Perbandingan Ketahanan Algoritma LSB dan F5 dalam Steganografi Citra

Ricardo Pramana Suranta / 13509014[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13509014@itb.ac.id

***Abstract*—Steganografi adalah ilmudan seni untuk menyembunyikan keberadaan suatu pesan pada suatu *steganogram* (*carrier*), yang diaplikasikan baik pada media analog/konvensional maupun digital. Salah satu kriteria steganografi yang baik adalah ketahanan *(robustness)* steganogram hasil penyisipan terhadap berbagai modifikasi terhadap steganogram. Metode LSB dan algoritma F5 memiliki ketahanan yang kurang atas modifikasi citra (untuk *steganogram* berbentuk citra).**

***Index Terms*— LSB, F5, steganografi, ketahanan**

I. PENDAHULUAN

Steganografi adalah ilmu dan seni dalam menyembunyikan (*embedding*) informasi dengan cara menyisipkan pesan rahasia di dalam pesan lain, sehingga tidak dicurigai dan dapat dibaca / dipahami oleh pihak yang tidak diinginkan. Steganografi sendiri sudah ada sejak abad ke 18, yang berkembang pesat di kalangan pujangga, pelajar, militer, dan kriminal.

Steganografi memiliki beragam bentuk, dan hal inilah yang membuat steganografi sendiri menjadi sebuah "seni". Pada zaman lampau, steganografi hadir dalam berbagai bentuk, seperti surat yang ditulis dengan tinta yang tidak kelihatan, tato pada kepala seorang budak yang hanya bisa dibaca dengan membotak kepala budak tersebut, surat yang terlihat "biasa" namun menyimpan pesan rahasia pada susunan hurufnya, dan banyak bentuk lainnya. Saat ini, steganografi sendiri banyak diaplikasikan pada data-data digital, dimana pesan yang dapat berupa data apapun disembunyikan kedalam berkas (*file*) lainnya yang umum digunakan dan sama sekali tidak terlihat mencurigakan, seperti berkas gambar, musik, dan video.

Terdapat banyak algoritma untuk menerapkan steganografi pada data digital, oleh karena inilah, steganografi disebut sebagai "seni". Selain itu, steganografi sendiri juga disebut sebagai "ilmu", oleh karena terdapat aspek-aspek tertentu yang dapat diukur dan perlu dipenuhi untuk menentukan apakah sebuah algoritma yang digunakan dalam steganografi sudah cukup baik atau tidak.

Karena bertujuan utama untuk menyembunyikan pesan, faktor ketahanan (*robustness*) pesan terhadap perubahan yang dapat terjadi tidak terlalu diperhatikan. Pada makalah ini, penulis hendak membandingkan ketahanan dari beberapa algoritma steganografi yang umum (yang menggunakan LSB) dan algoritma yang diklaim lebih kuat daripada yang lainnya, yakni F5.

II. DASAR TEORI

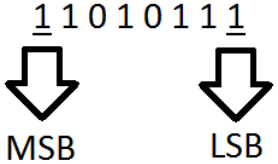
2.1 Steganografi

Beberapa kriteria yang menentukan bagus tidaknya sebuah algoritma steganografi adalah :

1. *Fidelity*
   Setelah penyisipan pesan rahasia, mutu steganogram (media pembawa pesan) tidak jauh berubah dari yang asli, sehingga tidak disadari adanya pesan rahasia oleh pihak selain pengirim dan penerima pesan.
2. *Robustness*
   Apabila steganogram mengalami operasi manipulasi (mis. operasi pengolahan citra seperti penajaman dan pemotongan untuk steganogram berbentuk citra), pesan yang disembunyikan tidak rusak.
3. *Recovery*
   Pesan yang disembunyikan harus dapat diekstraksi dari steganogram.

2.2 Metode Modifikasi LSB (Least Significant Bit)

LSB (*least significant bit*) adalah bit yang paling sedikit pengaruhnya dalam suatu *byte*. Selain LSB, terdapat pula MSB (*most significant bit*), yang merupakan bit yang paling besar pengaruhnya dalam suatu *byte*.



