Integrasi Metode Steganografi DCS Pada Image Dengan Kriptografi Blowfish Sebagai Model Anti Forensik Untuk Keamanan Ganda Konten Digital

Ermadi Satriya Wijaya[1]
Program Magister Teknik Informatika,
Fakultas Teknologi Industri,
Universitas Islam Indonesia
Jl. Kaliurang Km. 14,5 Yogyakarta 55501
Telp. (0274) 895287 ext. 114, Faks. (0274) 895007
ermadi_satriya@yahoo.com

Yudi Prayudi[2]
Program Magister Teknik Informatika,
Fakultas Teknologi Industri,
Universitas Islam Indonesia
Jl. Kaliurang Km. 14,5 Yogyakarta 55501
Telp. (0274) 895287 ext. 114, Faks. (0274) 895007
prayudi@fti.uii.ac.id

***Abstrak*—Anti forensik merupakan segala hal yang berkaitan dengan upaya-upaya untuk mempersulit dalam hal pelacakan barang bukti pada kasus kejahatan digital (*Cyber Crime*). Salah satu teknik anti forensik yang paling dikenal adalah metode penyembunyian data (*Data Hiding*) yang mempunyai dua metode yaitu *Steganografi*, serta *Kriptografi*. Dalam penelitian ini akan membahas bagaimana menerapkan dan menguji konsep keamanan ganda pada konten digital dalam pengembangan model anti forensik dengan mengintegrasikan antara *Steganografi* teknik DCS (*Dinamic Cell Spreading*) yang merupakan teknik menyembunyikan data menggunakan *buffer* memori sebagai media penggabungan, dan *Kriptografi* algoritma *Blowfish* yang merupakan teknik enkripsi atau dekripsi pada konten digital dengan memiliki kemampuan lebih dari algoritma *kriptografi* lainnya. Rancangan pengujian pada penelitian ini didasarkan pada dari hasil uji kapasitas metode *steganografi*, uji ketahanan sistem *steganografi*, dan uji deteksi anti *steganografi*. Hasil pengujian yang ada merupakan konsep untuk membuat standar permodelan anti forensik untuk keamanan ganda. Hasil dari penelitian ini tentang impelementasi integrasi antara *Steganografi* DCS dengan *Kriptografi* algoritma *Blowfish* serta menguji keluaran (*output*) dari hasil proses integrasi yang ada sehingga dapat menjadi referensi perbandingan pengujian kemampuan pengembangan standar keamanan ganda konten digital.**

***Kata kunci*—*Anti Forensik*, Cyber Crime, Data Hiding, Steganografi *Teknik* Dinamic Cell Spreading, Kriptografi *Algoritma* Blowfish.**

I. PENDAHULUAN

Selain berkembangnya teknik forensik untuk menemukan barang bukti ternyata berkembang pula teknik keamanan yang dari satu aspek memberikan manfaat bagi pengguna dalam meningkatkan keamaan, pada aspek yang lain lebih berdampak pada terpenuhinya teknik anti forensik. Anti forensik merupakan segala hal yang berkaitan dengan upaya-upaya untuk mempersulit dalam hal pelacakan barang bukti pada kasus kejahatan digital (*Cyber Crime*), diantaranya menurunkan kualitas atau mengkaburkan barang bukti digital, konsep yang menyebabkan berpindahnya barang bukti ketempat lain hingga menghilangkannya barang bukti serta menyebabkan barang bukti tersebut sulit untuk terlacak atau diungkap.

Konsep *Steganografi* yang digunakan pada penelitian ini menggunakan teknik DCS dikarenakan berdasarkan pernyataan dari Wijaya & Prayudi [9] bahwa teknik DCS mempunyai teknik penyembunyian yang unik yaitu dengan menggunakan buffer memori sebagai media bantu penyisipan pada LSB menjadikan teknik DCS mempunyai teknik penyisipan yang lebih terstruktur dalam menerapkan konsep *Steganografi*. Selain itu pada penelitian ini juga menggunakan konsep *Kriptografi* dengan algoritma *Blowfish* yang dijelaskan oleh Irawan [3] bahwa konsep *Kriptografi* dengan algoritma *Blowfish* merupakan *Kriptografi* dengan merupakan algoritma enkripsi dengan model private key/kunci pribadi yang mempunyai key dekripsi sama dengan key enkripsi sedangkan penilaian tingkat keamanan pada *Blowfish* sangat tinggi, terbukti sampai saat ini belum ada *Cryptoanalisis* yang berhasil memecahkan kelemahan algoritma *Blowfish*.

