TUGAS AKHIR

ANTI FORENSIK DALAM PENYAMARAN FILE DENGAN KRIPTOGRAFI DAN STEGANOGRAFI

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Strata- 1
Teknik Informatika

Oleh

Nama : Gerry Junior Wibawa

No. Mahasiswa : 07523169

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
2012

PENGAKUAN

Demi Allah, Saya akui karya ini adalah hasil kerja saya sendiri kecuali beberapa isi dari landasan teori dan ringkasannya pada penelitian ini yang setiap satunya telah saya jelaskan sumbernya, serta segala macam tool dan aplikasi yang saya gunakan yang dimana merupakan hasil karya orang lain. Jika di kemudian hari ternyata terbukti pengakuan saya ini tidak benar dan melanggar peraturan yang sah dalam karya tulis dan hak intelektual maka saya bersedia ijazah yang telah saya terima untuk ditarik kembali oleh Universitas Islam Indonesia.

Yogyakarta, Januari 2012

Penyusun,

Gerry Junior Wibawa

ANTI FORENSIK DALAM PENYAMARAN FILE DENGAN KRIPTOGRAFI DAN STEGANOGRAFI

TUGAS AKHIR

Oleh

Nama : Gerry Junior Wibawa

No. Mahasiswa : 07523169

Yogyakarta, Januari 2012

Pembimbing I

YudiPrayudi, S.Si, M.Kom

ANTI FORENSIK DALAM PENYAMARAN FILE DENGAN KRIPTOGRAFI DAN STEGANOGRAFI

TUGAS AKHIR

Oleh

Nama : Gerry Junior Wibawa

No. Mahasiswa : 07523169

Telah dipertahankan di Depan Sidang Penguji Sebagai

Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Strata-1

Teknik Informatika

Yogyakarta, Januari 2012

Tim Penguji

Yudi Prayudi, S.Si, M.Kom _____

Ketua

Syarif Hidayat, S.Kom, MIT _____

Anggota I

Zainuddin Zukhri , ST, MIT _____

Anggota II

Mengetahui,

Ka. Prodi Teknik Informatika

Fakultas Teknologi Industri

Universitas Islam Indonesia

Yudi Prayudi, S.Si, M.Kom

HALAMAN PERSEMBAHAN

Karya ini saya persembahkan untuk kedua orangtua saya:

Ayahanda Misrun S.Sos, SH, M.Si

Ibunda Armianti SH

Dan ketiga adik saya:

Zulfikar Satya Nugraha

Rizky Nur Aishah Pratiwi

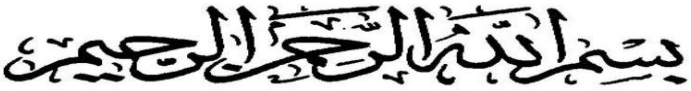
Muhammad Romi Fikri Imaduddin

-GerryJW-

MOTTO

- *Sekali melangkah, there is no turning back*
- *Nothing is true, Everything is Permitted (Assassin Creed)*
- *Segala sesuatu harus dilakukan dengan niat yang sungguh-sungguh dan tidak boleh setengah-setengah*
- *Don't Just See, Observe (Sherlock Holmes)*

KATA PENGANTAR



Assalamualaikum Wr, Wb

Puji dan syukur penulis panjatkan atas kehadirat Allah SWT, yang telah memberikan karunia dan hidayahnya sehingga laporan tugas akhir ini dapat terselesaikan dengan baik. Serta tidak lupa juga untuk memanjatkan shalawat serta salam kepada junjungan kita nabi muhhammad SAW yang menjadi suri tauladan bagi kita sampai akhir zaman.

Tugas akhir saya yang berjudul Anti Forensik dalam penyamaran file dengan Kriptografi dan Steganografi, digunakan untuk melindungi data milik pribadi agar tidak disalahgunakan oleh orang yang kurang bertanggung jawab.

Dalam kesempatan kali ini saya juga tidak lupa untuk berterima kasih kepada:

1. Allah SWT, atas karunia, hidayah, serta pertolongannya selama ini.
2. Bapak Ir. Gumbolo Hadi Susanto M.Sc, selaku dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
3. Bapak Yudi Prayudi, S.Si, M.Kom, selaku ketua jurusan Teknik Informatika dan juga selaku dosen pembimbing tugas akhir yang telah memberikan bimbingan dan pengarahan selama pelaksanaan tugas akhir saya ini.
4. Kedua orang tua saya yang sangat saya cintai, yaitu Ayahanda saya Misrun Nurdin S.sos, S.H, M.Si. dan ibunda saya Armianti S.H. serta ketiga adik saya, Zulfikar Satya Nugraha, Rizky Nur Aishah Pratiwi, Romi Fikri Imaduddin, dan seluruh keluarga yang telah memberikan semangat kepada saya.
5. Sahabat-sahabat saya yang selalu memberi motivasi pantang menyerah yaitu, Miemie, Onez, Doe, Yoedha, Djabul, Putra, Rangga.
6. Teman-teman INCLUDE 07 yang tidak bisa disebutkan satu per satu.

Penulis juga menyadari bahwa penyusunan laporan tugas akhir ini masih belum sempurna, karena keterbatasan dan pengalaman. Oleh karena itu penulis mengharapkan kritik dan saran yang membangun untuk membantu penulis di masa yang akan datang.

Akhir kata, penulis berharap agar laporan ini dapat bermanfaat bagi semua pihak, Amin.

Wassalamualaikum Wr, Wb.

Yogyakarta,

Penulis

TAKARIR

File Carrier	: File yang digunakan untuk membawa file steganografi
Encode	: Proses penyamaran atau penyandian suatu pesan
Decode	: Proses pengembalian hasil penyamaran dan penyandian
Interface	: Tampilan antar muka yang memudahkan user untuk berinteraksi secarang langsung dengan sistem
Stegofile	: File yang didalamnya mengandung steganografi
Extract	: Suatu proses untuk mengeluarkan suatu data atau file dari dalam file lainnya.
Steganalysis	: Ilmu yang mempelajari untuk mendeteksi steganografi
Stegotext	: Pesan yang berisi *embedded message*
Eavesdropper	: Orang yang mencoba untuk menangkap pesan selama pesan ditransmisikan
Non-repudiation	: Layanan untuk mencegah entitas yang berkomunikasi melalui penyangkalan
Authentication	: Layanan yang berhubungan dengan identifikasi baik mengidentifikasi kebenaran pihak-pihak yang berkomunikasi maupun mengidentifikasi kebenaran sumber pesan
Confidentiality	: Layanan yang ditujukan agar pesan tidak terbaca oleh pihak-pihak yang tidak berhak
Data integrity	: Layanan yang menjamin bahwa pesan masih asli atau belum pernah dimanipulasi selama pengiriman
Plaintext	: Sebuah pesan sebelum di ubah menjadi *ciphertext,* atau pesan yang belum dienkripsi
Ciphertext	: Sebuah pesan yang telah dienkripsi

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

ABSTRAKSI

Anti Forensik adalah ilmu tentang bagaimana cara untuk mengamankan suatu bukti digital, salah satu teknik anti forensik adalah kriptografi dan steganografi, yang dimana kedua teknik tersebut merupakan teknik yang sudah sangat populer dalam penyandian dan penyembunyian pesan. Dari hal tersebut dilakukan penelitian tentang bagaimana melakukan teknik penyamaran file dengan kriptografi dan steganografi dan teknik seperti apa yang dilakukan agar penyamaran file tersebut sulit untuk ditemukan.

Pengumpulan data yang dilakukan adalah dengan mengambil data yang relevan melalui studi literatur dan pustaka. Sebelum penyamaran file dilakukan, maka dibuat konsep penyamarannya dan penentuan kasusnya terlebih dahulu. Kasus penyamaran filenya yaitu terdapat 3 teknik penyamaran file, dari hasil penyamaran file yang dilakukan, di analisa teknik mana yang penyamarannya aman. Dan setelah dianalisa, diperoleh hasil bahwa terdapat 2 teknik yang penyamaran filenya aman yaitu, Teknik 2 dan Teknik 3. Dari kedua teknik tersebut kemudian digabungkan dan menghasilkan suatu kombinasi teknik penyamaran file yang lebih kompleks dan aman.

Kata kunci = Anti Forensik, Kriptografi, Steganografi

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Anti forensik yang merupakan *rival* dari komputer forensik adalah ilmu yang mempelajari cara untuk mengamankan suatu bukti digital agar sulit bahkan tidak bisa terdeteksi oleh para investigator komputer forensik. Salah satu teknik anti forensiknya adalah Kriptografi dan Steganografi. Kriptografi dan Steganografi merupakan teknik yang sangat populer dalam penyembunyian pesan, mulai dari jaman Julius cesar, kedua teknik tersebut sudah digunakan untuk penyamaran pesan agar tidak bisa terbaca dan terdeteksi oleh pihak lawan. Pada jaman modern sekarang ini steganografi dan kriptografi telah berkembang dan masuk ke dunia digital. Di dunia digital steganografi dan kriptografi digunakan untuk menyembunyikan dan menyamarkan data penting agar tidak bisa di akses oleh orang lain. Maka dari itulah steganografi dan kriptografi ini masuk dalam teknik Anti-Forensik.

Salah satu tindakan steganografi dalam dunia digital adalah seperti laporan yang disampaikan oleh BBC News bahwa pada tahun 2001 jaringan teroris dunia yaitu Al-Qaeda menggunakan steganografi untuk menyembunyikan pesan rahasia ke dalam gambar-gambar porno dan menyebarkannya lewat media internet yaitu web. Dari laporan tersebut dapat disimpulkan bahwa steganografi merupakan teknik yang sangat efektif dalam penyembunyian pesan.

Dari uraian tersebut, perlu dilakukan penelitian tentang bagaimana melakukan teknik penyamaran file yang aman menggunakan kriptografi dan steganografi.

1.2 Perumusan Masalah

Berdasarkan latar belakang diatas, maka perumusan masalahnya adalah:

1. Bagaimana melakukan teknik penyamaran file menggunakan kriptografi dan steganografi.
2. Teknik seperti apa yang dilakukan untuk membuat penyamaran suatu file sulit untuk dideteksi dan ditemukan.

1.3 Review Penelitian atau Kegiatan Sejenis.

Review beberapa penelitian sejenis:

a. Penelitian tentang Anti-Forensik yang dilakukan oleh Mahendra (2011) yang berjudul Teknik Anti Forensik untuk memanipulasi File. penelitian yang dia lakukan adalah bagaimana melakukan manipulasi suatu file dengan teknik-teknik anti forensik yaitu, dengan mengganti ekstensi file, melakukan perubahan pada tanggal suatu file (*time stamp)*, dan menghapus file dengan menggunakan *secure delete.* Kemudian dia melakukan pendeteksian terhadap file yang telah dimanipulasi, apakah file tersebut bisa terdeteksi oleh software forensik atau tidak.
b. Kemudian ada lagi penelitian tentang anti-forensik yang dilakukan oleh Garfinkel (2006) yang berjudul Anti-Forensics: Techniques, Detection, and Countermeasures. Penelitian yang dilakukan sebenarnya hampir sama dengan yang dilakukan oleh Mahendra (2011), hanya saja penelitian yang dilakukan adalah melakukan kegiatan anti-forensik dengan mengunakan tool anti-forensik untuk melakukan serangan terhadap tool-tool komputer forensik yaitu dengan cara mengeksploitasi bugs yang terdapat didalam tool-tool tersebut, dan kemudian untuk hasil akhirnya dilakukan pengevaluasian terhadap keefektifan tool yang digunakan untuk menyerang tool komputer forensik, strategi yang digunakan untuk pendeteksian, dan mendiskusikan tentang tindakan balasan yang digunakan.

1.4 Batasan Masalah

Agar dalam penelitian ini masalahnya tidak terlalu melebar, maka diberikan batasan-batasan untuk pendekatan permasalahan agar lebih terperinci dalam pelaksanaannya, yaitu:

1. Tidak membahas kriptanalisis.
2. *File* yang akan disamarkan adalah *file* Txt.
3. *File Carrier* yang digunakan berekstensi Jpeg, HTML, dan Flv.
4. Tidak melakukan *Unhiding* dan *decode* dari penyamaran file.

1.5 Tujuan dan Manfaat Penelitian

1.5.1 Tujuan

Tujuan dari penelitian ini adalah:

1. Untuk mengetahui bagaimana melakukan teknik penyamaran suatu file dengan menggunakan kriptografi dan steganografi.
2. Untuk mengetahui apa saja yang dilakukan dalam penyamaran file agar penyamaran suatu file sulit untuk ditemukan.
3. Untuk mengetahui penyamaran file yang dilakukan masih bisa terdeteksi atau tidak dengan menggunakan tool *steganalysis.*

1.5.2 Manfaat

Manfaat dari penelitian ini adalah:

1. Memahami cara kerja penyamaran file yang aman.
2. Meningkatkan pengetahuan tentang anti-forensik.
3. Dapat diterapkan untuk menyembunyikan dan mengamankan file penting.

1.6 Metodologi Penelitian

Metodologi Penelitian adalah Pembahasan tentang cara atau metode yang digunakan dalam penelitian, antara lain:

1.6.1 Studi dan Pengumpulan data

Pengumpulan data yang dilakukan didalam penelitian ini adalah dengan mengambil data atau informasi yang didapat dengan melakukan beberapa percobaan atau *experiment* penelitian terkait, selain itu pengumpulan data juga dilakukan dengan mencari informasi yang relevan yang terdapat pada berbagai literatur seperti jurnal, *website,* atau artikel yang terkait dengan penelitian yang dilakukan.

1.6.2 Kerangka Konseptual

Sebelum penelitian dilakukan maka akan dibuat kerangka konseptual terlebih dahulu yang dimana didalam kerangka tersebut akan dijelaskan tentang langkah-langkah penyamaran file yang akan dilakukan.

1.6.3 Analisa Data

Pada tahapan ini akan dilakukan analisis terhadap hasil dari penyamaran file yang dilakukan sehingga dari proses analisis yang dilakukan bisa diketahui benar atau tidaknya suatu hipotesis.

1.7 Sistematika Penulisan

BAB I PENDAHULUAN

Pada Bab ini akan dijelaskan tentang latar belakang pengambilan judul anti forensik dalam penyamaran dengan kriptografi dan steganografi, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II LANDASAN TEORI

Pada Bab ini akan dijelaskan tentang berbagai teori yang relevan yang berkaitan dengan penelitian anti forensik dalam penyamaran file dengan kriptografi dan steganografi, yaitu berupa tinjauan pustaka dan teori dasar.