***Gambar 2.1** Posisi LSB dan MSB pada sebuah byte big-endian*

Pada steganografi citra dengan metode LSB, LSB pada

suatu steganogram citra digunakan untuk menampung bit-bit dari pesan. Untuk memperkuat penyembunyian pesan, *byte* yang diambil dari citra untuk disisipi tidaklah urut. Untuk menentukan *byte* mana dari suatu citra yang akan disisipi bit dari pesan, digunakan *pseudo-random-number-generator* (PRNG), yang membangkitkan posisi *pixel* dari citra yang hendak disisipi oleh bit pesan, sesuai dengan kunci rahasia.

Setiap citra digital terdiri atas sejumlah *pixel*. Sebuah *pixel* memiliki sebuah ukuran, misalkan 8-bit, 24-bit, 32-bit, dsb. Tiap bit dalam sebuah *pixel* menyatakan derajat keabuan dari *pixel* tersebut. Pada sebuah citra 24-bit (*true-color*), tiap *byte* (1 *byte* = 8 bit) merepresentasikan salah satu dari komponen RGB (*red-green-blue*) *color model*, dimana *byte* pertama merepresentasikan komponen *red*, *byte* kedua merepresentasikan komponen *green*, dan *byte* ketiga merepresentasikan komponen *blue*.

2.3 Metode F5

F5 adalah sebuah metode / algoritma yang diajukan oleh Andreas Westfeld dari Technische Universitat Dresden, Institute for System Architecture, Jerman. F5 adalah perbaikan dari kedua algoritma yang telah beliau ajukan sebelumnya, yakni F3 dan F4. Algoritma ini tidak menggunakan metode LSB, namun ia menghitung penyebaran *byte-byte* dari steganogram citra (dalam hal ini citra berformat JPEG (*Joint Photographic Experts Group*)) baik positif maupun negatif, namun bukan 0, menyisipkan (*embedding*) bit dari pesan rahasia ke beberapa *byte* tersebut dengan operasi XOR, lalu mengurangi nilai (*decrement*) dari *byte* tersebut, baik yang disisipi oleh bit dari pesan rahasia maupun tidak. Secara rinci, berikut adalah langkah-langkah dari algoritma F5 :

1. Memulai kompresi JPEG, hinga penghitungan koefisien citra selesai.
2. Bangkitkan secara acak sebuah angka yang kuat (dari sudut kriptografi). Angka tersebut juga dapat dibangkitan secara acak berdasarkan kunci rahasia yang dimasukkan.
3. Menginstansiasikan sebuah permutasi, dengan dua parameter, yakni angka hasil pembangkitan acak tahap (2) dan koefisian penghitungan citra dari tahap (1)
4. Tentukan nilai parameter *k* dari kapasitas dari steganogram dan panjang dari pesan rahasia.
5. Hitung panjang *code word* (*byte* penampung bit dari pesan rahasia), dimana $n = 2^k - 1$, dengan *n* adalah panjang *code word*, dan *k* dari tahap (4).
6. Lakukan *embedding* pesan rahasia dengan (1,*n*,*k*) *matrix encoding* :
    a. Penuhi sebuah *buffer* dengan *n* koefisien selain nol.
    b. Lakukan *hashing* terhadap buffer ini.
    c. Tambahkan *k* bit selanjutnya dari pesan kedalam nilai *hash* (tiap bit, dengan operasi XOR)
    d. Apabila hasil dari (c) adalah 0, *buffer* tersebut tidak diubah. Selain itu, hasil dari (c) pastilah index *buffer*, nilai absolut dari elemen tersebut dikurangi sebanyak satu (*decrementing*).
    e. Lakukan ujicoba untuk pengecilan nilai (*shrinkage*) yang dapat menghilangkan nilai dari pesan yang dimasukkan kedalam *buffer*, misalkan apakah kita menghasilkan sebuah nilai nol. Apabila benar demikian, maka kita harus menyesuaikan *buffer* tersebut (mengeliminasi kemungkinan nol tersebut dengan mengambil koefisien selain nol yang lain). Bila tidak terjadi pengecilan nilai, maka lanjutkan pemeriksaan ke koefisien baru yang berada tepat setelah *buffer* saat ini. Bila masih ada bit dari pesan rahasia yang belum dimasukkan, ulangi tahap (a).
7. Lanjutkan kompresi JPEG (*Huffman coding*, dsb.)