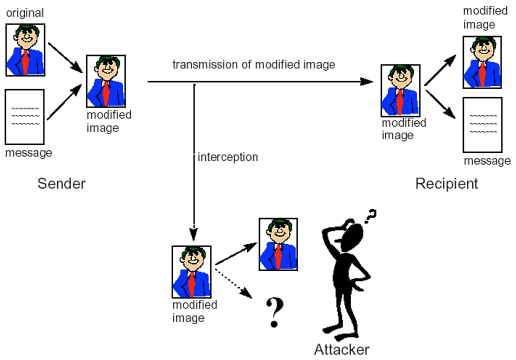
Hasil implementasi ini adalah melakukan uji keberhasilan ekstraksi data dan ketahanan image *steganografi* pada tingkat keamanan dalam konsep integrasi antara *Kriptografi* dengan algoritma *Blowfish* dan *Steganografi* dengan teknik DCS. Penerapan standar keamanan ganda dimaksudkan agar mampu menutupi kelemahan pada metode *Steganografi* dengan cara menambahkan konsep *Kriptografi*. Standar keamanan ganda tersebut diharapkan mampu menjawab konsep penyebaran pesan rahasia yang lebih aman serta tidak terlacak keberadaannya.

II. LANDASAN TEORI

Teknik anti forensik merupakan segala hal yang berkaitan dengan upaya-upaya untuk mempersulit dalam hal pelacakan barang bukti pada kasus kejahatan digital (*Cyber Crime*), diantaranya menurunkan kualitas atau mengkaburkan barang bukti digital, konsep yang menyebabkan berpindahnya barang bukti ketempat lain hingga menghilangkannya barang bukti serta menyebabkan barang bukti tersebut sulit untuk terlacak atau diungkap.

A. Konsep Steganografi

Zöllner, et al. [10] mengatakan bahwa *Steganografi* adalah ilmu pengetahuan dan seni dalam menyembunyikan komunikasi. Suatu sistem *Steganografi* sedemikian rupa menyembunyikan isi suatu data di dalam suatu sampul media yang tidak dapat di duga oleh orang biasa sehingga tidak membangunkan suatu kecurigaan kepada orang yang melihatnya, dapat anda lihat dalam gambar 1. Di masa lalu, orang-orang menggunakan tato tersembunyi atau tinta tak terlihatan untuk menyampaikan isi *Steganografi*. Hari ini, teknologi jaringan dan komputer menyediakan cara yang mudah dalam menggunakan jaringan komunikasi untuk *Steganografi*.

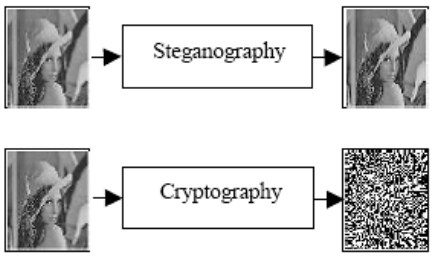


Gambar 1 Gambaran umum tentang proses *Steganografi.*

Provos & Honeyman [5] berpendapat tujuan *Steganografi* modern adalah untuk mempertahankan suatu media yang tidak bisa mendeteksi, tetapi karena sistem *Steganografi* masih memiliki kelemahan yang meninggalkan jejak dibelakang sampul media sehingga dapat ditemukan. Sekalipun isi rahasia tidaklah diungkapkan, keberadaan tentang memodifikasi sampul media dapat merubah sifat statistik, jadi para peneliti dapat mendeteksi distorsi dihasil dari proses media stego dengan sifat stastistik. Maka proses untuk pencarian dan mendeteksi penyimpangan di dalam media yang distorsi disebut sebagai "*Statistical Steganalysis*".

B. Perbedaan Steganografi dengan Kryptografi

Zöllner, et al. [10] menjelaskan tentang perbedaan antara *Steganografi* dengan *Kriptografi*, letak perbedaannya adalah hasil keluarannya. Hasil dari *Kriptografi* biasanya berupa data yang berbeda dari bentuk aslinya dan biasanya datanya seolah-olah berantakan (tetapi dapat dikembalikan ke bentuk semula) sedangkan hasil keluaran dari *Steganografi* ini memiliki bentuk persepsi yang sama dengan bentuk aslinya, tentunya persepsi disini oleh indera manusia, tetapi tidak oleh komputer atau perangkat pengolah digital lainnya.



Gambar 2 Ilustrasi *Steganografi* dan *Kriptografi* pada Citra.

C. Konsep teknik DCS (Dinamic Cell Spreading)

Teknik DCS (*Dynamic Cell Spreading*) merupakan *steganografi* dengan menggunakan model proteksi terhadap deteksi yang dikembangkan oleh Ohmacht [4] dengan memiliki konsep yaitu menyembunyikan file pesan dalam bentuk semua data digital ke dalam media gambar BMP (Bitmap) dengan menggunakan cara menyisipkanya pada bit rendah LSB (*Least Significant Bit*) pada data pixel yang menyusun file tersebut dengan bantuan menggunakan *buffer* memori sebegai media penyimpan sementara. Dalam proses penggabungan (*stego*) antara file gambar dengan teks maka prinsip dasarnya adalah sebagai berikut seperti kita ketahui untuk file bitmap 24 bit maka setiap pixel (titik) pada gambar tersebut terdiri dari susunan tiga warna merah, hijau dan biru (RGB) yang masing-masing disusun oleh bilangan 8 bit (byte) dari 0 sampai 255 atau dengan format biner 00000000 sampai 11111111. Dengan demikian pada setiap pixel file bitmap 24 bit kita dapat menyisipkan 3 bit data. Contohnya huruf A dapat kita sisipkan 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 menyisipkannya pada data pixel diatas maka akan dihasilkan:

(00100111 11101000 11001000)
(00100110 11001000 11101000)
(11001001 00100111 11101001)

Terlihat hanya empat bit rendah yang berubah, untuk mata manusia maka tidak akan tampak perubahannya. Secara rata-rata dengan metoda ini hanya setengah dari data bit rendah yang berubah, sehingga bila dibutuhkan dapat digunakan bit rendah kedua bahkan ketiga.

