Techno.COM, Vol. 15, No. 1, Februari 2016: 1-6

PENGAMANAN FILE MULTIMEDIA DENGAN METODE STEGANOGRAFI END OF FILE UNTUK MENJAGA KERAHASIAAN PESAN

Muslih[1], Eko Hari Rachmawanto[2]
[1,2]Fakultas Ilmu Komputer, Universitas Dian Nuswantoro Semarang
Jl. Nakula I, No. 5-11, Semarang, Kode Pos 50131, Telp. (024) 3515261, 3520165 Fax: 3569684
E-mail : muslih@dsn.dinus.ac.id [1], eko.hari@dsn.dinus.ac.id[2]

Abstrak

Makalah ini mengusulkan tentang pengamanan file multimedia dari serangan baik itu hacker maupun cracker dengan menggunakan metode steganografi. Dalam hal ini, algoritma yang digunakan adalah End Of File (EoF) dikarenakan algoritma ini dapat menampung kapasitas pesan yang cukup besar dengan tidak mempengaruhi kualitas dari file tempat pesan disembunyikan. Selain itu, algoritma ini dirasa cukup aman dari beberapa serangan dengan dibuktikan dari paper-paper yang telah ada sebelumnya. Berdasarkan dari beberapa hasil percobaan yang telah dilakukan dalam penelitian ini, metode steganografi untuk pengamanan file multimedia dengan algoritma EoF dapat mengamankan pesan serta dapat menjaga kerahasiaan isi data pesan dan juga tidak mempengaruhi kualitas file induk yang merupakan tempat untuk persembunyian pesan.

Kata Kunci: *Steganografi, End of File (EoF), file, multimedia.*

Abstract

This paper proposes about securing multimedia files from both the hacker attack and cracker using steganography. In this case, the algorithm used is the End Of File (EoF) algorithm because it can accommodate a large enough capacity to message does not affect the quality of the file where the message is hidden. Additionally, these algorithms are considered safe from attacks with evident from the papers that have been there before. Based on some of the results of experiments that have been conducted in this study, steganography method for securing multimedia files with EoF algorithm can secure messaging as well as to maintain the confidentiality of the data content of the message and does not affect the quality of the master file which is a place for hiding messages.

Keywords: *Steganoraphy, End of File (EoF), file, multimedia.*

1. PENDAHULUAN

Berkembangnya dunia teknologi yang pesat sekarang ini, maka akan berdampak pada meningkatnya kemampuan sumber daya manusia untuk mengoperasikan sebuah aplikasi yang ada di dalam komputer sehingga hal ini memungkinkan orang dapat menggunakan kemampuan tersebut secara positif maupun secara negatif.
Informasi merupakan hal sangat penting dalam komunikasi. Apabila salah dalam penyampaian informasi, maka akan salah pula dalam penerimaan informasi sehingga dapat dipastikan bahwa dalam penyampaian informasi harus sesuai dengan apa yang nantinya akan diterima oleh penerima informasi tersebut.

Perkembangan teknologi saat ini mengakibatkan pengiriman informasi dapat dilakukan dari jauh dengan bantuan internet. Pada saat tertentu informasi yang dikirimkan tidak ditujukan kepada semua orang namun ditujukan hanya kepada orang atau badan usaha tertentu. Seiring dengan

perkembanagn teknologi tersebut, ancaman terhadap keamanan informasi yang dibutuhkan semakin besar, terutama untuk informasi yang dirahasiakan tersebut. Berbagai ancaman dalam dunia maya seperti hacker, cracker, carder membuat orang khawatir akan keamanan informasi yang dikirimnya[1]. Kekhawatiran inilah yang membuat pengiriman informasi sedikit terhambat, sedangkan informasi tersebut sangat penting bagi orang-orang tertentu.

Oleh karena itu, diperlukan sebuah tool untuk mengamankan informasi yang nantinya akan dikirimkan kepada penerima. Salah satu cara yang digunakan untuk mengamankan data yaitu Steganografi. Steganografi merupakan cara yang paling aman dikarenakan informasi tersebut akan disembunyikan ke dalam media lain yang sekiranya tidak mudah dideteksi oleh hacker maupun cracker [2].

Pada steganografi sendiri terdapat beberapa algoritma, salah satunya yaitu End Of File (EOF). Dalam penelitian ini akan digunakan EOF sebagai algoritma yang digunakan untuk mengamankan file multimedia.

Menurut Sembiring [3] dalam penelitiannya menggunakan EOF untuk mengamankan pesan teks pada file induk berupa gambar RGB. Hasil penelitian yang dilakukan menunjukkan bahwa EOF mempunyai kemampuan untuk mengamankan data. Operasi EOF dilakukan dengan mengubah pesan yang disisipkan ke dalam bentuk biner ascii kemudian menurut warna pada file induk yang digunakan, data diubah ke dalam bentuk heksadesimal yang kemudian diiterasi sampai nilai tertentu sehingga akan terbentuk indeks dari palet warna.