F5 memberikan perbaikan terhadap algoritma F3 yang hanya memperhitungkan *byte* yang bernilai genap, sedangkan citra dengan format JPEG memiliki lebih banyak *byte* bernilai ganjil daripada genap (walaupun pengurangan nilai (*decrement*) dari bilangan genap akan menghasilkan bilangan ganjil), yang dapat terdeteksi bila dilakukan serangan statistic, dan penyebaran "lokasi" penyisipan bit pesan yang jauh lebih merata daripada F4. Oleh pembuatnya sendiri, F5 dikatakan sebagai algoritma yang "tahan terhadap serangan statistik dan visual, namun menyediakan kapasitas yang cukup besar", dengan *payload* sekitar 12% dari gambar.

k	n	change density	embedding rate	embedding efficiency
1	1	50.00%	100%	2
2	3	25.00%	66.67%	2.67
3	7	12.50%	42.86%	3.43
4	15	6.25%	26.67%	4.27
5	31	3.12%	16.13%	5.16
6	63	1.56%	9.52%	6.09
7	127	0.78%	5.51%	7.06
8	255	0.39%	3.14%	8.03
9	511	0.20%	1.76%	9.02

Tabel 2.1 *Hubungan antara kepadatan perubahan (change density)dan laju penyisipan (embedding)pada suatu citra, sesuai dengan k (bit hasil ekstraksi code word dari fungsi hash) dan n (panjang code word)*

III. PERBANDINGAN KETAHANAN ALGORITMA LSB DAN F5 DALAM STEGANOGRAFI CITRA

Dalam melakukan perbandingan ini, penulis menggunakan citra dengan format JPEG 24-bit (RGB), sehingga dapat digunakan baik untuk algoritma F5 dan metode LSB. Untuk pengukuran algoritma F5, penulis menggunakan perangkat lunak dengan bahasa Java yang sudah disediakan oleh Andreas Westfeld, yang dapat diunduh pada http://code.google.com/p/f5-steganography/, dan untuk pengukuran metode LSB, penulis menggunakan salah satu perangkat lunak yang dibuat oleh teman penulis, Abraham Giuseppe Andrea P E S (13509040), Daniel Widya Suryanata (13509083), dan Auliya Unnisa Fitri S (13509067) untuk Tugas Besar pertama IF3058 Kriptografi. Perangkat lunak tersebut menggunakan metode LSB untuk penyisipan data kedalam steganogram citra.

Berikut adalah citra yang digunakan penulis :

Gambar 3.1 *Aula Barat ITB, 640 x 426 pixel, 39.9 KB*

Gambar 3.2 *Lena,512 x 512 pixel, 67.8 KB*

Untuk pesan yang disisipkan kedalam citra, penguji menggunakan empat buah pesan, masing-masing berukuran 11 *bytes*, 6.81 *kilobytes*, 12.415 *kilobytes*, dan 20.95 *kilobytes*. Isi berkas uji pertama adalah sebuah kata, yakni "steganografi", isi berkas uji kedua dan ketiga adalah pengulangan dari kata-kata dibawah ini :

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Isi berkas uji keempat adalah pengulangan dari artikel berikut :

Malin Kundang

Pada suatu hari, hiduplah sebuah keluarga di pesisir pantai wilayah Sumatra. Keluarga itu mempunyai seorang anak yang diberi nama Malin Kundang. Karena kondisi keluarga mereka sangat memprihatinkan, maka ayah malin memutuskan untuk pergi ke negeri seberang.