Sedangkan pada proses penggabungan file gambar dengan data elektronik maka prosesnya hampir sama tetapi lebih kompleks karena membutuhkan media memori sebagai perantara untuk menghitung jumlah keseluruhan bit yang terdapat didalam file gambar maupun didalam data elektronik yang akan diembedding sehingga memudahkan proses embedding itu sendiri.

Penghitungan aritmatika dalam melakukan embedding maupun ekstrakting ini menggunakan perintah

assembler karena menyangkut bit-bit yang terdapat di dalam memori.

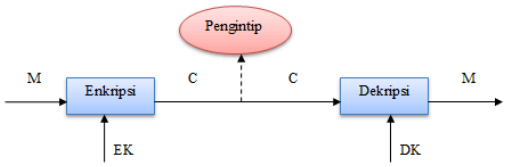
D. Konsep Kriptografi

Andi [1] menyebukan bahwa Kriptografi adalah suatu ilmu ataupun seni mengamankan pesan, dan dilakukan oleh *Cryptographer*. Sedangkan *Cryptoanalysis* adalah suatu ilmu dan seni membuka (*breaking*) *chipertext* dan orang yang melakukannya disebut *Cryptoanalyst*. *Cryptographic System* atau *Cryptosystem* adalah suatu fasilitas untuk mengkonversikan *plaintext* ke *chipertext* dan sebaliknya. Dalam sistem ini, seperangkat parameter menentukan transformasi penchiperan tertentu yang disebut suatu set kunci. Proses enkripsi dan dekripsi diatur oleh satu atau beberapa kunci Kriptografi. Secara umum, kunci-kunci yang digunakan untuk proses pengekripsian dan pendekripsian tidak perlu identik, tergantung pada sistem yang digunakan.

Algoritma *Kriptografi* terdiri dari algoritma enkripsi (E) dan algoritma dekripsi (D). Algoritma enkripsi menggunakan kunci enkripsi (EK), sedangkan algoritma dekripsi menggunakan kunci dekripsi (DK). Secara umum operasi enkripsi dan dekripsi dapat diterangkan secara matematis sebagai berikut:

EK (M) = C ➔ E (Proses Enkripsi)
DK (C) = M ➔ D (Proses Dekripsi)
Dimana: M = Plaintext, C = Chipertext

Pada saat proses enkripsi kita menyandikan pesan M dengan suatu kunci EK lalu dihasilkan pesan C. Sedangkan pada proses dekripsi, pesan C tersebut diuraikan dengan menggunakan kunci DK sehingga dihasilkan pesan M yang sama seperti pesan sebelumnya. Untuk lebih jelasnya lihat gambar 3.



Gambar 3. *Cryptosystem* Secara Umum

E. Konsep algoritma Blowfish

Supani [8] menjelaskan bahawa *Blowfish* dirancang oleh Bruce Schneier yang ditujukan untuk mikroprosesor 32-bit ke atas dengan cache memori. *Blowfish* di kembangkan untuk memenuhi kriteria desain sebagai berikut:

1. Cepat. Pada implementasi yang optimal *Blowfish* dapat mencapai kecepatan 26 *clock cycle per byte*.
2. Tersusun rapi (*compact*). *Blowfish* dapat berjalan pada memori kurang dari 5 KB.
3. Sederhana (*simple*). *Blowfish* hanya menggunakan operasi yang sederhana yaitu penjumlahan, XOR dan penelusuran tabel (*table lookup*) pada operand 32-bit. Desainnya mudah untuk dianalisa yang membuatnya tahan terhadap kesalahan (*errors)* implementasi.
4. Keamanannya yang variabel. Panjang key (kunci) dapat bervariasi dan dapat menjadi sepanjang 448 bit (56 byte).

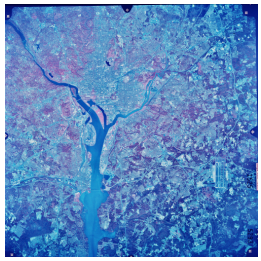
III. IMPLEMENTASI, HASIL DAN PEMBAHASAN

A. Implementasi

Pada tahap ini akan dilakukan implementasi sistem yaitu pembuatan program integrasi *Steganografi* teknik DCS dengan *Kriptografi* algoritma *Blowfish* menggunakan bahasa pemrograman Pascal dan pembuatan interface sistem dengan menggunakan aplikasi pemrograman visual Borland Delphi 7. Hasil dari proses implementasi yang didapatkan berupa program integrasi antara *Steganografi* teknik DCS dengan *Kriptografi* algoritma *Blowfish* adalah program "StegoBlow"

B. Sample Image

Dalam proses analisa hasil maka dibutuhkan sebuah *sample image* untuk dapat menganalisa program-program yang akan diujikan, untuk hal tersebut maka *sample image* diambil dari *database* laboratorium SIPI (*Signal and Image Processing Institute*) USC (*University of Southern California*). *Database* gambar pada USC-SIPI adalah koleksi gambar digital. Hal ini dipertahankan terutama untuk mendukung penelitian dalam pengolahan gambar, analisis citra, dan visi mesin.