BAB III METODOLOGI

Pada Bab ini akan dijelaskan tentang langkah-langkah perancangan dalam melakukan penelitian tentang anti forensik dalam penyamaran file dengan kriptografi dan steganografi.

BAB IV IMPLEMENTASI HASIL DAN PEMBAHASAN

Pada Bab ini akan dijelaskan tentang uji coba serta pembahasan dari anti forensik dalam penyamaran dengan kriptografi dan steganografi.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dan saran yang didapat dari penelitian tentang anti forensik dalam penyamaran file dengan kriptografi dan steganografi.

BAB II

LANDASAN TEORI

2.1 Komputer Forensik

Definisi komputer forensik adalah suatu proses mengidentifikasi, memelihara, menganalisa, dan mempergunakan bukti digital menurut hukum yang berlaku Budiman (2003). Sedangkan menurut Dr.HB Wolfre dalam makalah Eko Indrajit (2010), Komputer forensik adalah suatu rangkaian metodologi yang terdiri dari teknik dan prosedur untuk mengumpulkan bukti-bukti berbasis entitas maupun piranti digital agar dapat dipergunakan secara sah sebagai alat bukti di pengadilan.

2.1.1 Tujuan dan Fokus Komputer Forensik

Menurut Eko Indrajit (2010) Tujuan dan fokus komputer forensik adalah:

1. Untuk membantu memulihkan, menganalisa, dan merepresentasikan materi/entitas berbasis digital atau elektronik sedemikina rupa sehingga dapat dipergunakan sebagai alat bukti yang sah di pengadilan.
2. Untuk mendukung proses identifikasi alat bukti dalam waktu yang relatif, agar dapat diperhitungkan perkiraan potensi dampak yang ditimbulkan akibat perilaku jahat yang dilakukan oleh kriminal terhadap korbannya, sekaligus mengungkapkan alasan dan motivasi tindakan tersebut sambil mencari pihak-pihak yang terkait secara langsung maupun tidak langsung dengan perbuatan tidak menyenangkan dimaksud.

Sedangkan fokus data yang dikumpulkan dapat dikategorikan menjadi 3 domain utama, yaitu:

1. Active Data, yaitu informasi terbuka yang dapat dilihat oleh siapa saja, terutama data, program, maupun file yang dikendalikan oleh system informasi.

2. Archival Data, yaitu informasi yang telah menjadi arsip sehingga telah disimpan sebagai backup dalam berbagai bentuk alat penyimpanan, seperti harddisk eksternal. CD ROM, backup tape, DVD, dan lain-lain.
3. Latent Data, yaitu informasi yang membutuhkan alat khusus untuk mendapatkannya karena sifatnya yang khusus, misalnya telah dihapus, ditimpa data lain, rusak(corrupted file), dan lain sebagainya.

2.2. Anti Forensik

Didalam makalah Kessler (2006), Roger seorang praktisi digital forensik dan juga seorang investigator mendefinisikan anti-forensik sebagai sebuah percobaan atau usaha negatif untuk mempengaruhi keberadaan, kuantitas, dan kualitas dari barang bukti yang didapat dari tempat kejadian perkara, atau membuat agar pemeriksaan terhadap barang bukti sulit dilakukan. Sedangkan menurut Vincent Liu yang terdapat di dalam buku Arryawan (2010) mendefinisikan anti-forensik sebagai metode sains yang digunakan terhadap media digital untuk membuat suatu barang bukti atau informasi menjadi tidak berlaku untuk di review di pengadilan.

Jadi inti dari definisi anti-forensik adalah suatu usaha atau proses untuk mengamankan suatu data agar tidak bisa diakses selain oleh orang pemilik data tersebut.

2.2.1 Tujuan Anti Forensik

Menurut Arryawan (2010) Tujuan Anti-Forensik, yaitu:

1. Bagaimana membuat supaya data tidak bisa ditemukan atau dibuka, misalnya dengan disembunyikan (hidden), disandikan atau dienkripsi, steganografi, secure delete, dan sebagainya.
2. Bagaimana mengupayakan agar andaikata suatu data berhasil ditemukan, maka data tersebut tetap tidak layak sesuai dengan standar hukum, mungkin karena

integritasnya sudah rusak dan meragukan, misalnya saja dengan mengubah tanggal dan sebagainya.

Selanjutnya Arryawan (2010) mengemukakan Perbandingan komputer forensik dan anti-forensik adalah sebagai berikut:

Tabel 2.1 Perbandingan komputer forensik dan anti-forensik

Komputer Forensik	**Anti-Forensik**
Menjebol proteksi	Mengamankan proteksi
Memulihkan data yang dihapus	Memastikan data yang sudah dihapus tidak disalahgunakan dan tidak bisa ditemukan oleh pihak lain
Mengakses data	Melindungi data
Membuka penyamaran data	Menyamarkan data
Mencari jejak	Menghapus semua jejak yang ada
Melacak kejahatan	Melindungi data privasi

2.2.2 Prinsip-prinsip Anti-Forensik

Berikut ini adalah sejumlah prinsip anti forensik:

1. Mencegah keberadaan data dianggap lebih ideal ketimbang menghapusnya
2. Memakai program yang bersifat *portable* dianggap lebih ideal dibandingkan dengan menggunakan program yang perlu diinstall, karena program yang perlu diinstall seringkali meninggalkan jejak bahkan setelah di uninstall, misalnya di folder *Windows*, di *Registry*, dan sebagainya.
3. Memakai program yang umum dipakai masyarakat lain dianggap lebih ideal jika ada *software* bagus dan umum, yang bisa dipakai untuk tujuan anti forensik. *software* yang standar tentu lebih tidak menarik perhatian.

2.2.3 Kategori Metode Anti Forensik

Menurut Roger dalam makalah Kessler (2006), anti-forensik dapat dikategorikan menjadi 4 kategori, yaitu:

1. Data Hiding
2. Artefact Wiping
3. Trail Obfuscation
4. Attack against the computer forensics process or tool.

2.3 Kriptografi

Berdasarkan Munir (2006) Kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan atau data. Sedangkan berdasarkan Ganley (2004), Kriptografi adalah suatu ilmu sains yang menggunakan matematika untuk meng-enkripsi dan men-dekripsi suatu data. Kriptografi dapat digunakan untuk mengirimkan suatu informasi melewati insecure network atau jaringan yang tidak aman, agar tidak dibaca oleh orang lain selain penerima informasi tersebut.

2.3.1 Terminologi Kriptografi

Kemudian Munir (2006) mengemukakan bahwa Didalam kriptografi sering ditemukan berbagai istilah atau terminologi. Beberapa istilah yang penting untuk diketahui adalah seperti dibawah ini:

a. Pesan, *Plaintext*, dan *Ciphertext*

   Pesan (*Message)* adalah data atau informasi yang dapat dibaca dan dimengerti maknanya. Nama lain untuk pesan adalah *Plaintext* atau teks jelas (*Cleartext).* Pesan dapat berupa data atau informasi yang dikirim melalui kurir, saluran telekomunikasi, dsb, atau yang disimpan didalam media perekaman seperti kertas,

storage, dsb. Pesan yang tersimpan tidak hanya berupa teks, tetapi juga dapat berbentuk citra (*image),* suara/bunyi (audio*),* dan video.

Agar pesan tidak dapat dimengerti oleh pihak lain, maka pesan perlu disandikan ke bentuk lain yang tidak dapat dipahami. Bentuk pesan yang tersandi disebut dengan *ciphertext* atau *cryptogram.*

b. 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. Entitas disini dapat berupa orang, mesin (komputer), kartu kredit, dan sebagainya.

c. Enkripsi dan Dekripsi

Proses menyandikan *plaintext* menjadi *ciphertext* disebut dengan enkripsi (*encryption)* atau *enciphering.* Sedangkan proses untuk mengembalikan *ciphertext* menjadi *plaintext* semula dinamakan dekripsi atau *deciphering.*

Gambar 2.1 Proses enkripsi dan dekripsi

Sumber: (Ganley, 2004)

d. Cipher dan Kunci

Algoritma kriptografi disebut juga *cipher* atau aturan untuk *enciphering* dan *deciphering,* atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi.

e. Sistem Kriptografi

Kriptografi membentuk sebuah sistem yang dinamakan sistem kriptografi. Sistem kriptografi (*cryptosystem)* adalah kumpulan yang terdiri dari algoritma kriptografi, semua *plaintext* dan *ciphertext* yang mungkin, dan kunci.di dalam sistem kriptografi, *cipher* hanyalah salah satu komponen saja.

f. Penyadap

Penyadap (*eavesdropper)* adalah orang yang mencoba untuk menangkap pesan selama pesan ditransmisikan. Tujuan penyadap ini adalah untuk mendapatkan informasi sebanyak-banyaknya mengenai sistem kriptografi yang digunakan untuk berkomunikasi dengan maksud untuk memecahkan *ciphertext.* Nama lain dari penyadap adalah: *enemy, adversary, intruder, interceptor, bad guy.*

g. Kriptanalisis dan Kriptologi

Kriptografi berkembang sedemikian rupa sehingga melahirkan bidang yang berlawanan yaitu kriptanalisis. Kriptanalisis (*cryptanalysis)* adalah ilmu dan seni untuk memecahkan *ciphertext* menjadi *plaintext* tanpa mengetahui kunci yang digunakan. Pelakunya disebut kriptanalis.

2.3.2 Tujuan Kriptografi

Berdasarkan Munir (2006) Tujuan Kriptografi adalah:

a. Kerahasiaan (*confidentiality),* adalah layanan yang ditujukan untuk menjaga agar pesan tidak terbaca oleh pihak-pihak yang tidak berhak.
b. Integritas data (*data integrity),* adalah layanan yang menjamin bahwa pesan masih asli/utuh atau belum pernah dimanipulasi selama pengiriman.
c. Otentikasi (*authentication),* adalah layanan yang berhubungan dengan identifikasi baik mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (*user authentication* atau *entity authentication)* maupun mengidentifikasi kebenaran sumber pesan (*data origin authentication).*

d. Nirpenyangkalan (*non-repudiation),* adalah layanan untuk mencegah entitas yang berkomunikasi melalui penyangkalan, yaitu pengirim pesan menyangkal melakukan pengiriman atau penerima pesan menyangkal telah menerima pesan.

2.3.3 Advanced Encryption Standard

Berdasarkan Ariyana (2004) *Advanced Encryption Standard* (AES) merupakan algoritma cryptographic yang dapat digunakan untuk mengamankan data. Algoritma AES adalah blok chipertext simetrik yang dapat meng-enkripsi (*enchiper*) dan dekripsi (*dechiper*) informasi. Standar yang digunakan pada AES adalah menggunakan algoritma *rijndael.* Algoritma AES ini menggunakan kunci kriptografi 128 bits, 192 bits, dan 256 bits untuk meng-enkripsi dan dekripsi pada blok 128 bits.

2.3.4 Algoritma Rijndael

Berdasarkan Rahayu (2005) Algoritma *Rijndael* yang dikembangkan oleh joan daemen dan vincent rijmen dipilih sebagai standar dalam *advanced encryption standard.*

Berdasarkan Surian (2006) *Rijndael* termasuk dalam jenis algoritma kriptografi yang sifatnya simetri dan *cipher block.* Dengan demikian algoritma ini menggunakan kunci yang sama saat enkripsi dan dekripsi serta masukan dan keluarannya berupa blok dengan jumlah bit tertentu.

Rijndael mendukung berbagai variasi ukuran blok dan kunci yang akan digunakan. namun *rijndael* tetap mempunyai ukuran blok dan kunci yang tetap sebesar 128, 192, 256 bit. Pemilihan ukuran blok data dan kunci akan menentukan jumlah proses yang harus dilalui untuk proses enkripsi dan dekripsi. Berikut adalah perbandingan jumlah proses yang harus dilalui untuk masing-masing masukan.

Tabel 2.2 Jumlah proses berdasarkan bit blok dan kunci

Panjang kunci	Ukuran blok data	Jumlah proses
(Nk)	(Nb)	(Nr)
4	4	10
6	4	12
8	4	14

Blok-blok data masukan dan kunci dioperasikan dalam bentuk *array.* Setiap anggota *array* sebelum menghasilkan keluaran *ciphertext* dinamakan *state.* Setiap *state* akan mengalami proses yang secara garis besar terdiri dari 4 tahap yaitu, *AddRoundKey, SubBytes, ShiftRows,* dan *MixColumns.* Kecuali tahap *mixcolumns,* ketiga tahap lainnya akan diulang pada setiap proses, sedangkan pada tahap *mixcolumns* tidak akan dilakukan pada tahap terakhir. Karena terjadi beberapa tahap dalam proses enkripsi, maka diperlukan *subkey-subkey* yang akan dipakai pada tiap tahap. Pengembangan jumlah kunci yang akan dipakai diperlukan karena kebutuhan *subkey-subkey* yang akan dipakai dapat mencapai ribuan bit, sedangkan kunci yang disediakan secara *default* hanya 128-256 bit. Jumlah total kunci yang diperlukan sebagai *subkey* adalah sebanyak Nb (Nr + 1), dimana Nb adalah besarnya blok data dalam satuan *word.* Sedangkan Nr adalah jumlah data yang harus dilalui dalam satuan *word.* Sebagai contoh, bilaman digunakan 128 bit (4 *word*) blok data dan 128 bit (4 *word*) kunci maka akan 10 kali proses. Dengan demikian dari rumus didapatkan 4 (10 + 1)= 44 word=1408 bit kunci. Untuk melakukan pengembangan jumlah kunci yang akan dipakai dari kunci utama maka dilakukan *key schedule.*

1. *Key Schedule*

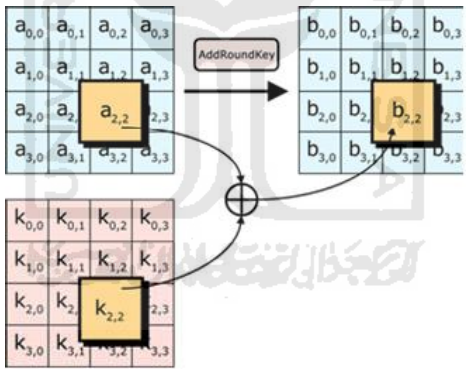
   Proses *key schedule* diperlukan untuk mendapatkan *subkey-subkey* dari kunci utama agar cukup untuk melakukan enkripsi dan dekripsi. Proses ini terdiri dari beberapa operasi, yaitu:

   a. Operasi *Rotate,* yaitu operasi perputaran 8 bit pada 32 bit dari kunci.

b. Operasi *SubBytes*, pada operasi ini 8 bit dari *subkey* di substitusikan dengan nilai dari S-Box.
c. Operasi *Rcon*, operasi ini dapat diterjemahkan sebagai operasi pangkat 2 nilai tertentu dari *user*.
d. Operasi XOR dengan w[i-Nk] yaitu word yang berada pada Nk sebelumnya.