Besar harapan malin dan ibunya, suatu hari nanti ayahnya pulang dengan membawa uang banyak yang nantinya dapat untuk membeli keperluan sehari-hari. Setelah berbulan-bulan lamanya ternyata ayah malin tidak kunjung datang, dan akhirnya pupuslah harapan Malin Kundang dan ibunya.

Setelah Malin Kundang beranjak dewasa, ia berpikir untuk mencari nafkah di negeri seberang dengan harapan nantinya ketika kembali ke kampung halaman, ia sudah menjadi seorang yang kaya raya. Akhirnya Malin Kundang ikut berlayar bersama dengan seorang nahkoda kapal dagang di kampung halamannya yang sudah sukses.

Selama berada di kapal, Malin Kundang banyak belajar tentang ilmu pelayaran pada anak buah kapal yang sudah berpengalaman. Malin belajar dengan tekun tentang perkapalan pada teman-temannya yang lebih berpengalaman, dan akhirnya dia sangat mahir dalam hal perkapalan.

Banyak pulau sudah dikunjunginya, sampai dengan suatu hari di tengah perjalanan, tiba-tiba kapal yang dinaiki Malin Kundang di serang oleh bajak laut. Semua barang dagangan para pedagang yang berada di kapal dirampas oleh bajak laut. Bahkan sebagian besar awak kapal dan orang yang berada di kapal tersebut dibunuh oleh para bajak laut. Malin Kundang sangat beruntung dirinya tidak dibunuh oleh para bajak laut, karena ketika peristiwa itu terjadi, Malin segera bersembunyi di sebuah ruang kecil yang tertutup oleh kayu.

Malin Kundang terkatung-katung ditengah laut, hingga akhirnya kapal yang ditumpanginya terdampar di sebuah pantai. Dengan sisa tenaga yang ada, Malin Kundang berjalan menuju ke desa yang terdekat dari pantai. Sesampainya di desa tersebut, Malin Kundang ditolong

oleh masyarakat di desa tersebut setelah sebelumnya menceritakan kejadian yang menimpanya. Desa tempat Malin terdampar adalah desa yang sangat subur. Dengan keuletan dan kegigihannya dalam bekerja, Malin lama kelamaan berhasil menjadi seorang yang kaya raya. Ia memiliki banyak kapal dagang dengan anak buah yang jumlahnya lebih dari 100 orang. Setelah menjadi kaya raya, Malin Kundang mempersunting seorang gadis untuk menjadi istrinya.

Setelah beberapa lama menikah, Malin dan istrinya melakukan pelayaran dengan kapal yang besar dan indah disertai anak buah kapal serta pengawalnya yang banyak. Ibu Malin Kundang yang setiap hari menunggui anaknya, melihat kapal yang sangat indah itu, masuk ke pelabuhan. Ia melihat ada dua orang yang sedang berdiri di atas geladak kapal. Ia yakin kalau yang sedang berdiri itu adalah anaknya Malin Kundang beserta istrinya.

Malin Kundang pun turun dari kapal. Ia disambut oleh ibunya. Setelah cukup dekat, ibunya melihat belas luka dilengan kanan orang tersebut, semakin yakinlah ibunya bahwa yang ia dekati adalah Malin Kundang. "Malin Kundang, anakku, mengapa kau pergi begitu lama tanpa mengirimkan kabar?", katanya sambil memeluk Malin Kundang. Tetapi Kundang segera melepaskan pelukan ibunya dan mendorongnya hingga terjatuh. "Wanita tak tahu diri, sembarangan saja mengaku sebagai ibuku", kata Malin Kundang pada ibunya. Malin Kundang pura-pura tidak mengenali ibunya, karena malu dengan ibunya yang sudah tua dan mengenakan baju compang-camping. "Wanita itu ibumu?", Tanya istri Malin Kundang. "Tidak, ia hanya seorang pengemis yang pura-pura mengaku sebagai ibuku agar mendapatkan harta ku", sahut Malin kepada istrinya. Mendengar pernyataan dan diperlakukan semena-mena oleh anaknya, ibu Malin Kundang sangat marah. Ia tidak menduga anaknya menjadi anak durhaka. Karena kemarahannya yang memuncak, ibu Malin menengadahkan tangannya sambil berkata "Oh Tuhan, kalau benar ia anakku, aku sumpahi dia menjadi sebuah batu". Tidak berapa lama kemudian angin bergemuruh kencang dan badai dahsyat datang menghancurkan kapal Malin Kundang. Setelah itu tubuh Malin Kundang perlahan menjadi kaku dan lama-kelamaan akhirnya berbentuk menjadi sebuah batu karang.