Gambar 4. *Sample Image* diambil dari Database SIPI-USC

Sedangkan untuk melakukan uji perbandingan hasil menggunakan format data berbentuk BMP (*Bitmap*) maka perlu adanya proses konversi terlebih dahulu dengan menggunakan program Photoshop untuk mengubah dari format TIFF menjadi BMP tanpa menghilangkan nilai parameter-paremeter yang ada didalam image. Proses pengujian pada penelitian ini menggunakan *sampel image* BMP dikarenakan format BMP merupakan format standar yang dapat digunakan maupun dilihat dengan berbagai macam program gambar standar apapun dan belum mengalami perubahan maupun kopresi warna, setelah mendapatkan hasil konversi dalam bentuk BMP maka dapat dilakukan uji *sample image* yang pada penelitian ini juga menggunakan program pembanding yaitu program Stego PNG dan OpenPuff.

C. Program Stego PNG

Stego PNG merupakan program yang dikembangkan oleh perusahaan *Hermetic System*. Stego PNG adalah program *Windows* untuk menyembunyikan file data dalam PNG tunggal atau BMP file gambar dan untuk mengekstraksi file data yang disembunyikan dengan cara ini. File mungkin apapun, bukan hanya file teks, dan dengan demikian mungkin file seperti dokumen MS Word, spreadsheet Excel atau file gambar lain.

D. Program OpenPuff

Sedangkan program OpenPuff dikembangkan oleh perusahaan *Cosimo Oliboni*, dengan menggunakan konsep TrueCrypt yang merupakan penggunaan konsep chiper dengan gabungan antara AES, *Serpent*, dan *Twofish*. Sedangkan pada fungsi hash *kriptografi* menggunakan RIPEMD- 160, SHA-512, dan *Whirlpool*. TrueCrypt mendukung konsep yang disebut *deniability*, dengan memungkinkan satu "Volume tersembunyi" yang akan dibuat dalam volume lainnya. Selain itu, versi Windows TrueCrypt memiliki kemampuan untuk membuat dan menjalankan sistem operasi tersembunyi dan terenkripsi yang keberadaan mungkin tidak terdeteksi.

E. Analisis Matematis

Pada penerapan konsep integrasi antara *Steganografi* DCS dengan *Kriptografi* algoritma *Blowfish* agar menghasilkan model anti forensik untuk keamanan ganda dapat kita sampaikan dalam persamaan permodelan matematis.

Konsep utama yang membangun sebuah metode *steganografi* adalah pemilihan *cover medium,* algoritma *embedding* dan *extracting,* serta manajemen *stego key*. Secara matematis skema steganografi sudah dapat ditentukan. Anggap K_s adalah *stego key* dari susunan *K*, semua kunci rahasia stego. *M* adalah susunan semua pesan yang dapat disisipkan, dan ***c*** adalah susunan semua cover medium. Skema steganografik dibentuk oleh dua pemetaan yaitu, pemetaan penyisipan atau *Emb*, dan pemetaan ekstraksi atau *Ext* dengan persamaan sebagai berikut:

$$Emb : \boldsymbol{c} \times K \times M \rightarrow C$$
$$Ext : \boldsymbol{c} \rightarrow M_1 \quad \text{.....(1)}$$

Sehingga $Ext(Emb(\boldsymbol{c}, K_s, \mathbf{m})) = \mathbf{m}$ untuk semua $\boldsymbol{c} \in C$, $K_s \in K$, and $\mathbf{m} \in M$.

$\mathbf{s} = Emb(\boldsymbol{c}, K_s, \mathbf{m})$ disebut *stego work*.

Banyaknya teknik dalam *steganografi* menyebabkan diperlukan adanya pengelompokan atas jenis-jenisnya. Pengelompokan ini diharapkan akan memudahkan pengguna *steganografi* untuk memilih teknik yang sesuai dan pembuat *steganografi* untuk mengembangkan teknik-teknik baru dengan lebih terarah.

F. Uji Kapasitas

Uji kapasitas yang akan dilakukan adalah dengan melakukan pengujian menggunakan sample data yang dilakukan secara berulang-ulang dengan menggunakan tabel pengujian untuk mendapatkan hasil dari tingkat kapasitas yang dapat ditampung pada sebuah image *steganografi* dan juga maksimal kapasitas yang dapat dicapai oleh file target. Selain itu dalam melakukan proses penyisipan juga dilakukan penghitungan waktu proses dalam melakukan satu proses penyisipan.

