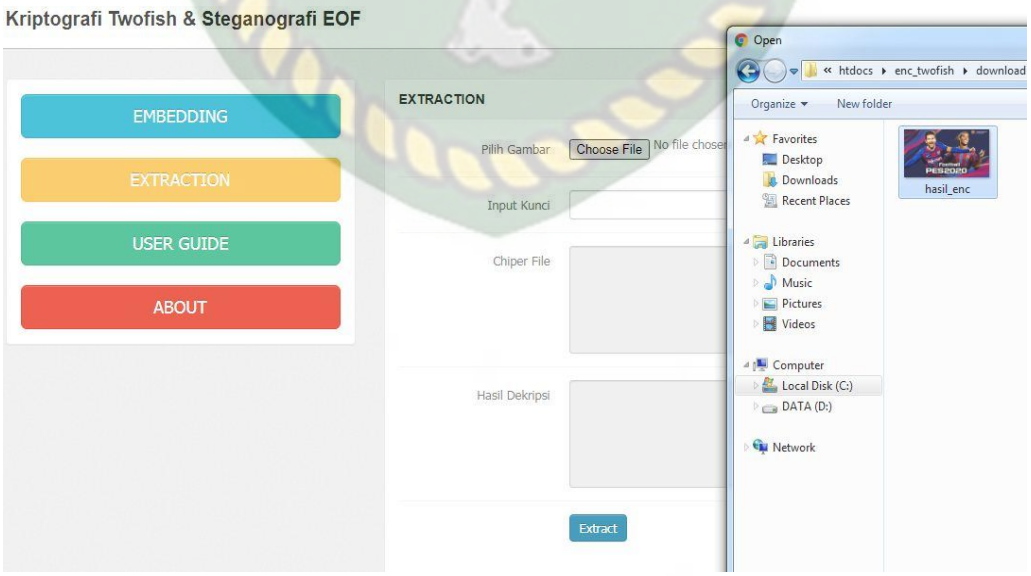
Berdasarkan pengujian yang telah dilakukan dapat ditarik kesimpulan bahwa dalam pengujian *black box* yang telah dilakukan terhadap sistem pada form embedding telah sesuai dengan harapan.

4.2.2 Pengujian Black Box Form Extraction



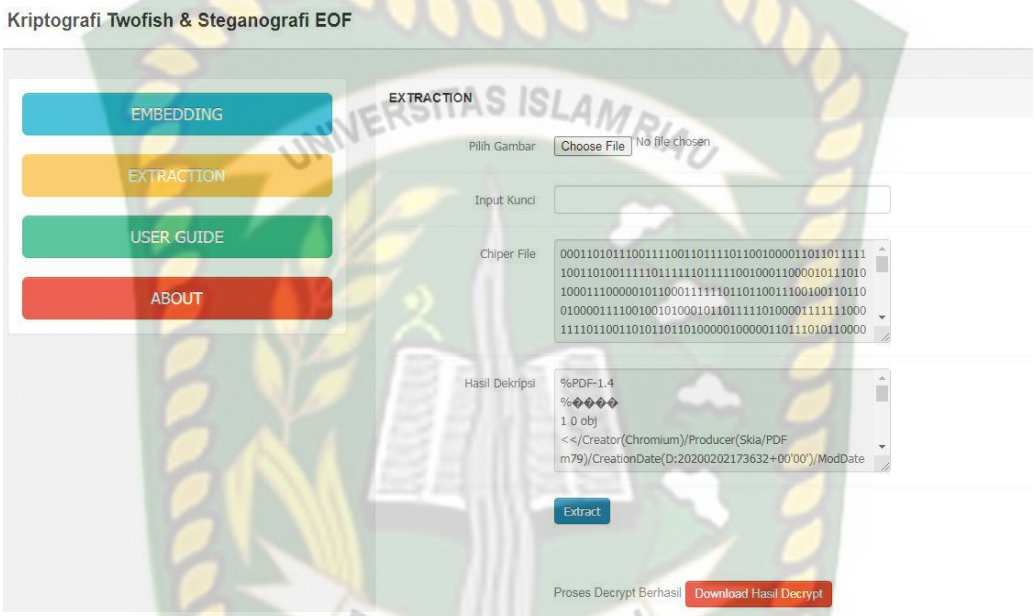
Gambar 4.7 Pengujian Menu Extraction

Pada gambar 4.7 dijelaskan bahwa ketika pengguna membuka menu extraction maka terdapat 2 *field* yang harus diisi yaitu Choose file dan input kunci.