2. *AddRoundKey*

Pada proses ini *subkey* digabungkan dengan *state*. Proses penggabungan ini menggunakan operasi XOR untuk setiap *byte* dari *subkey* dengan *byte* yang bersangkutan dari *state*. Untuk setiap tahap, *subkey* dibangkitkan dari kunci utama dengan menggunakan proses *keyschedule*. Setiap *subkey* berukuran sama dengan *state* yang bersangkutan. Proses *AddRoundKey* bisa dilihat pada gambar berikut.



Gambar 2.2 Proses *addroundkey*

Sumber: (Surian, 2006)

3. *Subbytes*

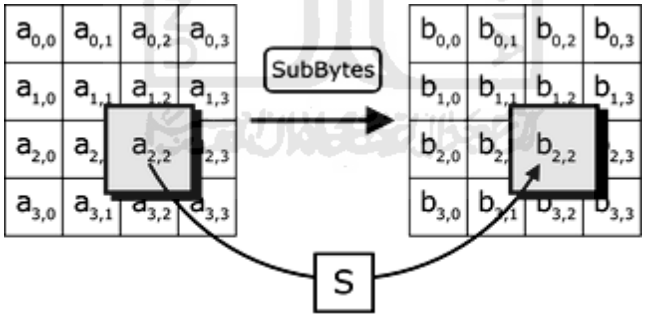
Proses *subbytes* adalah operasi yang akan melakukan substitusi tidak linear dengan cara mengganti setiap *byte state* dengan *byte* pada sebuah tabel yang dinamakan tabel S-Box. Sebuah tabel S-Box terdiri dari 16 x 16 baris dan kolom

dengan masing-masing berukuran 1 *byte.* berikut adalah gambar tabel S-Box dan proses *subbytes.*

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d0	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Gambar 2.3 S-Box

Sumber: (Surian, 2006)



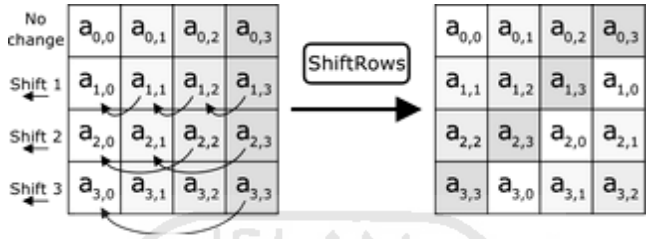
Gambar 2.4 Proses *subbytes*

Sumber: (Surian, 2006)

4. *Shift Rows*

Proses *shift rows* akan beroperasi pada tiap baris dari tabel *state.* Proses ini akan bekerja dengan cara memutar *byte-byte* pada 3 baris terakhir (baris 1, 2, dan 3)

dengan jumlah perputaran yang berbeda-beda. Baris 1 akan diputar sebanyak 1 kali, baris 2 akan diputar sebanyak 2 kali, dan baris 3 akan diputar sebanyak 3 kali. Sedangkan baris 0 tidak akan diputar. Proses *shift rows* bisa dilihat pada gambar berikut.



Gambar 2.5 Proses *shift rows*

Sumber: (Surian, 2006)

5. *MixColumns*

Proses *mixcolumn* akan beroperasi pada tiap kolom dari tabel *state.* Operasi ini menggabungkan 4 *bytes* dari setiap kolom tabel *state* dan menggunakan transformasi linear. Operasi *mixcolumns* memperlakukan setiap kolom sebagai polinomial 4 suku dalam *galois field* dan kemudian dikalikan c(x) modulo (x4+1), dimana $c(x)=3x^3+x^2+x+2$. Kebalikan dari polinomial ini adalah $c(x)=11x^3+13x^2+9x+14$. Operasi *mixcolumn* juga dapat dipandang sebagai perkalian *matrix*. Langkah *mixcolumn* dapat ditunjukkan dengan mengalikan 4 bilangan didalam *galois field* oleh *matrix* berikut ini.

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

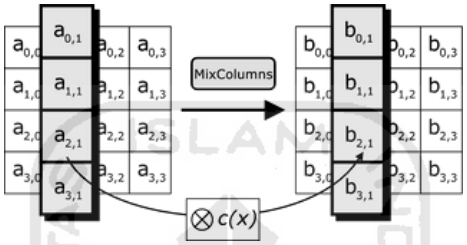
Gambar 2.6 *matrix*

Sumber: (Surian, 2006)

Atau bila dijabarkan:

r0=2a0+a3+a2+3a1
r1=2a1+a0+a3+3a2
r2=2a2+a1+a0+3a3
a3=2a3+a2+a1+3a0

Operasi penjumlahan diatas menggunakan operasi XOR, sedangkan operasi perkalian dilakukan dalam *galois field.*



Gambar 2.7 Proses *mixcolumns*

Sumber: (Surian, 2006)

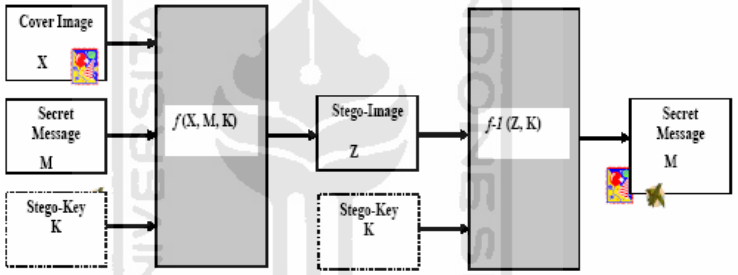
2.3.5 Fungsi Hash dan MD5

Berdasarkan Rahayu (2005) Fungsi hash adalah fungsi yang memproduksi output dengan panjang tetap dari input yang berukuran variabel. Output dari fungsi hash disebut sebagai fungsi *digest.* Fungsi hash memiliki karakteristik fungsi satu arah karena file asli tidak dapat dibuat dari *message digest.* Salah satu fungsi hash adalah MD5, MD5 adalah algoritma *message digest* yang dikembangkan oleh Ronald Rivest pada tahun 1991. MD5 mengambil pesan dengan panjang sembarang dan menghasilkan *message digest* 128 bit. Pada MD5 pesan diproses dalam blok 512 bit dengan empat round berbeda.

2.4 Steganografi

Menurut Arryawan (2010) Steganografi adalah seni menyamarkan data. Steganografi berasal dari kata yunani stagonos, yang berarti tertutup atau tercover, dan grafi yang artinya tulisan. Jadi steganografi adalah seni atau ilmu untuk menyamarkan sebuah pesan atau data rahasia di dalam data atau media yang tampaknya biasa saja sehingga keberadaan pesan rahasia itu sulit untuk diketahui,

Definisi steganografi menurut Krenn (2004) adalah seni menyembunyikan informasi didalam informasi. Sedangkan menurut Minhajuddin (2011) steganografi adalah seni atau teknik untuk menyembunyikan informasi sedemikian rupa sehingga tidak ada orang lain selain penerima yang tau akan keberadaan informasi tersebut.



Gambar 2.8 Model sistem steganografi modern

Sumber: (Vembrina, 2004)

Tujuan dari steganografi adalah merahasiakan dan menyembunyikan keberadaan dari sebuah pesan tersembunyi atau sebuah informasi.

2.4.1 Terminologi Steganografi

Menurut Munir (2006) Terdapat beberapa istilah yang berkaitan dengan steganografi:

1. *Hiddentext* atau *Embedded message*, yaitu pesan yang disembunyikan.
2. *Covertext* atau *cover-object* atau *file carrier*, yaitu pesan yang digunakan untuk menyembunyikan *embedded message.*

3. *Stegotext* atau *steg-object* atau *stegofile*, yaitu pesan yang berisi *embedded message*.

2.4.2 Teknik Steganografi Modern

Ada beberapa teknik steganografi modern yang sangat popular digunakan, yaitu:

a. LSB atau Least Significant Bit

Menurut Munir (2006) Metode LSB merupakan metode steganografi yang paling sederhana dan paling mudah diimplementasikan. Untuk menjelaskan metode ini kita menggunakan citra digital sebagai *covertext*. Setiap *pixel* di dalam citra berukuran 1 sampai dengan 3 *byte*. pada susunan bit di dalam sebuah *byte* (1 *byte* = 8 bit), ada bit yang paling berarti (*most significant bit)* dan bit yang kurang berarti (*least significant bit).* Misalkan pada *byte* 11010010. But 1 yang pertama (digarisbawahi) adalah bit *most significant bit.* Dan bit 0 yang terakhir (digarisbawahi) adalah bit *least significant bit.* Bit yang cocok untuk diganti dengan bit pesan adalah bit LSB, sebab modifikasi hanya mengubah nilai *byte* tersebut satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya. Misalkan *byte* tersebut didalam gambar memberikan persepsi warna merah, maka perubaha satu bit LSB hanya mengubah persepsi warna merah menjadi tidak terlalu berarti. Mata manusia tidak dapat membedakan perubahan yang kecil ini. Contohnya huruf A dapat disisipkan dalam 3 pixel, misalnya data raster original adalah sebagai berikut:

(00100111	11101001	11001000)
(00100111	11001000	11101001)
(11001000	00100111	11101001)

Sedangkan representasi biner huruf A adalah 10000011. Dengan menyisipkan-nya pada data *pixel* diatas maka akan dihasilkan:

(00100111	1110100**0**	11001000)
(0010011**0**	11001000	1110100**0**)
(1100100**1**	00100111	11101001)

Terlihat hanya empat bit rendah yang berubah, untuk mata manusia maka tidak akan tampak perubahannya.

b. End Of File

Berdasarkan Utami (2007) Teknik ini menggunakan cara dengan menyisipkan data pada akhir file. teknik ini dapat digunakan untuk menyisipkan data yang ukurannya sesuai dengan kebutuhan. Ukuran file yang telah disisipkan data sama dengan ukuran file sebelum disisipkan data ditambah dengan ukuran data yang disisipkan ke dalam file tersebut. Dan berdasarkan Krisnawati (2008) contohnya adalah seperti berikut. Pada sebuah citra grayscale disisipkan pesan yang berbunyi "aku". Kode ASCII dari pesan tersebut adalah 97 107 117.

Misalkan matrik tingkat derajat keabuan citra sebagai berikut.

196	10	97	182	101	40
67	200	100	50	90	50
25	250	45	200	75	28
176	56	77	100	25	200
101	34	250	40	100	60
44	66	99	125	190	200

Kode biner pesan disisipkan pada akhir citra sehingga citra menjadi.

196	10	97	182	101	40
67	200	100	50	90	50
25	250	45	200	75	28
176	56	77	100	25	200
101	34	250	40	100	60
44	66	99	125	190	200

97 107 117

2.4.3 Steganalysis

Menurut Raggo (2004) Definisi *Steganalaysis* adalah proses untuk mengidentifikasi keberadaan dari pesan tersembunyi, dan juga proses untuk mengidentifikasi *tool* apa yang digunakan untuk menyembunyikan pesan tersebut. jika *tool* yang digunakan untuk menyembunyikan pesan tersembunyi tersebut bisa diketahui, *tool* tersebut bisa digunakan untuk meng-*extract* pesan yang disembunyikan.

2.5 File

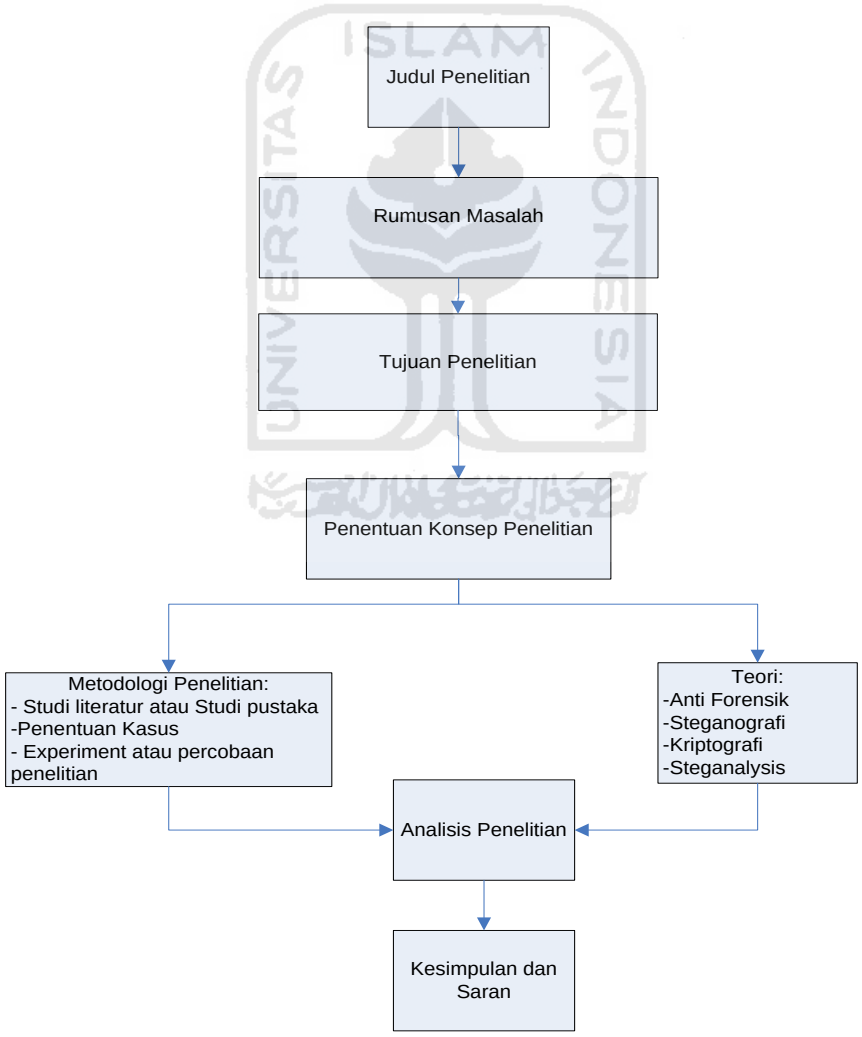
Menurut Utami (2007) File atau berkas adalah sekumpulan data (informasi) yang berhubungan yang diberi nama dan tersimpan didalam media penyimpanan sekunder (*secondary storage).* File memiliki ekstensi. Ekstensi berkas merupakan penandaan jenis berkas lewat nama berkas. Ekstensi biasanya ditulis setelah nama berkas dipisahkan oleh sebuah titik. Pada sistem yang lama (MS-DOS), ekstensi hanya diperbolehkan maksimal 3 huruf, contohnya: exe, bat, com, txt. Batasan tersebut dihilangkan pada sistem yang lebih baru (Windows), contohnya: mpeg, java. Pada UNIX bahkan dikenal ada file yang memiliki lebih dari satu ekstensi, contohnya: Taz.gz.