Tabel 1. Hasil Uji Proses Kapasitas Penyisipan menggunakan sample file JPG

Nama Program	Ukuran Images Asli (Byte)	Alokasi Kapasitas Stego (%)	Ukuran File (Byte)	Alokasi Penggunaan Byte (%)	Alokasi Sisa Byte (%)
StegoBlow	15192056	12,50	30656	1.61	98.39
Stego PNG	15192056	8,63	30656	2.34	97.66
OpenPuff	15192056	1,15	30656	7.03	92.97

Dari tabel 1. dapat diketahui bahwa sample data yang digunakan adalah jenis file gambar berformat JPG dengan kapasitas ukuran 4 kb, dari hasil yang telah diujikan maka dapat kita lihat bahwa dengan alokasi peggunaan byte yang lebih besar berbanding dengan ukuran data yang disisipkan maka program StegoBlow mendapatkan hasil yg lebih efisien dari dari program Stego PNG dan OpenPuff karen masih memiliki alokasi sisa byte tempat penyimpanan sebesar 98,39%.

Perhitungan untuk Alokasi penggunaan byte dihasilkan dari pencarian nilai FN_{max} terlebih dahulu dengan menggunakan sampel data atau nilai *F* berbanding dengan ukuran image asli atau nilai *B* menggunakan percobaan program dari $F_1 - FN_{max}$, Setelah hasil FN_{max} diketahui maka dapat menghitung alokasi penggunaan byte berdasarkan persamaan matematika sebagai berikut:

AH = Alokasi penggunaan byte
FN_{max} = Maksimal ukuran file konten digital
F = Ukuran file konten digital
B = Ukuran image asli
E = Alokasi kapasitas stego

$$F(F1, \dots, FN_{Max}) = AH = \frac{F}{FNmax} \; x\; 100\% \text{.....(2)}$$

Setelah hasil perhitunggan alokasi penggunaan byte atau nilai *AH* maka kita dapat menghitung alokasi kapasitas stego dengan nilai *E* yang ada dengan persamaan sebagai berikut:

$$E = \frac{FNmax}{B} \; x\; 100\% \text{ .....(3)}$$

Hasil persamaan matematika diatas merupakan nilai kunci sebagai batas nilai maksimal sebuah konten digital dapat disisipkan kedalam sebuah image dan nilai maksimal tersebut dapat menjadi prediksi apakah sebuah kontek digital nilai *F* berhasil disisipkan atau tidak, jika konten digital atau nilai *F* yang disisipkan melebihi nilai *E* atau alokasi kapasitas stego maka dipastikan proses akan mengalami kegagalan sesuai dengan algoritma baku yaitu

If (F ≤ E) then F = Succes; else F = Fail;

Dari hasil uji kapasitas yang telah dilakukan berdasarkan maka dapat disimpulkan dengan menggunakan uji perbandingan program menggunakan penilaian tabel seperti pada tabel 2. dibawah ini.

Tabel 2. Uji Perbandingan Program *Steganografi*

Nama Program	Alokasi Kapasitas Stego (%)	Metode Steganografi	Keamanan Data	Alat untuk Extraksi data
StegoBlow	12,50	LSB + DCS	Auto	Ada

Stego PNG	8,63	LSB	Manual	Ada
OpenPuff	2,87	LSB	Manual	Ada

Pada uji perbandingan program *steganografi* seperti yang tercatat pada tabel 1. dapat dilihat bahwa program StegoBlow yang menggunakan konsep integrasi antara *Steganografi* DCS dengan *Kriptografi* algoritma *Blowfish* unggul dalam penerapan alokasi kapasitas data yang dapat disisipkan kedalam image. Dari hasil uji standar alokasi kapasitas yang ada pada metode Steganografi yang digunakan beberapa program menggunakan nilai ambang batas 10% sedangkan pada teknik DCS mampu meningkatkan efisiensi alokasi kapasitas yang ada hingga 2,5% hingga didapatkan total alokasi sebesar 12,5%.

G. Uji Ketahanan

Pada proses ini akan dilakukan pengujian ketahanan sistem (*robustness*) yang ada yaitu hasil image dari proses *steganografi* akan dimanipulasi hasilnya dengan menggunakan beberapa tahap uji proses untuk membuktikan bahwa setelah melalui uji proses tersebut konten digital yang terkandung dalam image *steganografi* apakah masih dapat dilakukan proses ekstrasi hingga mendapatkan hasil konten digital sesunguhnya ataupun proses yang ada mengalami kegagalan.

Proses pengujian yang dilakukan melalui beberapa proses antara lain, proses *cropping*, *rotate*, *resize*, *adjustment contras*, *convert* BMP ke JPG dan *split* atau memecah sebuah image menjadi 2 bagian. Dalam proses pengujian ini menggunakan hasil image proses *Steganografi* dari program StegoBlow yang merupakan implementasi dari metode *Steganografi* DCS dengan *Kriptografi Blowfish* dan hasil image *steganografi* dari program Stego PNG serta hasil image *steganografi* dari program OpenPuff.