Berikut adalah ujicoba penyisipan dengan metode LSB :

Gambar 3.3 *penyisipan data uji pertama kedalam gambar pertama (aula ITB) dengan metode LSB*

Pada gambar diatas, terlihat bahwa pesan dapat dimasukkan kedalam steganogram dengan baik. Penyisipan juga berjalan dengan baik pada pesan lainnya, dan pada steganogram kedua (gambar lena).

Berikut adalah ujicoba penyisipan dengan algoritma F5, tanpa kunci rahasia:

```
D:\Kuliah\Semester 6\IF3058 Kriptografi\Tugas\3. Paper 1\uji>java -jar f5.jar e
-e teskecil.txt aulaitb.jpg aulaitba.jpg
DCT/quantisation starts
640 x 426
got 414720 DCT AC/DC coefficients
one=32137
large=21111
expected capacity: 36858 bits
expected capacity with
default code: 4607 bytes (efficiency: 1.3 bits per change)
(1, 3, 2) code: 3071 bytes (efficiency: 1.6 bits per change)
(1, 7, 3) code: 1974 bytes (efficiency: 1.9 bits per change)
(1, 15, 4) code: 1228 bytes (efficiency: 2.3 bits per change)
(1, 31, 5) code: 740 bytes (efficiency: 2.7 bits per change)
(1, 63, 6) code: 433 bytes (efficiency: 3.2 bits per change)
(1, 127, 7) code: 238 bytes (efficiency: 3.5 bits per change)
Permutation starts
Embedding of 128 bits (12+4 bytes) using (1, 127, 7) code
48 coefficients changed (efficiency: 2.6 bits per change)
24 coefficients thrown (zeroed)
128 bits (16 bytes) embedded
Starting Huffman Encoding.
```

Gambar 3.4 *penyisipan data uji pertama kedalam gambar pertama (aula ITB) dengan algoritma F5*

Pada gambar diatas, terlihat bahwa dengan algoritma F5, gambar pertama memiliki estimasi kapasitas 36858 *bits*, atau 4.607 *kilobytes* – sekitar 11.5% dari ukuran berkas mula-mula. Hal ini menyebabkan pesan kedua, ketiga, keempat tidak dapat dimasukkan seluruhnya kedalam steganogram tersebut. Hal ini juga terjadi pada penyisipan citra kedua, yang memiliki kapasitas 41324 *bits*, atau 5.1655 *kilobytes*. Berikut adalah ekstraksi dari pesan uji keempat dari gambar Lena :

```
D:\Kuliah\Semester 6\IF3058 Kriptografi\Tugas\3. Paper 1\uji>java -jar f5.jar x
-e out.txt lenad.jpg
Huffman decoding starts
Permutation starts
393216 indices shuffled
Extraction starts
Length of embedded file: 12415 bytes
Default code used
Incomplete file: only 5194 of 12415 bytes extracted

D:\Kuliah\Semester 6\IF3058 Kriptografi\Tugas\3. Paper 1\uji>
```

Gambar 3.5 *eksttraksi data uji keempat dari gambar kedua (Lena) dengan algoritma F5*

Pesan yang disisipkan kedalam steganogram (baik yang waktu operasinya melebihi kapasitas ataupun tidak) tetap berhasil diekstraksi dengan baik. Berikut adalah steganogram hasil penyisipan pesan keempat pada kedua

gambar, dengan metode LSB dan algoritma F5 :

Gambar 3.6 *Steganogram citra Aula Barat ITB hasil penyisipan data uji keempat dari gambar pertama dengan metode LSB.*