BAB III

METODOLOGI

3.1 Kerangka Berfikir

Kerangka berfikir merupakan miniatur keseluruhan dari semua proses penelitian dari awal sampai akhir. Berikut merupakan susunan kerangka berfikir tentang anti forensik dalam penyamaran file dengan kriptografi dan steganografi.

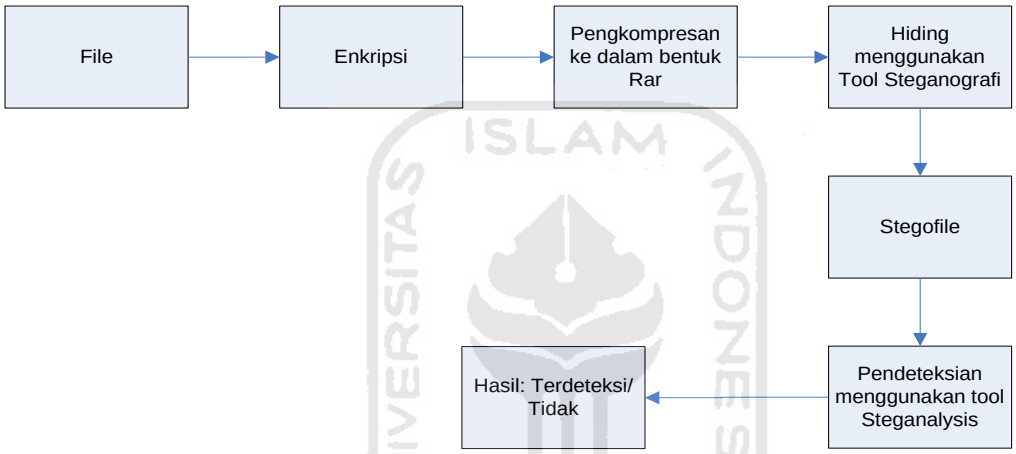


Gambar 3.1 Kerangka berfikir

3.2 Penentuan Konsep dan Hipotesis Penelitian

3.2.1 Penentuan Konsep

Sebelum penelitian ini dilakukan, maka akan dibuat konsep penyamaran file terlebih dahulu. Konsep disini merupakan tahap-tahap dari teknik penyamaran file yang akan dilakukan. Berikut konsep penyamaran file yang dilakukan.



Gambar 3.2 Konsep penyamaran file

Berikut penjelasan terhadap konsep penyamaran file yang dilakukan:

1. Pertama terdapat sebuah file yang akan disembunyikan, dalam hal ini file yang akan disembunyikan berupa file txt yang didalamnya terdapat sebuah pesan rahasia.

2. Langkah yang ke-2 adalah melakukan enkripsi terhadap file txt.
3. Langkah ke-3 adalah melakukan pengkompresan terhadap file yang sudah di enkripsi tadi.
4. Langkah ke-4, file yang hasil kompresan tadi di *hiding* atau disamarkan menggunakan tool steganografi. Hasil dari penyamaran yang dilakukan tool steganografi tersebut menghasilkan *stegofile*.
5. Langkah terakhir adalah melakukan pendeteksian menggunakan tool *steganalysis* terhadap *stegofile* tersebut, apakah steganografi yang terdapat pada *stegofile* tersebut bisa terdeteksi atau tidak.

3.3 Penentuan Kasus

Kasus penyamaran file yang akan dilakukan yaitu, terdapat 3 teknik penyamaran yang akan digunakan. konsep yang dilakukan pada ketiga teknik tersebut sama antara satu sama lain. Yang membedakannya adalah tool dan *file carrier* yang digunakan. bisa dilihat pada tabel berikut.

Tabel 3.1 Teknik penyamaran file

	Tool	File Carrier yang digunakan
Teknik 1	OurSecret	JPG
Teknik 2	WbStego	HTML
Teknik 3	OpenPuff v.3.40	JPG, FLV

Tujuan dari dilakukannya 3 teknik penyamaran file ini adalah untuk mencari dari ketiga teknik tersebut mana penyamaran file yang bagus dan aman.

3.4 Tool yang digunakan

3.4.1 Tool Penyamaran File yang digunakan

1. Notepad : File yang akan disamarkan
2. BCTextEncoder by Jetico Inc : Melakukan enkripsi teks.
3. Winrar : Melakukan Pengkompresan
4. OurSecret by SecureKit.net : Untuk melakukan *Hiding File*
5. WbStego4 : Untuk melakukan *Hiding File*
6. OpenPuff v.3.40 by Cosimo Oliboni : Untuk melakukan *Hiding File*

3.4.2 Tool Steganalysis yang digunakan

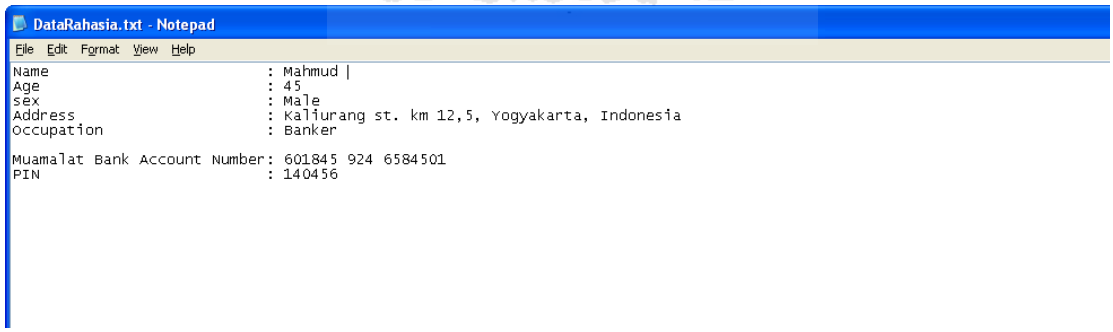
1. XSteg
2. StegSecret

BAB IV

IMPLEMENTASI HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan tentang teknik penyamaran file yang akan dilakukan dengan menggunakan kriptografi dan steganografi. Teknik penyamaran yang dilakukan berdasarkan pada konsep penyamaran file yang terdapat pada bab 3. Dijelaskan disini terdapat 3 teknik penyamaran file yang dilakukan, yang dimana pada setiap teknik penyamaran tersebut menggunakan tool dan *file carrier* yang berbeda. Dan untuk teknik *steganalysis* yang digunakan adalah dengan melakukan pendeteksian menggunakan tool *steganalysis.* Fungsi dari dilakukannya pendeteksian tersebut adalah untuk mengetahui apakah steganografi yang terdapat pada *stegofile* bisa terdeteksi atau tidak.

Tahap pertama sebelum melakukan teknik penyamaran file adalah dengan menyediakan file yang akan disamarkan atau file yang akan di *hiding.* File yang akan disamarkan disini adalah file Txt. Dicontohkan disini file yang akan disamarkan adalah file DataRahasia.txt. Didalam file DataRahasia.txt tersebut terdapat identitas rahasia berupa identitas seseorang dan nomor rekening beserta pin atm nya.



Gambar 4.1 DataRahasia.txt

Agar identitas tersebut tidak gampang untuk dibaca, maka identitas rahasia tersebut akan di enkripsi menggunakan tool BCTextEncoder, tool ini berfungsi untuk meng-enkripsi teks. BCTextEncoder ini menggunakan *advanced encryption standard*

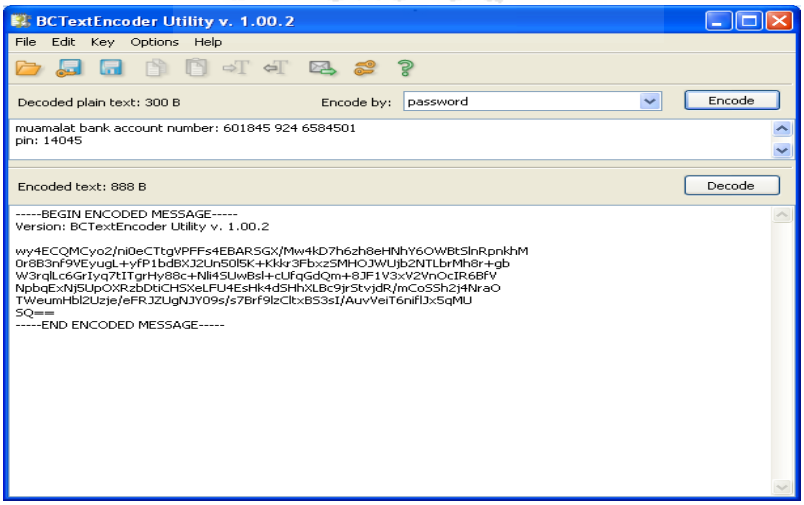
256 bit (AES-256) dalam melakukan enkripsi teks, yang dimana *advanced encryption standard* ini menggunakan algoritma *Rijndael* sebagai standar dalam pengenkripsian. Untuk lebih jelas mengenai algoritma *Rijndael* bisa dilihat pada bab landasan teori.

Langkah-langkah pengenkripsiannya adalah sebagai berikut:

1. *Copy Paste* isi dari file DataRahasia.txt ke dalam tool BCTextEncoder,
2. Kemudian lakukan enkripsi dengan mengklik tombol *encode.*
3. Setelah tombol *encode* diklik maka akan muncul *form* pengisian *password*. *password* yang digunakan menggunakan AES-256. Jika *password* telah di isi, klik OK.

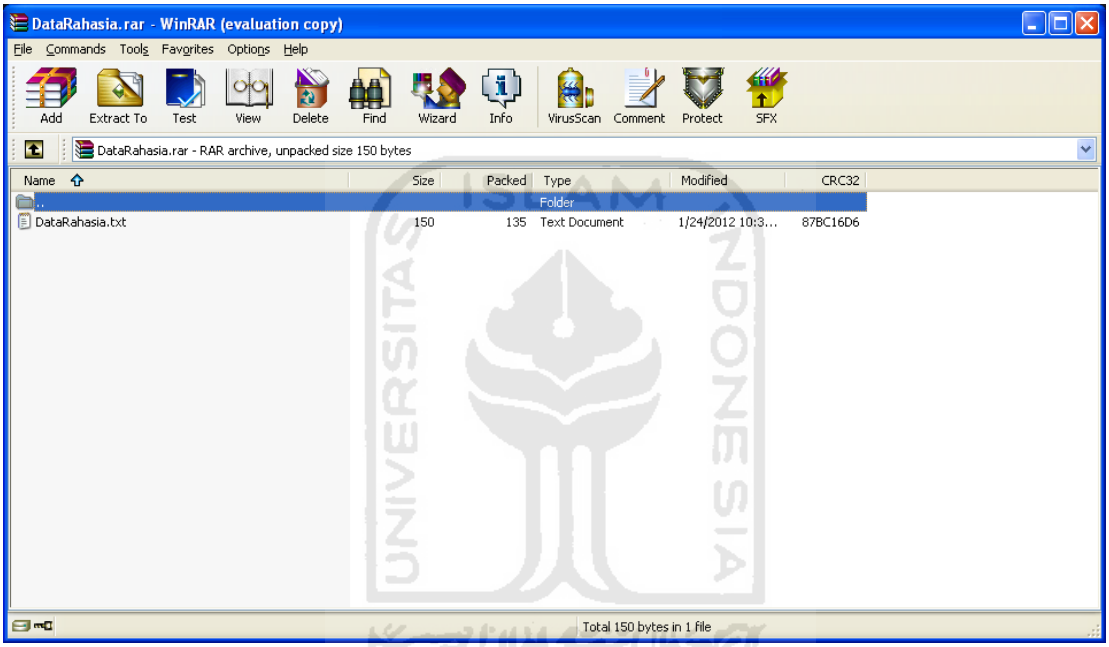
Gambar 4.2 *Password* AES 256

4. Setelah pengisian *password* selesai, maka akan keluar hasil seperti berikut. Isi dari pesan rahasia tadi sudah berupa *ciphertext.*



Gambar 4.3 Hasil enkripsi teks

Setelah isi file DataRahasia.txt tersebut sudah manjadi *ciphertext*, *copy paste*-kan *ciphertext* tersebut kedalam file DataRahasia.txt. lalu di *save.* Setelah itu, agar *ciphertext* yang terdapat didalam file DataRahasia.txt tersebut tidak rusak ketika dilakukan penyamaran menggunakan tool yang didalamnya juga terdapat kriptografinya, maka file DataRahasia.txt tersebut akan di kompres ke dalam bentuk Rar. Setelah di kompress maka menghasilkan file DataRahasia.rar



Gambar 4.4 DataRahasia.rar

Setelah pengekripisian file selesai, langkah selanjutnya adalah melakukan penyamaran file. Jadi file yang akan disamarkan nanti adalah file DataRahasia.txt yang telah di enkripsi dan di kompress ke dalam bentuk rar dan menjadi file DataRahasia.rar.

4.1 Penyamaran File

Penyamaran File adalah suatu proses untuk menyembunyikan suatu file baik itu dengan melakukan manipulasi atau dengan cara menyisipkannya kedalam file

lain. Fungsi dari penyamaran file adalah untuk mengamankan suatu data atau informasi agar tidak ditemukan oleh orang yang tidak berkepentingan.

4.2 Teknik 1

Teknik penyamaran file pertama adalah dengan melakukan penyamaran file DataRahasia.rar dengan menggunakan tool OurSecret. Terdapat 3 step penyamaran pada tool ini yaitu, step pertama adalah penginputan *file carrier* yang akan digunakan, step kedua adalah untuk penginputan file yang akan disamarkan, dan step yang ketiga adalah penginputan *password*. *Password* pada tool ini menggunakan fungsi hash MD5. Untuk lebih jelas tentang fungsi hash MD5 bisa dilihat pada bab landasan teori. Berikut tampilan tool OurSecret.

