PEMANFAATAN FILE TEKS SEBAGAI MEDIA MENYEMBUNYIKAN INFORMASI RAHASIA MENGGUNAKAN TEKNIK STEGANOGRAFI DENGAN MODUS PERUBAHAN WARNA HURUF

Haryansyah[1], Dikky Praseptian M.[2]

[1), 2)] Manajemen Informatika STMIK PPKIA Tarakanita Rahmawati
Jl Yos Sudarso No.8, Tarakan, Kalimantan Utara 77111
Email : ary.abec@gmail.com[1)], dikkypraseptian@gmail.com[2)]

Abstrak

Informasi bisa jadi merupakan sesuatu hal yang biasa saja dan tidak terlalu berarti, akan tetapi sebuah informasi bisa menjadi sesuatu yang sangat berharga apabila informasi tersebut dianggap sebagai informasi rahasia yang tidak ingin diketahui oleh orang lain. Informasi rahasia adalah informasi yang karena nilainya, perlu disembunyikan dan dilindungi agar tidak terbuka untuk umum atau jatuh kepada pihak lain, dimana apabila informasi tersebut diketahui oleh pihak lain maka akan menimbulkan kerugian. Jatuhnya pesan atau informasi rahasia kepada pihak lain dapat disebabkan beberapa faktor diantaranya adanya transaksi pengiriman (send), pencurian (teaf) dan berbagai macam cara lainya.

Oleh karena itu, banyak cara yang dilakukan para pemilik informasi untuk melindungi pesan atau informasi rahasia yang dimilikinya agar tidak jatuh kepada pihak yang tidak bertanggung jawab, salah satunya adalah dengan menyembunyikan pesan atau informasi tersebut sebelum diberikan kepada pihak tertentu yang diinginkan. Menyembunyikan pesan tentu menggunakan cara yang apik, agar pesan rahasia tersebut tidak menimbulkan kecurigaan pihak tertentu dan tidak memancing untuk memilikinya. Steganografi merupakan teknik yang sangat populer yang menjadi senjata ampuh untuk menyembunyikan pesan rahasia tanpa menimbulkan kecurigaan, karena informasi disembunyikan atau disisipkan pada berkas digital dengan melakukan perubahan tipis yang isinya tidak akan menarik perhatian dari penyerang potensial. Beberapa media digital yang umumnya digunakan diantaranya format gambar (image), format audio dan format lain seperti teks file, html, pdf dan lain-lain.

Pada penelitian ini, akan menggunakan media teks file untuk menyembunyikan informasi rahasia dengan cara memainkan perubahan warna pada file teks tersebut sebagai teknik penyisipan. Pemilihan media digital ini karena hasil yang didapatkan hampir tidak menunjukkan kecurigaan terhadap adanya perubahan file karena hasil seperti file ketikan biasa.

Kata kunci: *Steganografi, warna, Multimedia, Media Digital, Informasi Tersembunyi*

1. Pendahuluan

Informasi adalah pesan (ucapan atau ekspresi) atau kumpulan pesan yang terdiri dari order sekuens dari simbol, atau makna yang dapat ditafsirkan dari pesan atau kumpulan pesan. Informasi dapat direkam atau ditransmisikan. Hal ini dapat dicatat sebagai tanda-tanda, atau sebagai sinyal berdasarkan gelombang.

Dunia komunikasi berkembang sangat pesat setiap menit. Seiring dengan hal tersebut kebutuhan informasi tentunya akan berjalan beriringan dengan perkembangan jaman. Transaksi data dan informasi berjalan hampir tidak pernah berhenti. Hal ini melibatkan beberapa pihak yang mempunyai tujuan khusus dalam melakukan transaksi tersebut. Ada beberapa orang yang hanya sekedar *sharing* pengetahuan melauli blog pribadi, sebagian juga melakukan transaksi informasi dengan tujuan menyampaikan pesan rahasia kepada pihak tertentu.

Steganografi adalah seni dan ilmu menulis pesan tersembunyi atau menyembunyikan pesan dengan suatu cara sehingga selain si pengirim dan si penerima, tidak ada seorangpun yang mengetahui atau menyadari bahwa ada suatu pesan rahasia[3]. Sebaliknya, kriptografi menyamarkan arti dari suatu pesan, tapi tidak menyembunyikan bahwa ada suatu pesan. Kata "steganografi" berasal dari bahasa Yunani *steganos*, yang artinya “tersembunyi atau terselubung”, dan *graphein*, “menulis”[5]. Teknik steganografi meliputi banyak sekali metode komunikasi untuk menyembunyikan pesan rahasia (teks atau gambar) di dalam berkas-berkas lain yang mengandung teks, *image*, bahkan audio tanpa menunjukkan ciri-ciri perubahan yang nyata atau terlihat dalam kualitas dan struktur dari berkas semula. Metode ini termasuk tinta yang tidak tampak, *microdots*, pengaturan kata, tanda tangan digital, jalur tersembunyi dan komunikasi spektrum lebar.

Kini, istilah steganografi termasuk penyembunyian data digital dalam berkas-berkas (*file*) komputer. Contohnya, si pengirim mulai dengan berkas gambar biasa, lalu mengatur warna setiap pixel ke-100 untuk menyesuaikan suatu huruf dalam alphabet (perubahannya begitu halus sehingga tidak ada seorangpun yang menyadarinya jika ia tidak benar-benar memperhatikannya).