Tabel 3. Uji Proses Ketahanan Sistem program StegoBlow

Uji Proses	Ukuran Image Setelah Proses (Byte)	Proses Restore	Hasil Proses Ekstraksi	Keterangan
Cropping	13789496	Gagal	Gagal	Menghilangkan bagian image yg lain
Rotate	15192056	Berhasil	Berhasil	0° ke 180°
Resize	12000056	Berhasil	Gagal	Resize lebar dari 2250 px menjadi 2000 px
Adjustments	15192056	Berhasil	Gagal	Adjusements Contrast +30
Convert	9392201	Berhasil	Berhasil	BMP ke JPG Max Quality
Split	7596056	Berhasil	Berhasil	Split menjadi 2 bagian

Dari hasil pengujian ketahanan sistem (*robustness*) seperti pada tabel 3. dapat dilihat bahwa program StegoBlow atau metode *Steganografi* teknik DCS masih unggul terhadap kedua program lainnya karena berhasil dalam mengembalikan proses ektraksi walaupun telah mengalami proses manipulasi antara lain:

1. *Rotate* (rotasi) adalah suatu proses untuk mengubah posisi gambar sesuai dengan derajat kemiringan yang akan ditentukan. Proses ini tidak menimbulkan kerusakan pada konten digital didalamnya.
2. *Convert* (konversi) adalah proses perubahan data dari format bitmap atau BMP ke JPG maupun sebaliknya. Pada proses tersebut jika dilakukan maka dapat mengakibatkan perubahan parameter nilai warna yang terkandung pada gambar karena adanya proses kompresi.
3. *Split* (Pembagian) adalah proses perubahan pembagian atau memecah sebuah gambar menjadi dua bagian atau lebih. Hal tersebut jika dilakukan maka mengakibatkan perubahan parameter nilai warna yang terkandung pada gambar karena adanya proses pembagian.

Proses-proses yang mengalami kegagalan pada program StegoBlow diantaranya adalah:

1. *Cropping* merupakan proses menghilangkan sebagian image yang ada sehingga nilai paremeter-parameter dari LSB yang ada menjadi berubah atau termodifikasi menyebabkan konten digital yang didalamnya mengalami kerusakan stuktur.
2. *Resize* merupakan proses merubah luas bidang image menjadi lebih besar atau lebih kecil dari ukuran aslinya, dalam hal ini dengan mengubah ukuran dapat mengakibatkan pergeseran nilai warna dan LSB yang ada sehingga dengan berubahnya nilai parameter tersebut juga mengubah konten digital yang ada didalamnya.
3. *Adjustment* merupakan proses mengubah nilai toleransi pencahayaan dalam hal ini yang digunakan adalah nilai kontras. Dengan mengubah nilai kontras yang ada juga sama dengan proses kegagalan yang lainnya yaitu berubahnya nilai parameter warna yang ada, dengan perubahan tersebut juga mengakibatkan kandungan konten digital yang ada mengalami perubahan.

H. Uji Deteksi Anti Steganografi

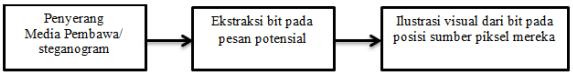
Program *XStegSecret Beta v0.1* yang dikenal dengan StegSecr*et* merupakan proyek steganalisis berkonsep *open source* (GNU / GPL) yang dikembangkan oleh Alfonso Muñoz dengan konsep memungkinkan untuk mendeteksi informasi yang tersembunyi dalam media digital yang berbeda. StegSecret adalah program steganalisis multiplatform berbasis java yang memungkinkan deteksi informasi yang tersembunyi dengan menggunakan metode *steganografi* yang paling dikenal dan mampu mendeteksi EOF, LSB, DCTs dan teknik lainnya. Konsep StegSecret ini adalah untuk mengumpulkan, untuk menerapkan dan untuk memudahkan penggunaan teknik steganalisis, terutama pada media digital, seperti gambar, audio dan video. Selain itu bertujuan untuk memperingatkan tentang ketidakamanan beberapa alat

steganografi dan algoritma *steganografi* yang tersedia di Internet.

Program StegSceret mempunyai beberapa konsep uji deteksi *steganografi* atau steganalisis yaitu:

1. Konsep Uji Hasil Visual Attack

Ide dari konsep *Visual Attack* atau serangan visual adalah untuk menghapus semua bagian dari gambar yang meliputi pesan. Dengan konsep tersebut maka mata manusia sekarang dapat membedakan apakah ada pesan potensial atau masih hanya ada konten gambar. Proses penyaringan dari serangan visual tergantung pada perkiraan utilitas *steganografi*, dan memiliki struktur sebagai berikut:



Gambar 5. Konsep Visual Attack

Dari Gambar 5. konsep diatas dapat dilihat perbandingan denggan menggunakan program *StegSecret* dengan menggunakan metode serangan visual berdasarkan *Byte Attack* pada LSB level 0 hasilnya adalah seperti pada gambar dibawah ini.

Gambar 6. Analisis Visual Attack LSB Level 0 hasil dari program StegoBlow menggunakan program StegSecret, Kiri gambar original dan Kanan gambar Stego

Dari hasil yang didapatkan pada analisis serangan visual sesuai dengan gambar 6. maka didapatkan tidak ada perbedaan atau perubahan konsep distorsi warna, yang terdapat pada LSB level 0 sampai dengan LSB level 7 merupakan bagian dari level terendah sehingga pada hasil yang didapatkan pada tingkatan nyata pada mata manusia sangat tidak terlihat perbedaannya.