Gambar 4.5 Tool OurSecret

Dan untuk file yang akan digunakan sebagai *file carrier* adalah file *image*, yaitu file Gambar.jpg. Gambarnya adalah sebagai berikut.

Gambar 4.6 File Gambar.jpg

Langkah-langkah penyamaran menggunakan tool OurSecret terbagi menjadi 3 step, yaitu:

1. Pada Step 1 inputkan file yang akan dijadikan sebagai *file carrier*, disini file yang akan digunakan adalah file Gambar.jpg.
2. Pada Step 2 inputkan file yang akan disamarkan atau di *hiding*, disini di inputkan file DataRahasia.rar.
3. Pada Step 3, isikan password pada kolom password, kemudian isikan *password* yang sama pada kolom *confirm*.
4. Setelah semuanya sudah dilakukan, klik tombol *hide* untuk melakukan penyamaran.

Hasil dari penyamaran menggunakan tool steganografi akan menghasilkan file yang telah tersteganografi atau yang biasa disebut sebagai *stegofile*. Hasil dari penyamaran tersebut adalah seperti berikut.

Gambar 4.7 *Stegofile* Gambar.jpg

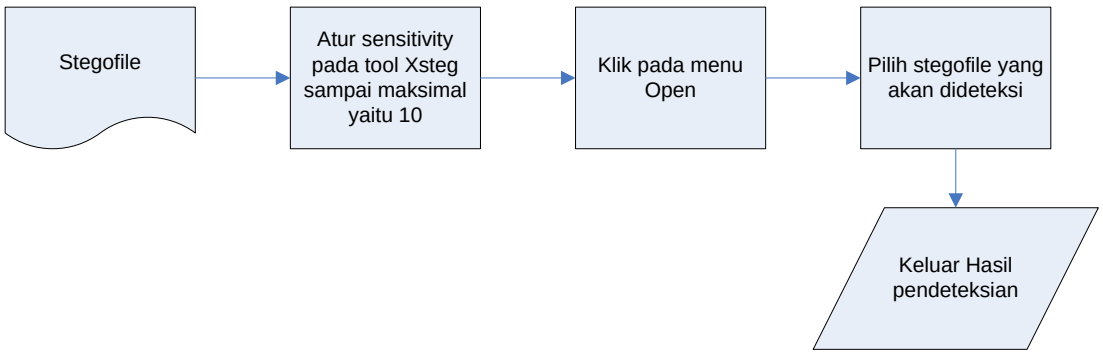
Dari penyamaran yang dilakukan tidak terdapat perbedaan antara file Gambar.jpg sebelum di steganografi dengan file Gambar.jpg setelah di steganografi. Setelah penyamaran file sudah selesai dilakukan, langkah terakhir yang akan dilakukan adalah melakukan pendeteksian terhadap file yang sudah di *hiding.*

4.2.1 Pendeteksian dengan XSteg dan StegSecret

Pendeteksian yang akan dilakukan adalah dengan menggunakan tool *steganalysis*, tool *steganalysis* ini digunakan dalam investigasi komputer forensik untuk mengetahui apakah suatu file mengandung steganografi atau tidak. tool yang akan digunakan disini ada 2, yaitu tool XSteg dan tool StegSecret.

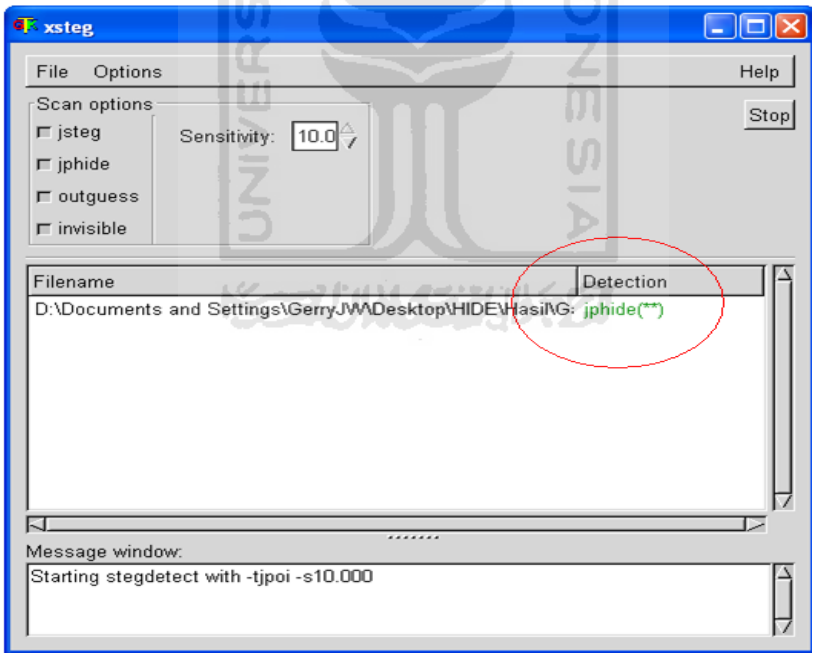
Tool pertama yang akan digunakan untuk mendeteksi adalah tool XSteg, tool XSteg ini dapat di unduh secara bebas di internet.

Langkah-langkah pendeteksian yang dilakukan pada tool ini adalah seperti gambar berikut.



Gambar 4.8 *Flowchart XSteg*

Setelah langkah pendeteksian telah dilakukan diperoleh hasil. Hasilnya bisa dilihat pada gambar dibawah yang keluar tulisan "jphide" berwarna hijau yang di tandai dengan lingkar merah yang menandakan bahwa terdeteksi steganografi pada file Gambar.jpg.



Gambar 4.9 Hasil pendeteksian menggunakan XSteg

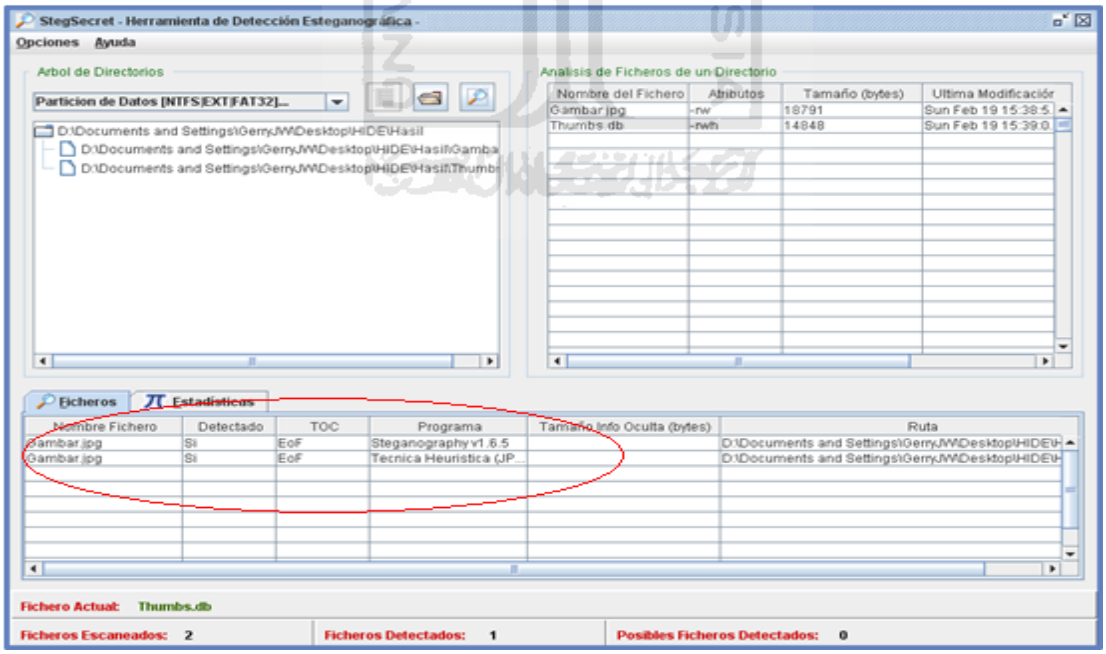
Kemudian langsung ke pendeteksian yang kedua, yaitu pendeteksian menggunakan tool StegSecret. StegSecret merupakan salah satu tool *steganalysis.*

Cara kerja StegSecret ini yaitu dengan melakukan *scanning* atau pencarian steganografi pada semua file yang terdapat didalam satu folder.

Langkah-langkah yang dilakukan adalah seperti gambar berikut.

Gambar 4.10 *Flowchart* StegSecret

Setelah langkah-langkah tersebut dilakukan, maka hasil pendeteksian pun keluar. Hasilnya adalah StegSecret berhasil mendeteksi steganografi pada *stegofile* Gambar.jpg.



Gambar 4.11 Hasil pendeteksian StegSecret

Bisa dilihat pada gambar di atas bahwa tool StegSecret juga menemukan teknik steganografi yang digunakan, yaitu EoF (*End of File*) dan juga program yang digunakan untuk menyamarkan file kedalam file Gambar.jpg, walaupun program yang disebutkan pada tool StegSecret tersebut sebenarnya kurang akurat.

Dari dua kali hasil pendeteksian yang dilakukan pada teknik 1 ini, disimpulkan bahwa hasil dari penyamaran file dengan menggunakan tool OurSecret dan *file carrier* berupa JPG bisa terdeteksi oleh tool *steganalysis.*

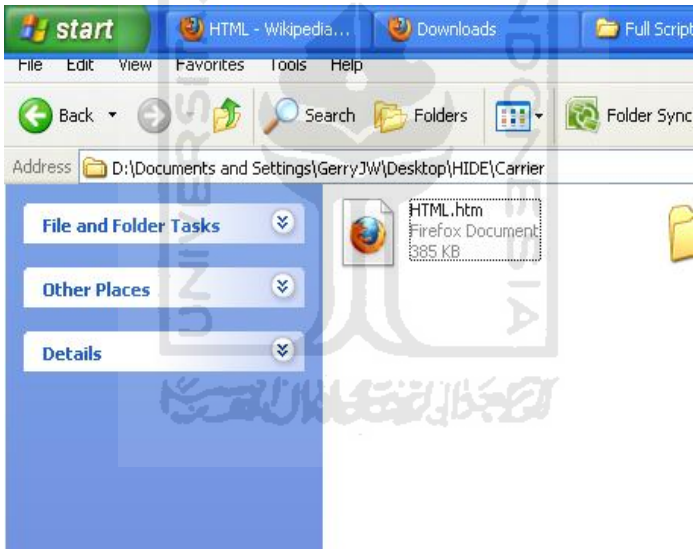
4.3 Teknik 2

Pada teknik yang kedua ini *file carrier* yang digunakan akan diganti dengan file HTML, dan tool yang digunakan untuk penyamaran file juga berbeda dari tool yang digunakan pada teknik yang pertama. Tool yang akan digunakan pada teknik yang kedua ini adalah tool WbStego4. Berikut adalah tampilan dari tool WbStego4.

Gambar 4.12 Tampilan WbStego4

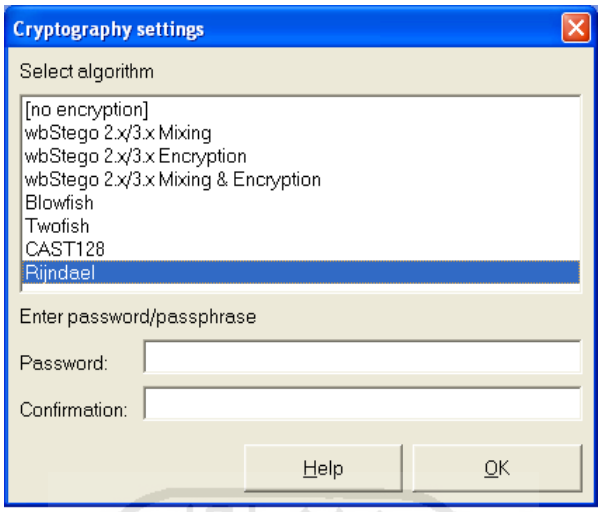
Terdapat beberapa langkah dalam penyamaran menggunakan tool WbStego4. Langkah-langkahnya adalah sebagai berikut:

1. Pada step 1 klik tombol *continue.*
2. Pada step 2 klik *Encode* untuk melakukan *hiding* data, lalu klik tombol *continue.*
3. Pada step 3, *input*-kan file yang akan di *hiding*, yaitu file DataRahasia.rar. lalu klik tombol *continue.*
4. Pada step 4, *input*-kan file yang akan digunakan sebagai *file carrier*. Disini *file carrier* yang akan digunakan adalah HTML dan bukan JPG. Dicontohkan file yang digunakan yaitu file HTML.htm. setelah file sudah di *input*-kan, klik *continue.*



Gambar 4.13 File HTML.htm

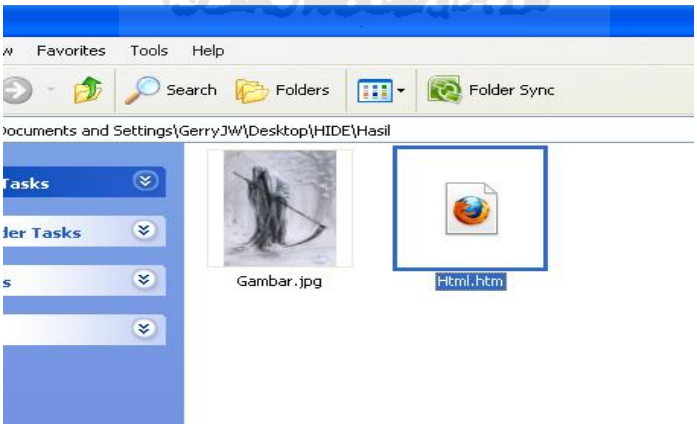
5. Pada step 5, klik pada tombol *Crypthography setting* untuk melakukan pengisian *password*. Teknik kriptografi yang digunakan adalah dengan menggunakan algoritma *rijndael.*



Gambar 4.14 *Crypthography setting*

setelah *password* diisi klik tombol OK, dan klik tombol *continue.*

6. Pada step 6, tentukan *destination* atau tujuan *output* dari WbStego4. Lalu klik tombol *continue.*
7. Pada step 7 langsung klik *continue* saja. Maka proses penyamaran file pun telah selesai. Hasilnya adalah seperti berikut.