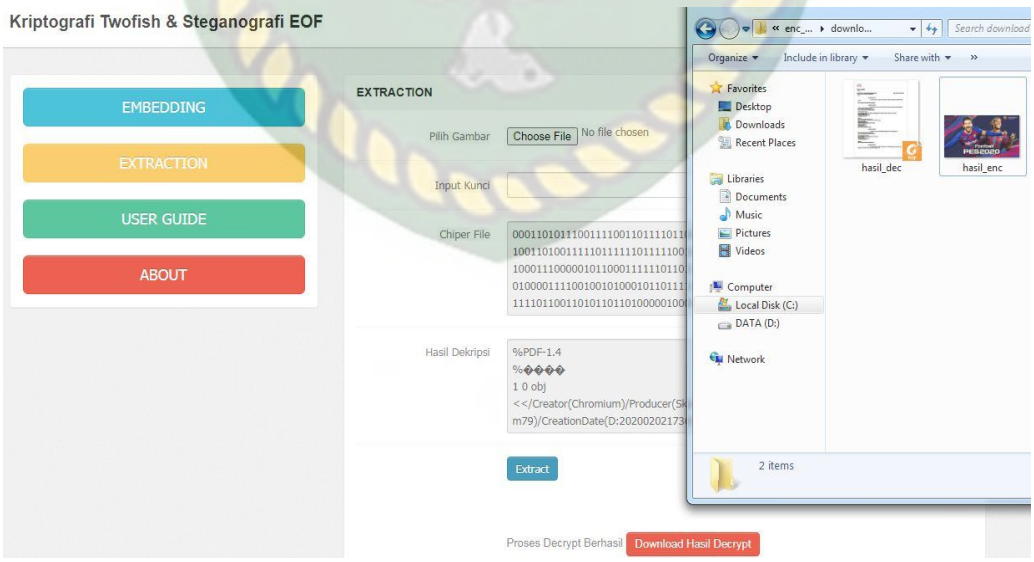


Gambar 4.8 Pengujian Tombol Choose File Gambar

Pada gambar 4.8 dijelaskan bahwa ketika pengguna menekan tombol Choose file maka sistem akan membuka kotak dialog *mycomputer* dan menampilkan file untuk dapat dipilih sebagai file gambar. Pengguna dapat memilih file yang akan diuraikan.



Gambar 4.9 Pengujian Tombol *Extract*



Gambar 4.10 Pengujian Tombol *Extract(2)*

Pada gambar 4.9 dijelaskan bahwa setelah memasukan keyword yang tepat, sistem akan melakukan proses penguraian file stegano dan melakukan deskripsi. Kemudian akan muncul tombol download hasil dekripsi. File hasil penguraian akan disimpan didalam folder yang telah ditentukan seperti gambar 4.10.

Tabel 4.2 Pengujian *Black Box Form Deskripsi*

No.	Komponen yang diuji	Skenario Penguji	Hasil yang diharapkan	Hasil
1.	Tombol *Choose* File	Menekan Tombol *Choose* File	Sistem akan menampilkan kotak dialog baru untuk memilih file yang diinginkan	[✓]Sesuai Harapan [ ] Tidak Sesuai Harapan
2.	Input kunci	Pengguna memasukan keyword yang sesuai	-	[✓]Sesuai Harapan [ ] Tidak Sesuai Harapan
3.	Tombol Extract	Menekan tombol extract	Sistem melakukan proses penguraian stegano dan deskripsi Kriptografi, lalu memunculkan tombol download hasil deskripsi.	[✓]Sesuai Harapan [ ] Tidak Sesuai Harapan