Gambar 3.7 *Steganogram ctra Aula Barat ITB hasil penyisipan data uji keempat dari gambar pertama dengan algoritma F5.*

Gambar 3.8 *Steganogram citra Lena hasil penyisipan data uji keempat dari gambar pertama dengan metode LSB.*

Gambar 3.9 *Steganogram citra Lena hasil penyisipan data uji keempat dari gambar pertama dengan algoritma F5.*

Pengujian ketahanan yang dilakukan oleh penulis untuk kedua citra adalah penajaman (*contrast*) citra dengan aplikasi Microsoft Office 2012. Penulis menaikkan *contrast* dari citra sebanyak 10% untuk masing-masing steganogram. Pesan yang awalnya dimasukkan tidak dapat diekstrak kembali setelah pengubahan *contrast.*

Gambar 3.11 *Ekstraksi dari steganogram citra Aula Barat ITB hasil penyisipan data uji pertama dari gambar pertama dengan metode LSB. Pada saat ekstraksi, program berhenti berjalan (hang) karena tidak dapat mengekstraksi pesan yang disisipkan.*

```
D:\Kuliah\Semester 6\IF3058 Kriptografi\Tugas\3. Paper 1\uji>java -jar f5.jar x
-e out.txt aulaitbacontrast.jpg
Huffman decoding starts
Permutation starts
414720 indices shuffled
Extraction starts
Length of embedded file: 1141549 bytes
(1, 32767, 15) code used
Incomplete file: only 1 of 1141549 bytes extracted
```

Gambar 3.10 *Ekstraksi dari steganogram citra Aula Barat ITB hasil penyisipan data uji pertama dari gambar pertama dengan algoritma F5 terlihat bahwa informasi pesan yang sebenarnya disisipkan juga ikut berubah (yang sebenarnya complete, namun menjadi incomplete pada ekstraksi)*

IV. KESIMPULAN

Secara umum, dapat disimpulkan bahwa setelah steganogram hasil penyisipan metode LSB dan algoritma F5 tidak terlihat berbeda secara visual (di mata manusia). Hanya saja, steganogram hasil dari kedua cara tersebut tidak dapat melewati salah satu pengujian ketahanan *watermarking*, yakni perubahan *contrast.*

Penulis mengakui bahwa pengujian yang dilakukan belum cukup baik, oleh karena keterbatasan kemampuan penulis. Pengujian dapat menjadi lebih baik apabila menggunakan standar pengujian ketahanan yang umumnya digunakan pada bidang *watermarking* secara lengkap. Untuk melengkapi pengujian, dapat dicantumkan nilai PSNR dari metode LSB dan algoritma F5.

V. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Dr. Rinaldi Munir selaku dosen mata kuliah IF3058 Kriptografi, dan atas bantuan yang diberikan, serta kepada Abraham Giuseppe Andrea P E S, Daniel Widya Suryanata, dan Auliya Unnisa Fitri S atas izin yang diberikan untuk

menggunakan Tugas Besar pertama pada mata kuliah IF3058 Kriptografi mereka untuk makalah ini.

REFERENSI

[1] Munir, Rinaldi, *Diktat Kuliah IF5054 Kriptografi*, Penerbit Informatika : Bandung, 2005

[2] http://faculty.ksu.edu.sa/ghazy/Setganog/References/ref44.pdf, F5-A Steganographic Algorithm, High Capacity Despite Better Steganalysis, diakses pada 29 Februari 2012, pukul 08.00 WIB.

DAFTAR GAMBAR

[1] Gambar 3.1, Aula Barat ITB, diambil dari http://sun1.lib.itb.ac.id/expo/d/84-2/Aula+Barat+ITB, diakses pada 19 Maret 2012, pukul 12.00 WIB

[2] Gambar 3.2, Lena, diambil dari http://microblog.routed.net/wp-content/uploads/2006/11/lena.jpg, diakses pada 19 Maret 2012, pukul 12.00 WIB

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 30 Maret 2012

Ricardo Pramana Suranta / 13509014