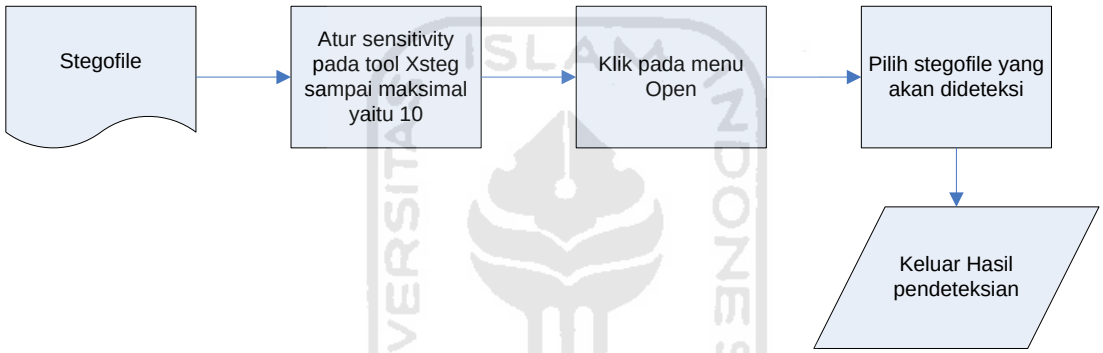
Gambar 4.15 *Stegofile* Html.htm

Proses penyamaran telah selesai dilakukan, langkah berikutnya adalah melakukan pendeteksian.

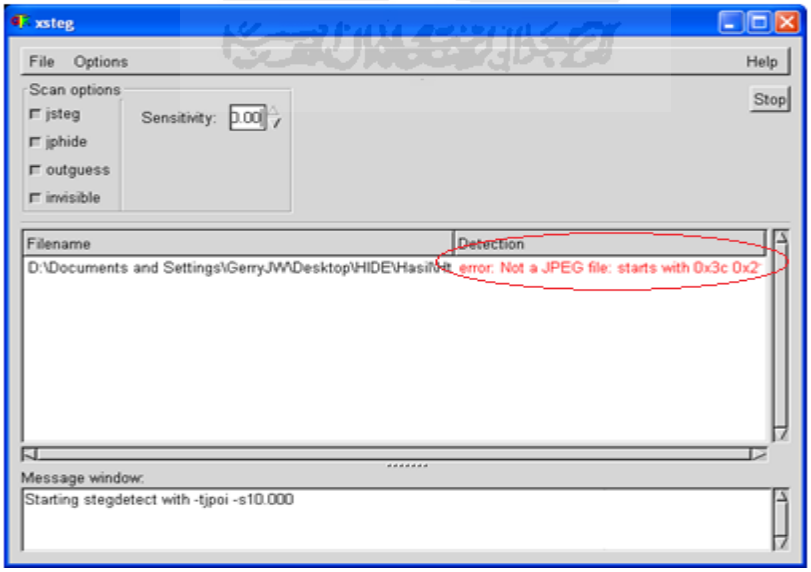
4.3.1 Pendeteksian dengan XSteg dan StegSecret

Tool yang digunakan untuk mendeteksi masih sama, yaitu tool XSteg dan StegSecret.

Pendeteksian pertama adalah menggunakan XSteg, Langkah-langkah yang dilakukan sama seperti yang dilakukan pada teknik 1, yaitu seperti gambar berikut.



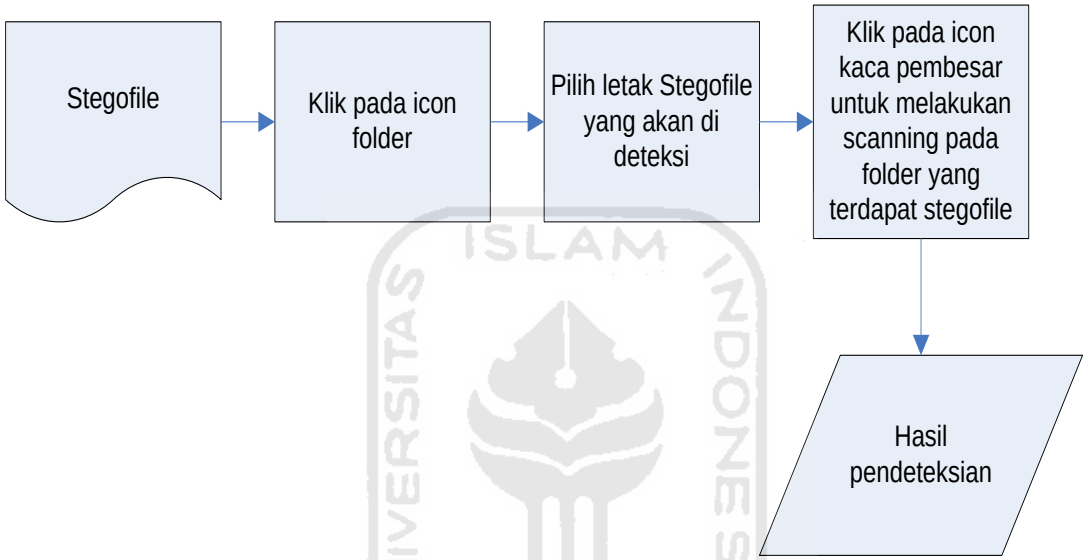
Gambar 4.16 *Flowchart* XSteg(2)



Gambar 4.17 Hasil pendeteksian menggunakan XSteg(2)

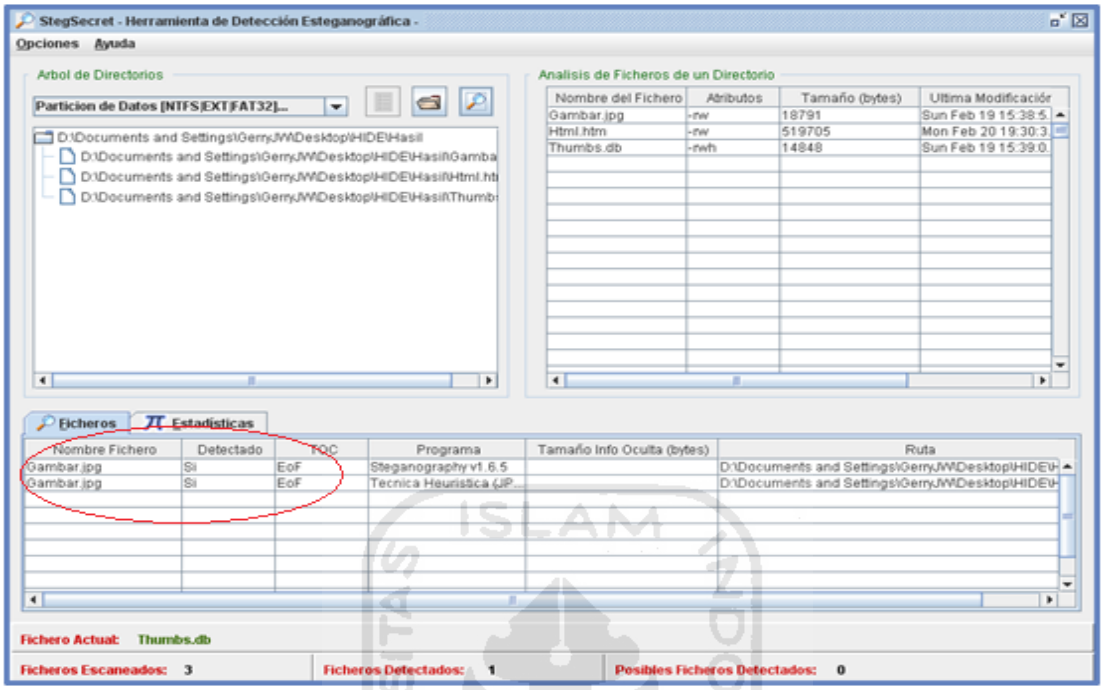
Setelah dilakukan pendeteksian pada *Stegofile* Html.htm, XSteg menghasilkan *output* "*error: Not a JPEG file".* Dari hasil *output* ini bisa disimpulkan bahwa tool XSteg tidak bisa mendeteksi *stegofile* selain dari file *image* yang berekstensikan JPEG.

Kemudian lanjut lagi ke pendeteksian yang kedua yaitu menggunakan tool StegSecret. Langkah-langkah yang dilakukan adalah sebagai berikut.



Gambar 4.18 *Flowchart* XSteg

Setelah pendeteksian dilakukan, hasil pendeteksian menggunakan tool StegSecret, menunjukkan bahwa *stegofile* Html.htm tidak terdeteksi steganografi, yang terdeteksi hanyalah *stegofile* Gambar.jpg yang merupakan hasil dari penyamaran file pada teknik 1. Karena stegofile hasil penyamaran pada teknik 1 berada dalam satu folder dengan stegofile hasil penyamaran pada teknik 2, maka stegofile teknik 1 juga ikut ter-*scanning.* Hasil pendeteksian tersebut bisa dilihat pada gambar berikut.



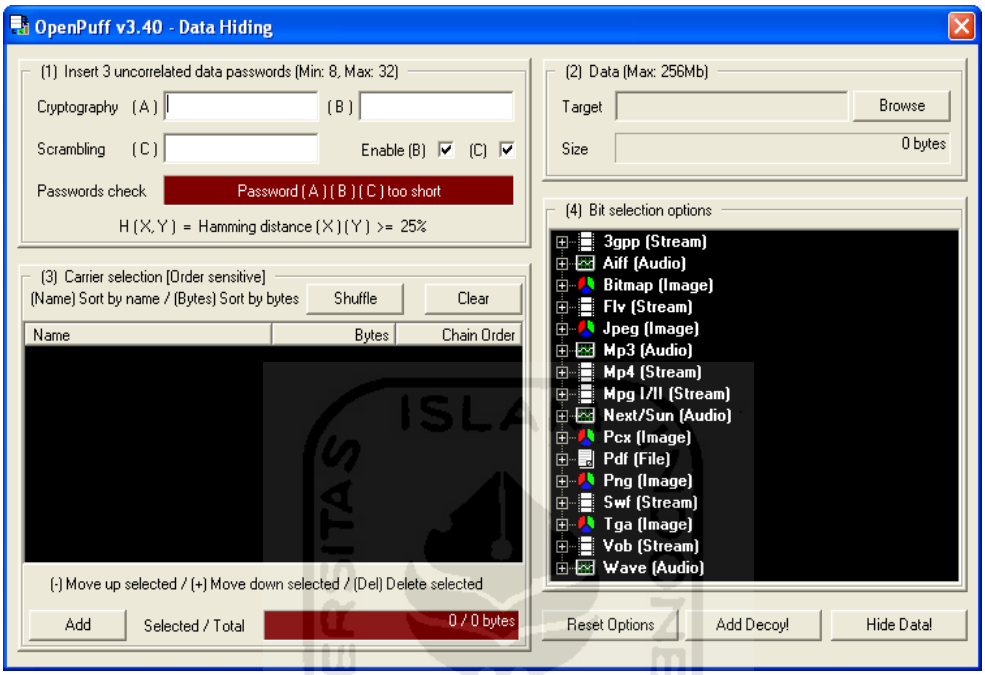
Gambar 4.19 Hasil pendeteksian StegSecret(2)

Belum diketahui mengapa steganografi yang terdapat pada file Html.htm tidak terdeteksi oleh tool StegSecret. Asumsi sementara yang didapat adalah bahwa tool StegSecret hanya bisa mendeteksi teknik steganografi yang menggunakan teknik EOF (*End of File)* yang digunakan untuk menyamarkan file. Sedangkan teknik yang digunakan oleh tool WbStego adalah teknik LSB (*Least Significant Bit).* Untuk penjelasan mengenai teknik steganografi EOF (*End of File)* dan LSB *(Least Significant Bit)* bisa dilihat pada bab landasan teori.

4.4 Teknik 3

Teknik ketiga menggunakan tool OpenPuff versi 3.40 untuk menyamarkan file DataRahasia.rar. OpenPuff merupakan salah satu tool steganografi professional, yang dimana di dalamnya terdapat beberapa yang tidak dimiliki tool penyamaran file sebelumnya, yaitu fitur *Multi File Carrier*, yaitu menggunakan lebih dari 1 *file carrier*, kemudian *Multi Cryphtography*, yaitu menggunakan lebih dari satu kali

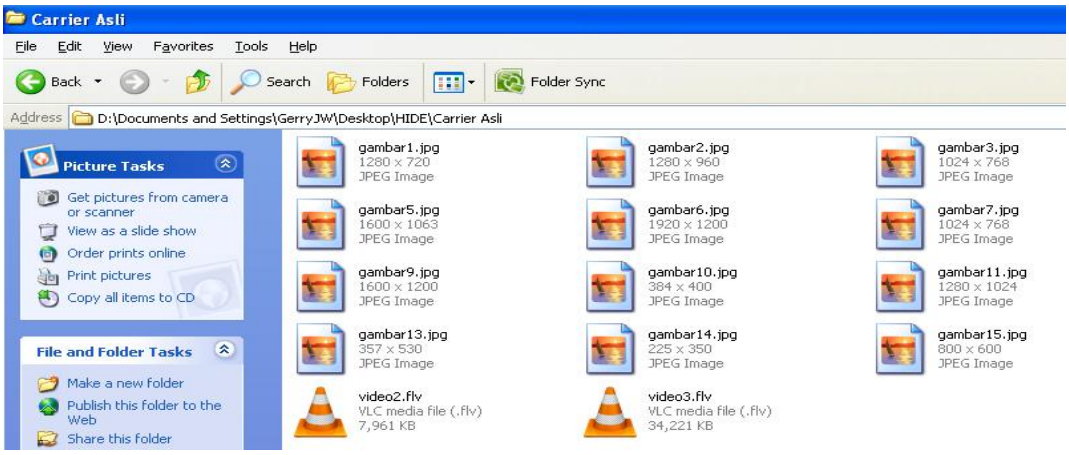
pengamanan *password*, dan fitur terakhirnya adalah *Decoy File*, yaitu membuat file palsu untuk melakukan pengecohan.



Gambar 4.20 Tampilan OpenPuff 3.40

Terdapat 5 langkah penyamaran pada tool OpenPuff v.340, Langkah-langkah penyamaran langkah-langkah tersebut adalah sebagai berikut:

1. Langkah pertama adalah pengisian *Multi Crypthography*, *multi crypthography* disini adalah berupa password, terdapat 3 kolom password, dan tiap kolom harus berisi password yang berbeda.
2. Langkah kedua adalah penginputan file yang akan disamarkan, pada langkah ini *input*-kan file DataRahasia.rar.
3. Langkah yang ketiga adalah penginputan *file carrier* yang akan digunakan. seperti yang diberi tahu sebelumnya bahwa tool OpenPuff ini menggunakan *multi file carrier* yaitu menggunakan lebih dari satu *file carrier* untuk menyamarkan suatu file. *File carrier* yang digunakan adalah seperti berikut.