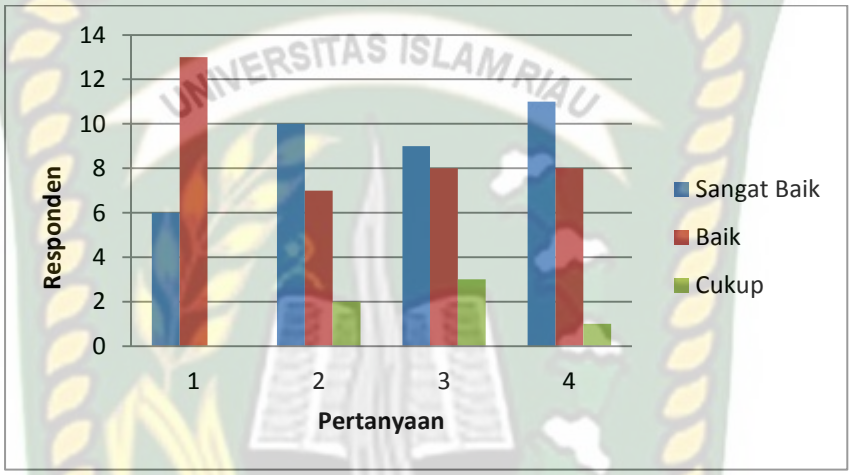
4.4 Pengujian Sistem Terhadap Pengguna

Salah satu pengujian sistem terhadap pengguna yang dilakukan yaitu dengan membagikan kuisioner kepada mahasiswa/i teknik informatika. Sebanyak 20 lembar kuisioner yang berisikan 4 pertanyaan. Adapun pertanyaan yang diberikan kepada responden adalah sebagai berikut :

1. Apakah Aplikasi mudah digunakan (*User Friendly*)?
2. Apakah bahasa yang digunakan dalam aplikasi ini dapat dimengerti dengan baik?

3. Apakah sistem ini bermanfaat?
4. Apakah sistem kombinasi Kriptografi ini dapat menjaga keamanan data dengan baik?

Tanggapan dari responden terhadap kinerja atau performance dari sistem berdasarkan pertanyaan yang diajukan adalah sebagai berikut :



Gambar 4.11 Grafik Hasil Kuisioner

Pada tabel 4.3 akan dijelaskan grafik hasil kuisioner yang menunjukan nilai untuk setiap pertanyaan-pertanyaan diatas adalah sebagai berikut :

Tabel 4.3 Jawaban Responden Terhadap Kuisioner

Pertanyaan	Sangat Baik	Baik	Cukup
1	6	13	0
2	10	7	2
3	9	8	3
4	11	8	1
Total	36	36	6

4.4.1 Hasil Presentase Kuisioner

Berdasarkan hasil kuisioner tersebut maka dapat disimpulkan bahwa sistem operasi LabIT ini memiliki persentase sebagai berikut :

1. Sangat baik = $\frac{6+10+9+11}{20*4} = \frac{36}{80}$ = 0.45 = 45%
2. Baik = $\frac{13+7+8+8}{20*4} = \frac{36}{80}$ = 0.45 = 45%
3. Cukup = $\frac{0+2+3+1}{20*4} = \frac{6}{80}$ = 0.075 = 7.5%

Dari hasil persentase tabel diatas, dapat diambil kesimpulan bahwa Teknik Penyembunyian Pesan PDF Terenkripsi Menggunakan Algoritma *Twofish* Dan Steganografi *End Of File* (EOF) Dalam Media Gambar ini memiliki performance sangat baik sebesar 45% dan baik sebesar 45% dengan nilai persentase kuisioner rata-rata sebesar 90% sehingga sistem ini dapat diimplementasikan.

4.5 Waktu yang dibutuhkan saat implementasi sistem

Berikut ini waktu yang dibutuhkan sistem dalam melakukan proses Kriptografi dan steganografi saat sistem dijalankan :

Tabel 4.4 Waktu yang dibutuhkan untuk Proses Kripto dan Stegano

No	Nama File	Ukuran File	Nama Gambar	Ukuran Gambar	Ukuran gambar Setelah Stegano Dan Kripto	Waktu Eksekusi Program Saat Enkripsi (Detik)	Waktu Eksekusi Program Saat Dekripsi (Detik)	Ukuran Gambar Setelah Dekripsi	Ukuran File setelah Dekripsi
1	Sample1.pdf	18 KB	Test1.png	1.009 KB	1.024 KB	10.5 S	2 S	1.009 KB	18 KB
2	Sample2.pdf	291 KB	Test2.png	1.620 KB	1.632 KB	12.21 S	3.2 S	1.620 KB	291 KB
3	Proposal.pdf	657 KB	Images.png	1.503 KB	1.645 KB	15 S	5.2 S	1.503 KB	657 KB
4	Jurnal.pdf	1.115 KB	Images1.png	1.324 KB	1.562 KB	21.2 S	8 S	1.324 KB	1.115 KB
5	Resume.pdf	2.606 KB	Images3.png	6.800 KB	6.974 KB	33.4 S	10 S	6.800 KB	2.606 KB
6	Skripsi.pdf	2.683 KB	Gambar.png	5.330 KB	5.523 KB	31 S	10 S	5.330 KB	2.683 KB
7	Lampiran.pdf	4.971 KB	Gambar2.png	5.480 KB	5.985 KB	45 S	14 S	5.480 KB	4.971 KB