Sedangkan menurut Sukrisno dkk, EOF juga digunakan untuk mengamankan data pesan. Namun pada penelitiian ini, EOF digunakan bersamaan dengan algoritma lain yaitu Rijndael, Shift Cipher dan fungsi Hash MD5. Dalam penelitian yang telah dilakukan, digunakan file berupa .html, .c, .jpg, .exe., .doc, .docx, file audio dan video. Hasilnya, penyisipan maupun ekstraksi pesan dapat dilakukan dengan baik namun terjadi selisih besar ukuran data dari data induk awal dan data setelah di sisipkan pesan [4].

Penelitian lain dalam ranah EOF juga telah dilakukan oleh Edisuryana dkk [5]. Dalam penelitiannya, file induk yang digunakan yaitu file gambar berformat bitmap sedangkan file pesan berupa file teks. Dalam penelitian ini steganografi dan kriptografi digunakan secara bersamaan sehingga mengkhasilkan ketahanan gambar yang baik, terlihat dari proses manipulasi citra yang telah dilakukan setelah file pesan dilakukan, antara lain dengan *bluring, cropping*, dan pemberian efek *sunspot.*

2. METODE

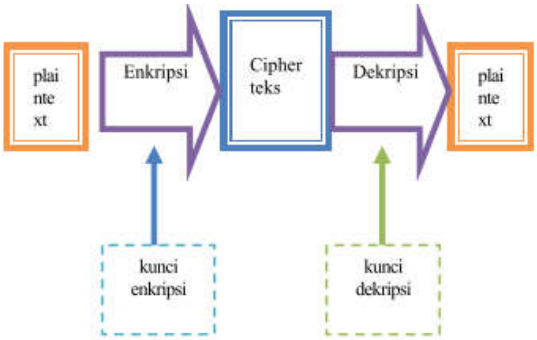
2.1 Steganografi

Steganografi merupakan salah satu teknik yang sampai saat ini masih digunakan untuk mengamankan data. Adapun jenis-jenis steganografi dikategorikan berdasarkan ketidaknampakannya yaitu invisibile dan visible, sedangkan dari media yang digunakan, steganografi dapat dibedakan menjadi 6 yaitu antara lain teks steganografi, *image* steganografi, *audio* steganorafi, *video* steganografi, dan *protocol* steganografi [6].

Pada penelitian ini, media yang digunakan yaitu semua ekstensi file. Adapun beberapa pengertian yang akan digunakan pada aplikasi steganografi yang akan dibuat [7] :

1. *Embedded message (hiddentext)*: pesan yang disembunyi.
2. *Cover-object(covertext):* pesan yang digunakan untuk menyembunyikan *embedded message*.
3. *Stego-object(stegotext):* pesan yang sudah berisi pesan *embedded message*.
4. *Stego-key*: kunci yang digunakan untuk menyisipan pesan dan mengekstraksi pesan dari *stegotext*.

Berikut merupakan bagan proses steganografi secara umum [8] :



Gambar 1. Proses Steganografi

Dalam pengembangannya, steganografi mempunyai berbagai macam algoritma, salah satunya yaitu Enf Of File (EOF).

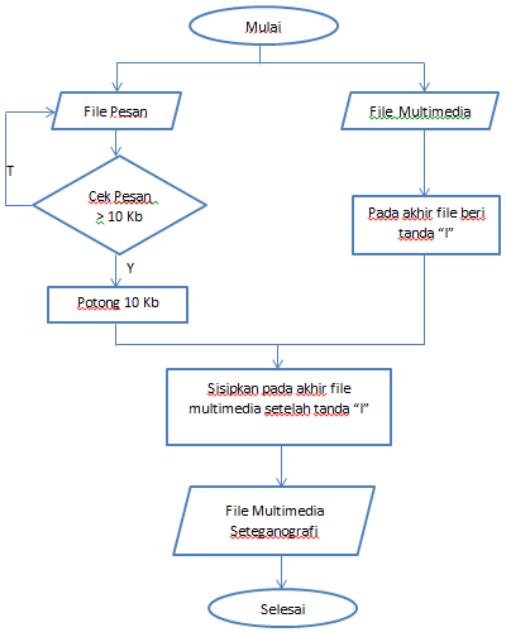
2.2 End of File (EOF)

Metode ini merupakan metode pengembangan LSB. Dalam metode ini pesan disisipkan diakhir berkas [3]. Pesan yang disisipkan dengan metode ini jumlahnya tidak terbatas. Akan tetapi efek sampingnya adalah ukuran berkas menjadi lebih besar dari ukuran semula. Ukuran berkas yang terlalu besar dari yang seharusnya, tentu akan menimbulkan kecurigaan bagi yang mengetahuinya [5].