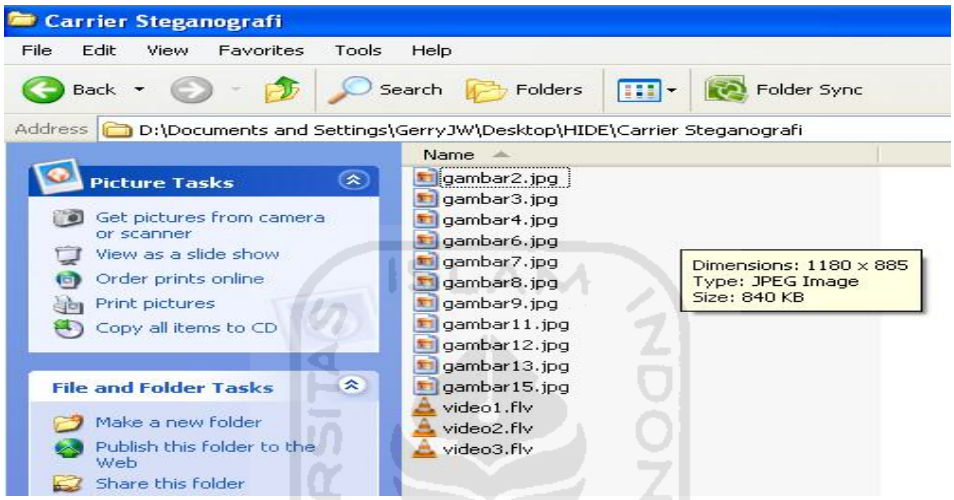
Gambar 4.21 *Multi File Carrier*

Dicontohkan *file carrier* yang digunakan disini adalah berupa file *image* yang berekstensi JPG dan file video yang berekstensi FLV dan jumlahnya lebih dari satu.

4. Dan untuk Langkah yang keempat adalah langkah untuk meningkatkan bit yang terdapat pada suatu *file carrier*. Tujuannya adalah agar *byte file carrier* lebih besar dari pada *byte* file yang DataRahasia.rar. jika pada waktu penginputan *byte file carrier* sudah lebih besar daripada *byte* file DataRahasia.rar, maka langkah keempat ini tidak perlu dilakukan.
5. Dan langkah terakhir sebelum melakukan proses *hiding* adalah langkah membuat *Decoy file* atau file palsu. Klik pada tombol *add decoy* pada OpenPuff. Langkah yang dilakukan hampir sama dengan penyamaran yang dilakukan pada file aslinya. Mulai dari pengisian password sampai penginputan file palsu yang akan disamarkan. Hanya saja password yang diinputkan dan file yang akan disamarkan berbeda. Dan ukuran file palsu yang digunakan pun tidak boleh jauh berbeda dari file aslinya. Misalnya ukuran file asli yang akan disamarkan berukuran 34 KB, maka file palsunya pun ukurannya tidak boleh terlalu jauh dari 34 KB, misal 32 KB, 36 KB.

Fungsi dari *decoy file* ini nanti adalah hanya dengan menginputkan password yang palsu, maka yang akan keluar adalah file yang palsu bukan file yang asli.

Setelah semua langkah sudah dilakukan, klik tombol *Hide* untuk menyamarkan file. berikut adalah hasil dari penyamaran menggunakan OpenPuff v.3.40.



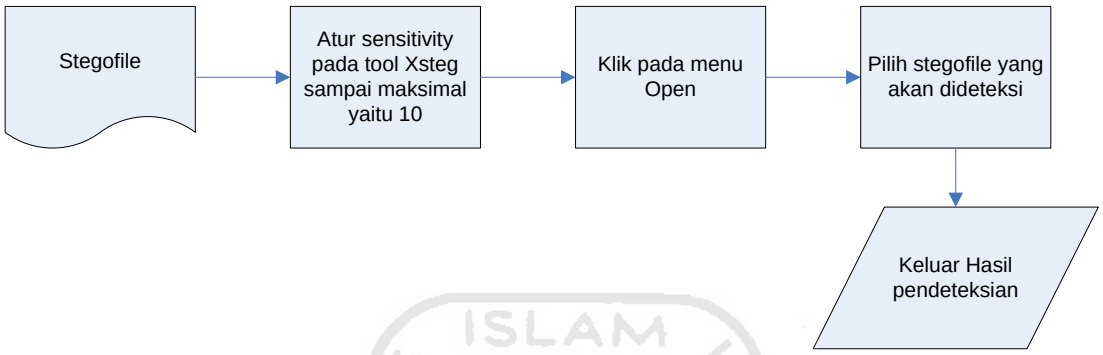
Gambar 4.22 *Output stegofile* OpenPuff v.3.40

Teknik penyamaran menggunakan OpenPuff ini menghasilkan lebih dari satu *stegofile*. File DataRahasia.rar yang disamarkan tidak akan bisa terbuka apabila salah satu dari *stegofile* tersebut hilang atau tidak lengkap, jadi agar file DataRahasia.rar bisa terbuka, maka semua *stegofile* tersebut harus lengkap, hal ini bisa menjadi kelebihan dan bisa juga menjadi kekurangan. Kelebihannya adalah dapat membuat suatu file semakin aman dan sulit untuk ditemukan, sedangkan kekurangannya adalah apabila salah satu file hilang, maka file DataRahasia.rar tidak akan bisa terbuka kembali.

Seperti pada teknik penyamaran file sebelumnya, pada teknik ini juga akan dilakukan pendeteksian. Dikarenakan *stegofile* pada penyamaran ini lebih dari satu dan terdapat dua ekstensi file yang berbeda yaitu JPG dan FLV. Pada tool XSteg pendeteksian yang dilakukan adalah dengan mengambil 1 sample file dari tiap ekstensi.

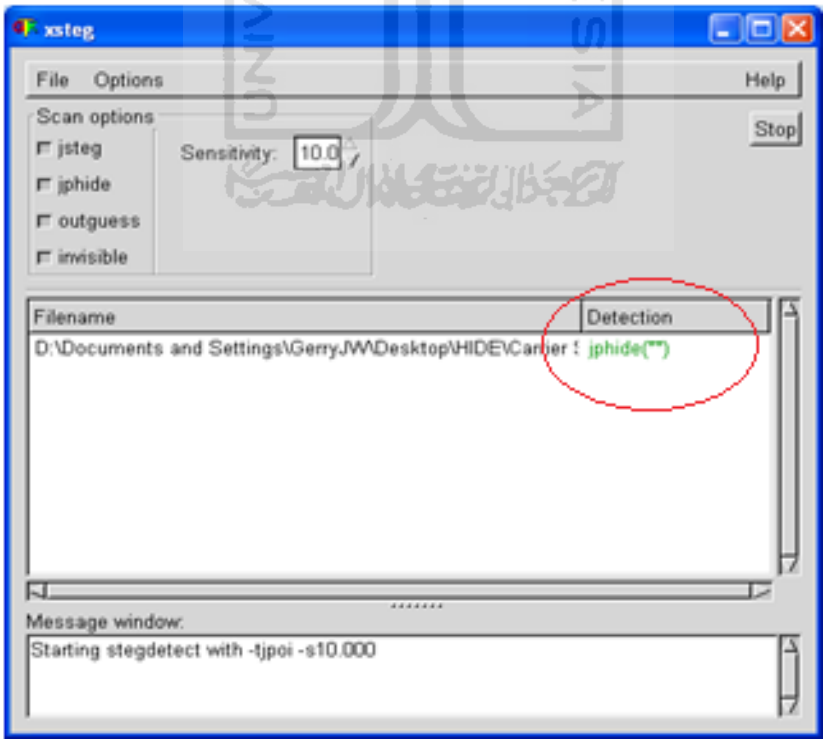
4.4.1 Pendeteksian dengan XSteg dan StegSecret

Pendeteksian pertama adalah menggunakan tool XSteg terhadap *stegofile* yang berektensi JPG. Langkah-langkah yang dilakukan adalah sebagai berikut.



Gambar 4.23 *Flowchart* XSteg(3)

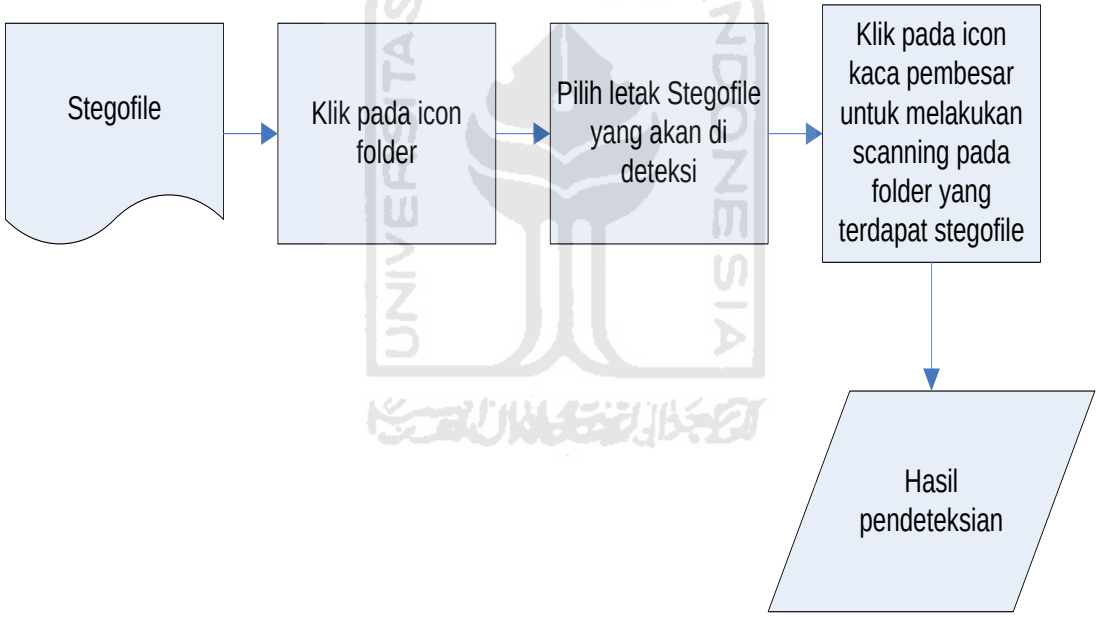
Setelah dilakukan pendeteksian, maka hasilnya adalah sebagai berikut.



Gambar 4.24 Hasil pendeteksian XSteg(3)

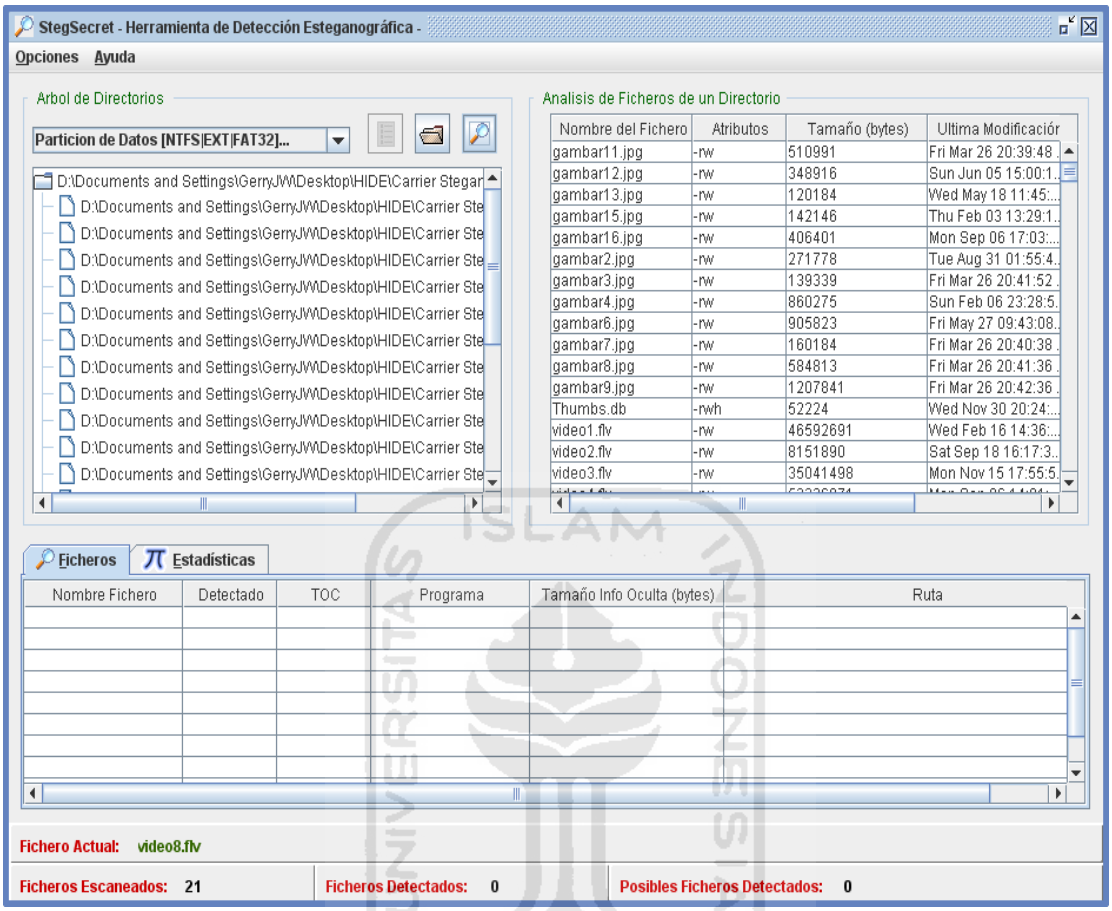
Setelah dilakukan pendeteksian ternyata XSteg berhasil mendeteksi steganografi yang terdapat pada *stegofile* yang berekstensi JPG. Selanjutnya untuk file yang berekstensi FLV tidak perlu dilakukan pendeteksian, karena seperti yang dilakukan pada teknik 2, pendeteksian selain file yang berekstensi JPG akan menghasilkan *error*.

Lanjut lagi ke pendeteksian yang kedua yaitu menggunakan tool StegSecret, StegSecret ini akan melakukan pendeteksian terhadap *stegofile* yang berekstensi JPG dan FLV sekaligus. Seperti yang diketahui sebelumnya bahwa cara kerja pendeteksian tool StegSecret ini adalah dengan melakukan scanning terhadap semua file yang terdapat didalam satu folder. Langkah-langkah yang dilakukan adalah sebagai berikut.