Hasil yang didapatkan melalui uji serangan visual seperti yang tampil dari Gambar 6. dapat diketahui bahwa seluruh proses *Steganografi* tidak terdeteksi dengan analisis *Enhanced* LSBs yang ada dikarenakan baik program StegoBlow, Stego PNG dan OpenPuff menggunakan sampel data sebesar 4 kb pada proses pengujiannya sehingga hasil *noisy* yang ditimbulkan pada proses *Steganografi.*

2. Konsep Uji Hasil RS-analisis

Pada pengujian menggunakan program *StegSecret* menggunakan sample data image steganografi dengan image yang masih original (Asli) didapatkan hasil analisa menggunakan RS-analisis seperti pada Gambar 7.

```
[RS Analysis] (RGB-disjoint groups)
 ==================================

Occupation [red channel]: 5.75309% = 0.05753 bpp (bits per pixel)
[Red] Approximate Size: 109218.90302 bytes.

Occupation [green channel]: 12.22067% = 0.12221 bpp (bits per pixel)
[Green] Approximate Size: 232001.85263 bytes.

Occupation [Blue channel]: 2.71604% = 0.02716  bpp (bits per pixel)
[Blue] Approximate Size: 51562.33333 bytes.


Occupation of the total possible: 6.8966% = 0.06897 bpp
APPROXIMATE SIZE OF HIDDEN INFORMATION (media):
130927.69633 bytes
```

Gambar 7. Hasil RS Analisis pada Image *Steganografi* hasil dari program StegoBlow

Adapun perhitungan untuk menentukan berapakah besar dari kemungkinan file target yang telah disisipkan kedalam media image *steganografi* adalah dengan menggunakan model perbandingan yaitu hasil RS-analisis pada image *steganografi* dikurangkan dengan hasil RS-analisis pada image original dengan perhitungan sebagai berikut:

$$RS_{stego} - RS_{original} = F_{target} \quad \text{.....(4)}$$

130927.69633 – 127050.95501 = 3876.74132 byte

Hasil nilai diatas merupakan kemungkinan besar dari file target yang disisipkan pada media image *steganografi* dapat kita bandingkan dengan tabel pengujian maka file yang terdeteksi dengan RS-Analisis ada dengan ukuran 3876 byte.

Hasil perngujian dari program StegoBlow, Stego PNG dan OpenPuff berdasarkan perbandingan dari perhitungan RS Analisis menggunakan sampel data sebesar 4 kb terlihat hasilnya seperti yang ada di Gambar 3.4, dapat kita simpulkan bahwa setiap image *steganografi* berisikan konten digital yang dihasilkan dari proses penyisipan menggunakan teknik LSB maka mempunyai nilai okupasi RGB diatas 10% dengan nilai tersebut dapat dinyatakan terdeteksi, berbeda dengan program StegoBlow yang menggunakan teknik LSB dan DCS maka hasil yang didapatkan hanya terpaut selisih sebesar 1% dari gambar original, maka dengan nilai toleransi antara 0%-1% maka dapat dikatakan tidak terdeteksi. Selain itu besar kecil dari nilai okupasi pada sebuah konten digital yang disisipkan juga dipengaruhi dari nilai perbandingan ukuran file / konten digital dengan alokasi kapasitas stego yang ditetapkan dari program yang digunakan, semakin besar alokasi kapsitas stego dan ukuran file / konten digital yang disisipkan maka nilai okupasi yang dihasilkan dari RS analisis akan semakin besar juga, begitu pula dengan kebalikannya.

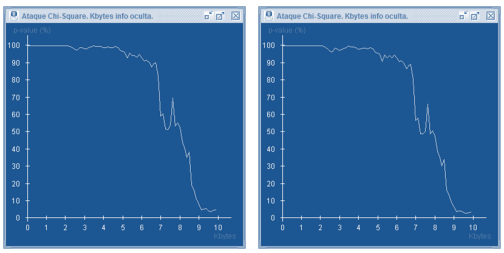
3. Konsep Uji Hasil Chi Square

Metode ini bekerja dengan melakukan perbandingan uji *chi-square* antara dua buah statistik distribusi frekuensi, yang pertama adalah statistik pada gambar yang dicurigai mengandung pesan tersembunyi, dan yang kedua adalah statistik yang diprediksi akan dimiliki oleh gambar tersebut apabila disisipi pesan. Apabila kedua statistik ini sama, atau

terdapat suatu bagian yang sama, maka kemungkinan besar terdapat suatu pesan dalam gambar.

Apabila distribusi frekuensi memiliki perbedaan yang signifikan maka distribusi dari LSB tidak bersifat acak, yang berarti kemungkinan besar tidak terdapat pesan rahasia. Sedangkan apabila kedua frekuensi ini tidak memiliki perbedaan yang signifikan maka distribusi LSB mendekati acak, sehingga kemungkinan besar terdapat pesan rahasia yang telah disisipkan pada LSB citra.