Teknik EOF atau End Of File merupakan salah satu teknik yang digunakan dalam steganografi. 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 [4].

2.3 Proses Penyisipan Pesan

Berikut ini merupakan bagan dari proses penyisipan pesan mengggunakan algoritma End Of File (EOF).



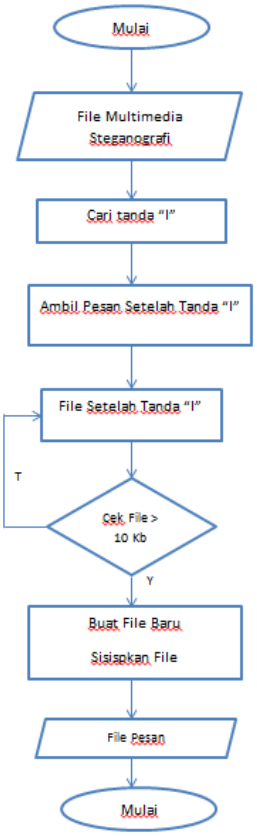
Gambar 2. Proses Penyisipan Pesan Menggunakan End Of File (EOF)

Pada Gambar 2, proses penyisipan pesan diawali dengan pengecekan terlebih dahulu pada file pesan melalui ukuran tertentu. Pada penelitian ini ditetapkan bahwa file pesan dicek dan dipotong per 10 kb dengan tujuan untuk mempercepat proses. Pada saat yang bersamaan, file multimedia yang

digubakan diberi tanda"I" pada akhir file. Setelah kedua proses ini selesai, maka EOF digunakan untuk menyisipkan file pesan setelah tanda "I" diberikan.

2.3 Proses Ekstraksi Pesan

Setelah proses penyisipan selesai, proses ekstaksi dilakukan untuk mengetahui file induk dan file pesan sebagai berikut:

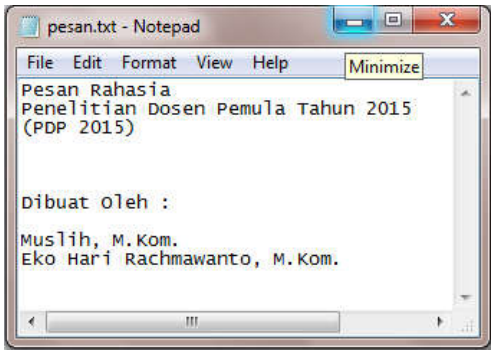


Gambar 3. Proses Ekstraksi Pesan Menggunakan End Of File (EOF)

Pada Gambar 3, proses ekstraksi file pesan dilakukan dengan mencari tanda "I" terlebih dahulu, setelah tanda "I" ditemukan maka akan dilakukan pengecekan ukuran file kemudian akan dibuat file baru dan file pesan awal akan terdeteksi.

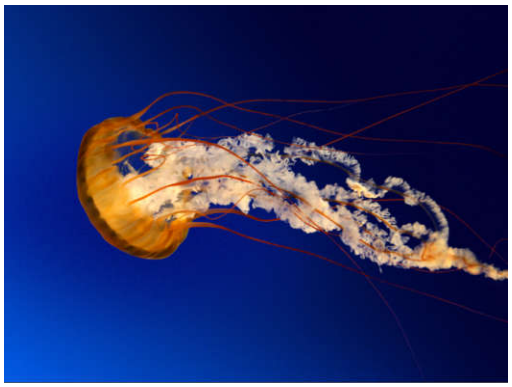
3. HASIL DAN PEMBAHASAN

Dari percobaan yang telah dilakukan dalam penelitian ini, file pesan maupun file induk yang digunakan dapat berupa file audio, video, gambar maupun teks. Interface yang digunakan untuk pembuatan aplikasi pengamanan data yaitu Visual Basic 6.0. Adapun tampilan aplikasi sebagai berikut:



Gambar 4. File pesan berupa file .txt

Pada gambar 4, merupakan file pesan rahasia yang nantinya akan dikirimkan oleh pihak pengirim kepada pihak penerima.



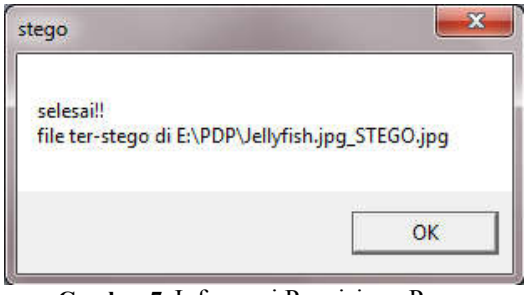
Gambar 5. File induk berupa file .jpg.