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisa dan pengujian pada bab sebelumnya dalam Teknik Penyembunyian Pesan PDF Terenkripsi Menggunakan Algoritma *Twofish* Dan Steganografi *End Of File* (EOF) Dalam Media Gambar, maka dapat ditarik beberapa kesimpulan yaitu :

1. Dengan adanya sistem Kombinasi Kriptografi dan Steganografi dapat membantu user menjaga kerahasiaan sebuah pesan sehingga sampai ke tangan orang yang berhak.
2. Dapat menghalangi serangan yang dilakukan oleh kriptanalis dengan mengggunakan Kriptografi dan Steganografi.
3. Hasil simulasi program adalah citra awal sebelum disisipkan pesan/file dan citra sesudah disisipkan pesan secara kasat mata susah dibedakan.

5.2 Saran

Adapun saran dari penulis dalam Teknik Penyembunyian Pesan PDF Terenkripsi Menggunakan Algoritma *Twofish* Dan Steganografi *End Of File* (EOF) Dalam Media Gambar ini adalah sebagai berikut:

1. Penelitian selanjutnya diharapkan dapat mengembangkan sistem ini untuk perangkat *MOBILE*.
2. Mengkaji lebih lanjut mengenai kombinasi algoritma kriptografi dengan metode dan data yang lain selain file, seperti video, ataupun audio

DAFTAR PUSTAKA

Anhar., 2010, *Panduan Menguasai PHP & MYSQL Secara Otodidak,* Media Kita, Jakarta

Ariyus, Dony., 2008, *Pengantar Ilmu Kriptografi,* ANDI offset, Yogyakarta.

Arifpriyanto, B. (2013).Penyembunyian Pesan Text Terenkripsi Menggunakan Metode Kriptografi Stream Cipher dan Steganografi End Of File (EOF) dengan File Induk PDF. *Dokumen Karya Ilmiah Tugas Akhir Program Studi Teknik Informatika – S1 Fakultas Ilmu Komputer Universitas Dian Nuswantoro Semaranng 2013, 2013, 1-6*

Edisuryana Mukharrom, Isnanto R Riza, and Somantri Maman, "Aplikasi Steganografi Pada Citra Berformat Bitmap Dengan Menggunakan Metode End of File," *Transien*, vol. 2, no. 3, pp. 1–9, 2013.

Mu'mi, Nurul Fitriani Andi, 2017, Steganografi Citra Menggunakan Kriptografi *Hybrid Playfair Cipher* dan *Caesar Cipher*, Juli 2017.

Sukrisno, & Utami, E. (2007). Implementasi Steganografi Teknik EoF dengan Gabungan Enkripsi Rijndael, Shift Cipher dan Fungsi Hash MD5. *Seminar Nasional Teknologi 2007 (SNT 2007),* (November), 1-16.

Munir, R. 2004, *Diktat Kuliah IF5054 Kriptografi, Steganografi dan Watermarking*. Institut Teknologi Bandung.

Stalling, William, 2014, *Cryptography And Network Security Principles And Practice,* Person Education, Boston

Wijaya,Ermadi Satriya, dan Yudi Prayudi, 2004, *Konsep Hidden Message Menggunakan Teknik Steganografi Dynamic Cell Spreading*, Universitas Islam Indonesia, Yoyakarta.

Ratih, *Studi dan Implementasi Enkripsi Pengiriman Pesan Suara Menggunakan Algoritma Twofish.* National Conference On Computer Science & Information Technology VII, 2007.

Seti, *Enkripsi dan Deskripsi File Dengan Algoritma Twofish, Universitas Sumatera Utara*, 2009.