Sedangkan metode Chi-Square dikemukakan oleh Bimo [2], bahwa Chi-Square digunakan untuk melihat ketergantungan antara variabel bebas dan variabel tergantung berskala nominal dan ordonal. Prosedur uji Chi-Square menabulasi satu atau lebih variabel ke dalam kategori-kategori dan menghitung angka statistik Chi-Square. Untuk satu variabel dikenal sebagai uji keselarasan atau *goodness of fit test* yang berfungsi untuk membandingkan frekuensi yang diamati dan di harapkan ke dalam masing-masing ketegori untuk satu pengguna. Jika terdiri dari dua variabel dikenal dengan independensi yang berfungsi untuk hubungan dua variabel. Seperti sifatnya, uji Chi-Square dikelompokkan ke dalam statistik uji non parametrik.



Gambar 8. Hasil uji metode Chi-Square pada program StegoBlow, sebelah kiri adalah image Steganografi dan sebelah kanan adalah image yang original (Asli)

Dari hasil uji metode Chi-Square oleh hasil dari program StegoBlow, Stego PNG dan OpenPuff sesuai dengan Gambar 8, dapat kita lihat bahwa program StegoBlow mempunyai keunggulan untuk dapat tidak terdeteksi atau hampir tersamarkan pada metode Chi-Square. Hasil nilai persentase frekuensi gambar yang telah disisipi pesan sebagai *Pair of Value* (P-value) berbanding dengan kapasitas data gambar (Kbytes), maka hasil statistik Chi-Square yang dihasilkan dari teknik DCS berupa kurva yang sama persis dengan kurva image original, sebaliknya dengan program Stego PNG menghasilkan kurva yang linier pada hasil image steganografi nya sangat berbeda jauh dengan image original, sedangkan pada hasil program OpenPuff hasil yang didapatkan juga hampir menyerupai dengan image original tetapi masih berbeda dalam hasil kurvanya, maka hasil dari kedua progam tersebut dapat dinyatak terdeteksi pada pengujian Chi-Square.

IV. KESIMPULAN

Berdasarkan hasil yang didapatkan dari proses implementasi, hasil dan pembahasan, maka penelitian tentang integrasi metode *Steganografi* DCS pada image dengan *Kriptografi Blowfish* sebagai model Anti Forensik untuk keamanan ganda konten digital dapat ditarik beberapa kesimpulan:

1. Standar keamanan ganda pada permodelan anti forensik dapat dirancang menggunakan integrasi antara metode *Steganografi* teknik DCS dengan *Kriptografi* algoritma *Blowfish,* dan hasil rancangangan dapat diimplementasikan kedalam bentuk aplikasi StegoBlow.
2. Dari analisis yang telah dilakukan dapat terbukti bahwa implementasi dari program StegoBlow mendapatkan hasil pengujian dengan baik untuk penyebaran pesan rahasia yang lebih aman dan tidak terdeteksi dalam penerapan standar keamanan ganda pada permodelan anti forensik.
3. penerapan standar keamanan ganda pada integrasi antara metode *Steganografi* teknik DCS dengan *Kriptografi* algoritma *Blowfish* pada permodelan anti forensik, dapat mengakibatkan pembuktian proses forensika digital menjadi lebih sulit.

DAFTAR PUSTAKA

[1] Andi. (2003). *Memahami Model Enkripsi dan Security Data*. Yogyakarta, Andi Offset.

[2] Bimo, Suseno. (2013). Analisis Chi-Square, website pada http://www.statistikolahdata.com/2013/04/analisis-chi-square.html, di akses pada 18 September 2013.

[3] Irawan, Anton Nugroho. (2004). Study Dan Implementasi Algoritma Blowfish Untuk Security Dokumen Elektronik. Yogyakarta, Fakultas Teknologi Industri, Universitas Islam Indonesia.

[4] Ohmacht, Holger. Stegano Project, website pada http:// www.holger-ohmacht.de, diakses pada 23 February 2004.

[5] Provos, Niels. & Honeyman, Peter. (2003). Hide and Seek: An Introduction to Steganography. IEEE Computer Society, ISSN: 1540-7993, Volume 3.

[6] Schneier, Bruce. (2001). Descryption of a New Variable-Length Key 64/128-Bit Block Cipher (Blowfish).

[7] Suhono, H. Supangkat. & Kuspriyanto, Juanda. (2000). Watermarking sebagai Teknik Penyembunyian Hak Cipta pada Data Digital. Jurnal dari Departemen Teknik Elektro, Institut Teknologi Bandung, Volume 6, No. 3.

[8] Supani, Ahyar. (2002). Sistem Keamanan File Dan Folder Data Menggunakan Algoritma Blowfish. Bandung.

[9] Wijaya, Ermadi Satriya. & Prayudi, Yudi. (2004). Konsep Hidden Message Menggunakan Teknik Steganografi Dynamic Cell Spreading. Jurnal Media Informatika, ISSN: 0854-4743, Volume 2, No. 1, PP 23-38.

[10] Zöllner, J. et al. (2004). Modeling the Security of Steganographic System. Journal from Dresden University of Technology.