Sedangkan pada gambar 5, menunjukkan file induk dimana tempat bersembunyinya file pesan yang sangat rahasia. Dalam hal ini, sampel yang diambil berupa file *.jpg.

Percobaan dengan menggunakan file induk .jpg dan file pesan .txt maka pada aplikasi akan didapatkan tampilan sebagai berikut:

Gambar 6. Insert File Induk dan File Pada Proses Penyisipan

Pada Gambar 6, setelah file induk dan file pesan dipilih, selanjutnya tekan button "Stego". Ketika proses steganografi menggunakan algoritma EOF selesai maka akan muncul tampilan seperti pada Gambar 7 di bawah ini.



Gambar 7. Informasi Penyisipan Pesan

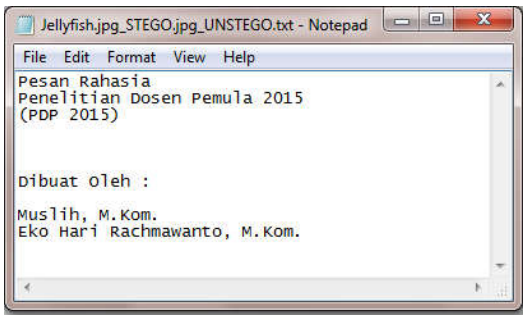
Apabila proses penyisipan pesan pada steganografi telah berhasil dilakukan, maka akan muncul tampilan seperti pada Gambar 7.

Gambar 8: Proses Ekstraksi File Pesan

Setelah proses penyispan pesan selesai dilakukan, maka untuk mengevaluasi proses penyisipan yang telah dilakukan dapat dilakukan proses ekstraksi seperti pada Gambar 8.

Gambar 9. Selesai Proses Unstego

Apabila proses ekstraksi pesan telah selesai, maka akan muncul tampilan seperti pada Gambar 9.



Gambar 10. File Pesan Hasil Proses Ekstraksi

Gambar 10 merupakan file hasil proses ekstraksi menggunakan algoritma End Of File (EOF). Hal ini dibuktikan dengan berubahnya nama file pesan awal sebelum proses penyisipan pesan dan file pesan setelah melalui proses ekstraksi.

4. KESIMPULAN DAN SARAN

4.1. Kesimpulan

Dari percobaan yang telah dilakukan dalam penelitian ini, file berhasil disembunyikan dengan baik pada file induk tanpa ada perubahan yang

signifikan terhadap file induk. Selain itu, file juga dapat diambil kembali untuk dilihat isi pesan atau informasi dengan jelas tanpa ada noise sedikitpun.

4.2. Saran

Dihimbau untuk menambahkan metode kriptografi sebelum file pesan disisipkan ke dalam file induk guna keamanan ganda supaya file lebih aman.

DAFTAR PUSTAKA

[1] F. J. Mabry, J. R. James, and A. J. Ferguson, "Unicode Steganographic Exploits," *IEE Security and Privacy*, 2007.

[2] S. K. Bandyopadhyay, D. Bhattacharyya, D. Ganguly, S. Mukherjee, and P. Das, "A Tutorial Review on Steganography," *IC3-2008 UFL & JIITU*, pp. 105-114, 2008.

[3] S. Sembiring, "MENYISIPKAN PESAN TEKS PADA GAMBAR DENGAN METODE END OF FILE," *Pelita Informatika Budi Darma*, vol. 4, no. 2, pp. 45-51, 2013.

[4] Sukrisno and E. Utami, "IMPLEMENTASI STEGANOGRAFI TEKNIK EOF DENGAN GABUNGAN ENKRIPSI RIJNDAEL , SHIFT CIPHER DAN FUNGSI HASH MD5," *Seminar Nasional Teknologi 2007 (SNT 2007)*, no. November, pp. 1-16, 2007.

[5] M. Edisuryana, R. R. Isnanto, and M. Somantri, "APLIKASI STEGANOGRAFI PADA CITRA BERFORMAT BITMAP DENGAN MENGGUNAKAN METODE END OF FILE," *Transient*, vol. 2, no. 3, 2013.

[6] Rakhi and S. Gawande, "A REVIEW ON STEGANOGRAPHY," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 2, no. 10, pp. 4635-4638, 2013.

[7] S. Mahajan and A. Singh, "A Review of Methods and Approach for Secure Stegnography," *October*, vol. 2, no. 10, pp. 67-70, 2012.

[8] Wasino, T. P. Rahayu, and Setiawan, "IMPLEMENTASI STEGANOGRAFI TEKNIK END OF FILE DENGAN ENKRIPSI RIJNDAEL," *Seminar Nasional Teknologi Informasi dan Komunikasi 2012 (SENTIKA 2012)*, pp. 150-157, 2012.