Gambar 4.25 *Flowchart* StegSecret(3)

setelah langkah-langkah pendeteksian dilakukan, maka hasil yang dikeluarkan oleh StegSecret adalah sebagai berikut.



Gambar 4.26 Hasil pendeteksian StegSecret (3)

Dari hasil pendeteksian yang dilakukan, tidak dideteksi bahwa *stegofile* yang berekstensi JPG dan FLV tersebut mengandung steganografi.

4.5 Kesimpulan

Berikut kesimpulan yang didapat dari hasil penyamaran yang dilakukan mulai dari pengenkripsian file sampai dengan teknik penyamaran file yang dilakukan. Hasil yang didapat dari penyamaran file yang dilakukan bisa dilihat pada tabel berikut.

Tabel 4.1 Hasil dari teknik penyamaran file

	Tool	**Carrier**	**Xsteg**	**StegSecret**
Teknik 1	OurSecret	Image (JPG)	Terdeteksi	Terdeteksi
Teknik 2	WbStego4	HTML	Error	Tidak Terdeteksi
Teknik 3	OpenPuff v.3.40	Image (JPG)	Terdeteksi	Tidak Terdeteksi
		Video (Flv)	Error	Tidak Terdeteksi

Dan untuk spesifikasi tool penyamaran file yang digunakan bisa dilihat pada tabel berikut.

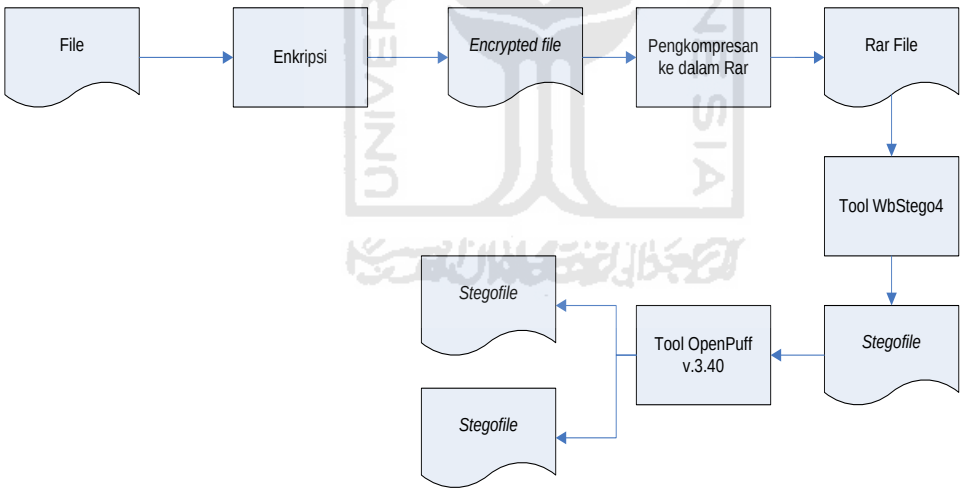
Tabel 4.2 Tool yang digunakan pada penyamaran file

No.	Tool	Kripto grafi	Stegano grafi	Fitur Lain	Carrier yang dapat digunakan
1	OurSecret	Ada	ada	Tidak ada	Wav, Mp3, Jpeg, Bitmap, Flv
2	WbStego4	Ada	ada	Tidak ada	Bitmap, TXT, HTML, PDF
3	OpenPuff v.3.40	Ada	ada	Multi Crypthography	Bitmap, Flv, Jpeg, Pcx, Pdf,
				Multi Carrier	Mp4, Mpg, Png, Swf, Tga, Wave,
				Decoy File	Mp3

Dari 3 teknik penyamaran file yang dilakukan, bisa dilihat bahwa *stegofile* yang terdapat pada teknik 2 berhasil lolos dari pendeteksian tool *steganalysis.* Sedangkan *stegofile* pada teknik 1 berhasil terdeteksi oleh tool *steganalysis.* Dan pada teknik 3, hanya tool XSteg saja yang berhasil mendeteksi steganografi pada

stegofile sedangkan tool StegSecret tidak dapat mendeteksi steganografi yang terdapat pada *stegofile*. Pada teknik 3 ini tool yang digunakan memiliki kelebihan dibanding 2 tool yang digunakan pada teknik yang lain, yaitu *multi file carrier, multi cryphtography* dan *decoy file.*

Dari 3 teknik penyamaran file yang dilakukan, bisa disimpulkan bahwa teknik 2 yang menggunakan tool WbStego4 dan HTML sebagai *file carrier* nya merupakan teknik yang berhasil dalam penyamaran file, karena penyamaran file yang dilakukan berhasil tidak terdeteksi oleh 2 tool *steganalysis* yang digunakan. Jika teknik 2 ini digabungkan dengan teknik 3 yang memiliki fitur *multi file carrier, multi crypthography* dan *decoy file* maka ada kemungkinan penyamaran file akan semakin bagus. Dan file yang disamarkan pun akan semakin sulit untuk di *decode*. Jika kedua teknik tersebut digabungkan, maka hasilnya adalah sebagai berikut.



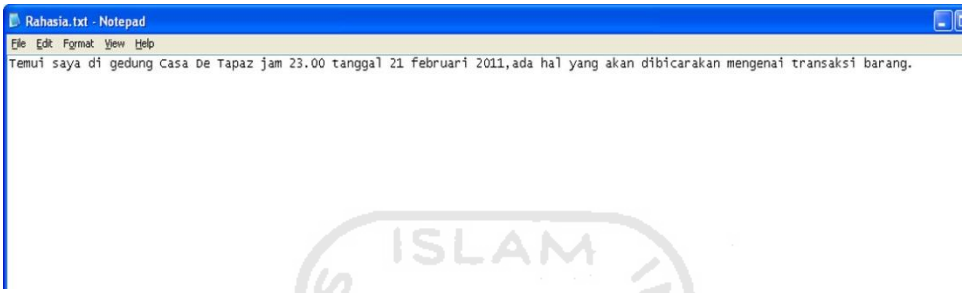
Gambar 4.27 Kombinasi 2 teknik penyamaran file

4.6 Kombinasi Teknik penyamaran file

Dari penggabungan dua teknik penyamaran file yaitu teknik 2 dan teknik 3 menghasilkan suatu kombinasi teknik penyamaran file yang lebih kompleks. Bisa

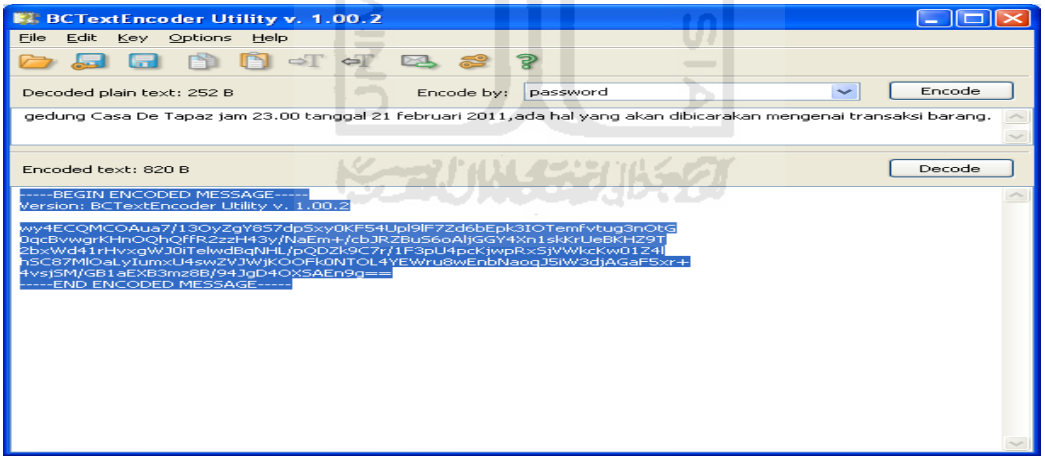
dilihat pada gambar 4.22 di atas. Penjelasan tentang penyamaran file yang dilakukan pada teknik kombinasi tersebut adalah sebagai berikut:

1. Sediakan file yang akan disamarkan, yaitu file txt, dicontohkan file yang digunakan disini adalah file Rahasia.txt. pada file Rahasia.txt ini berisi pesan rahasia.



Gambar 4.28 File Rahasia.txt

2. Lakukan enkripsi pada pesan rahasia tersebut dengan menggunakan tool BCTextEncoder.



Gambar 4.29 Hasil enkripsi teks (2)

3. Hasil dari enkripsi tersebut di *copy paste* kan ke dalam file Rahasia.txt dan kemudian di *save*. Setelah itu file Rahasia.txt tersebut di kompres kedalam bentuk Rar, dan menghasilkan file Rahasia.rar.

   Fungsi dari pengkompresan tersebut adalah agar enkripsi yang terdapat pada file Rahasia.txt tersebut tidak rusak.

4. Setelah di kompres dan menghasilkan file Rahasia.rar, file rar tersebut kemudian di *hide* menggunakan tool WbStego4 dan untuk *file carrier* yang digunakan adalah file HTML.htm. Untuk tahap-tahap penyamaran file nya sama seperti yang dilakukan pada teknik 2.

   Setelah proses penyamaran selesai maka dihasilkan *stegofile* HTML.htm
5. Dan *stegofile* hasil penyamaran menggunakan tool WbStego4, akan disamarkan kembali menggunakan tool OpenPuff 3.40. *File carrier* yang digunakan juga masih sama seperti pada teknik 3, yaitu JPG dan FLV. tahap-tahap penyamaran filenya juga sama seperti yang dilakukan pada teknik 3. Mulai dari pengisian *multi cryptography* sampai dengan pembuatan *decoy file*. Setelah penyamaran selesai dilakukan, maka dihasilkan lebih dari satu *stegofile*.
6. Setelah semua proses penyamaran file selesai, tidak perlu dilakukan pendeteksian lagi, karena sudah bisa diketahui hasilnya. Hasilnya sama seperti pendeteksian yang dilakukan pada teknik 3, yaitu hanya tool XSteg saja yang dapat mendeteksi, sedangkan tool StegSecret tidak dapat mendeteksi.

Dari kombinasi penyamaran file diatas bisa diketahui bahwa file Rahasia.txt akan semakin sulit untuk di *decode* dan ditemukan. Dengan melakukan penyamaran file menggunakan kombinasi teknik seperti diatas, maka keamanan file Rahasia.txt semakin berlapis.

Ada 2 fungsi yang dapat dilakukan dari melakukan teknik kombinasi dari penyamaran file diatas, yaitu:

1. Jika ingin menggunakan hasil kombinasi teknik penyamaran file di atas untuk melakukan pengiriman file kepada seseorang lewat media internet, bisa dilakukan dengan menyebarkan *stegofile* yang dihasilan ke beberapa media, misalnya *stegofile* 1 dikirim melalui facebook, *stegofile* 2 dikirim melalui email dan seterusnya. Maka pesan yang dikirim akan aman. walaupun misalnya ada salah satu *stegofile* ditemukan oleh seseorang selain *receiver*, dan dideteksi bahwa stegofile tersebut mengandung steganografi, orang tersebut tetap tidak akan bisa

membuka apa yang ada di dalam *stegofile* tersebut. Penyebaran *stegofile* tersebut harus diketahui terlebih dahulu oleh *receiver*, dan teknik *decode* nya pun harus sudah diketahui oleh *receiver* tersebut.

2. Selain untuk pengiriman pesan, teknik kombinasi tersebut juga bisa digunakan untuk pengamanan file penting yang ada di komputer. Misalnya terdapat sebuah file dokumen, dan isi dari file dokumen tersebut berisi data-data penting. Agar file tersebut tidak diganggu dan di rusak oleh orang yang tidak bertanggung jawab maka dengan melakukan penyamaran menggunakan teknik kombinasi tersebut, file dokumen tersebut akan aman dari orang yang tidak bertanggung jawab tersebut. contoh: *hacker*.

Teknik kombinasi penyamaran file diatas membuat penyamaran suatu file semakin aman dan sulit untuk ditemukan. Hanya seorang *receiver* dan orang yang menyembunyikannya saja yang dapat mengekstraksi kembali penyamaran file tersebut. Jika orang lain ingin mengekstraksi kembali file yang disamarkan tersebut, maka orang tersebut harus bekerja sangat keras. Seperti tujuan anti forensik yaitu Bagaimana membuat supaya data sulit ditemukan atau dibuka.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari penelitian tentang "Anti Forensik dalam penyamaran file dengan Kriptografi dan Steganografi" maka dapat disimpulkan bahwa:

1. Menggunakan fitur *multi crypthography, multi file carrier,*dan *decoy file* pada tool steganografi merupakan hal yang sangat baik karena dapat memaksimalkan penyamaran suatu file.
2. Dengan menggabungkan beberapa tool steganografi dan kriptografi dalam penyamaran file maka keamanan penyamaran suatu file akan lebih baik.
3. Tool XSteg tidak bisa mendeteksi *stegofile* selain file image yang berekstensi JPG.
4. Dengan mengkompres *ciphertext* ke dalam bentuk rar dapat membuat *ciphertext* tersebut tidak rusak ketika disamarkan menggunakan tool steganografi yang terdapat kriptografi di dalamnya. Karena jika *ciphertext* tersebut rusak, maka pesan yang terdapat didalamnya tidak akan bisa dibuka kembali.

5.2 Saran

Saran yang dapat diberikan setelah penelitian ini dilakukan adalah:

1. Tool *steganalysis* yang digunakan harus *up to date* agar dapat mendeteksi steganografi-steganografi yang terdapat pada tool-tool yang baru. Seperti tool XSteg, sebaiknya tidak hanya dapat mendeteksi file *image* yang berupa Jpeg saja,

tetapi juga dapat mendeteksi *stegofile* berupa file *image* yang berekstensi lain, video, dan audio.

2. Sebaiknya file dokumen yang digunakan tidak hanya berupa txt, tetapi juga berupa word, exel dan sebagainya.

3. Sebaiknya *file carrier* yang digunakan tidak hanya berupa file image dan audio saja, tetapi juga menggunakan file video.

4. Menggunakan *file carrier* lebih dari satu lebih aman dibandingkan hanya dengan menggunakan satu file carrier saja.

5. Dalam pemberian password, gunakan password yang berupa kombinasi antara huruf dan angka.

6. Sebaiknya pada setiap tool steganografi, didalamnya terdapat fitur *multi crypthography, multi file carrier,* dan *decoy file*.