Media teks dapat digunakan sebagai carrier (pembawa) pesan rahasia. Banyak teknik yang dapat dilakukan, salah satunya adalah permainan warna karakter (huruf) yang terdapat pada file teks yang digunakan. Untuk menghindari kecurigaan, maka pada penelitian menggunakan warna standart yaitu hitam, sehingga hasil dari file teks yang telah disisipkan pesan rahasia sangat mirip dengan file teks hasil ketikan biasa (tanpa diedit).

Hasil akhir dari penelitian ini yaitu menghasilkan file teks yang telah disisipkan pesan dalam bentuk file RTF. Pengguna dapat mengatur sendiri password yang akan digunakan untuk setiap pesan yang akan disisipkan. Hal ini juga memungkinkan penggunaan 2 (dua) password yang berbeda untuk untuk teks palsu yang sama.

2. Pembahasan

2.1 Steganografi

Steganografi adalah bentuk keamanan melalui ketidakjelasan. Ini adalah seni dan ilmu menulis pesan tersembunyi sedemikian rupa bahwa tidak ada orang kecuali pengirim dan penerima tersebut dapat memahami pesan tersembunyi tersebut[1].

Steganografi berbeda dengan kriptografi. Pada steganografi, penyadap pesan tidak dapat melihat pesan yang benar-benar ingin disampaikan, yang terlihat adalah file atau pesan lain yang menjadi persembunyian pesan sebenarnya. Pada kriptografi, penyadap pesan menyadari keberadaan pesan tersebut, hanya saja pesan tersebut tidak terbaca karena telah terenkripsi. Dengan enkripsi yang bagus mungkin penyadap pesan akan kesulitan membaca pesan tersebut, tetapi dengan menyadari keberadaan pesan tersebut, penyadap bisa berusaha untuk mendekripsikannya. Dengan steganografi, sniffer tidak menyadari keberadaan pesan, namun bila ditemukan, pesan akan mudah dibaca.

Terkadang kedua teknik tersebut dipakai bersamaan. Pesan dienkripsi terlebih dahulu menjadi suatu cipher text, kemudian cipher text tersebut disembunyikan ke dalam file lain. Sniffer akan lebih sulit mengetahui keberadaan pesan karena cipher text hanya terlihat seperti kode-kode ASCII tidak beraturan. Meskipun keberadaannya diketahui, cipher text tersebut harus didekripsi terlebih dahulu.

Watermark dan steganografi hampir sama. Hanya saja, maksud dan tujuannya berbeda. Steganografi bertujuan menyembunyikan pesan sehingga tidak terlihat selain oleh sang penerima. Watermark bertujuan mengidentifikasi kepemilikan suatu karya, misalnya seorang seniman mengunggah gambarnya (yang telah diberi watermark), kemudian seseorang mengklaim kepemilikan gambar tersebut, sang seniman dapat membantahnya dengan menunjukkan watermark pada gambar tersebut. Pada steganography, yang menjadi pesan utama adalah pesan yang disembunyikan. Sedangkan pada watermark, pesan yang utama tidak disembunyikan, watermark hanya untuk menandai.

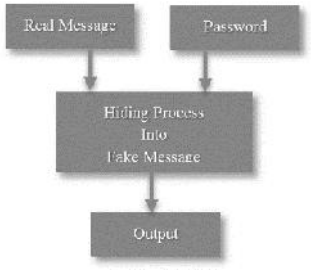
Tujuan akhir dari steganografi adalah menyembunyikan para pelaku komunikasi dalam hal ini adalah pengirim dan penerima informasi dengan menanamkan informasi yang dikirim kedalam objek digital seperti gambar, teks, audio dan berbagai file digital yang lain[4].

2.2 Algoritma Penyisipan Pesan pada File Teks

Pada penelitian ini steganografi yang digunakan adalah teknik menyembunyikan pesan atau informasi rahasia pada media *file* teks. Program tidak akan mengubah teks itu sendiri, akan tetapi mengubah beberapa attribut teks.

Implementasi steganografi pada media file teks pada penelitian ini akan bermain pada warna huruf yang digunakan sehingga pesan atau informasi rahasia dapat disisipkan pada nilai warna yang diterapkan pada media teks yang digunakan, namun warna yang digunakan untuk hasil akhir dari aplikasi ini adalah warna hitam (*black*). Tujuan dari pemilihan warna hitam ini agar tidak menimbulkan kecurigaan kepada para penyerang potensial, karena warna hitam pada teks merupakan warna yang umum digunakan.

Menyembunyikan pesan rahasia pada media file teks merupakan mode yang sangat dianjurkan dalam dunia steganografi, karena sangat stabil dan aman. Dalam mode ini akan mengubah warna karakter dalam teks palsu sesuai dengan warna yang dipilih dan warna program yang dihitung. Program akan mencari dan menemukan warna terdekat yang tidak mungkin untuk mengenali perbedaan dengan mata telanjang. Pada aplikasi ini warna yang digunakan untuk pesan rahasia dan pesan palsu (*fake message*) adalah warna hitam, sehingga seakan-akan tidak terjadi perubahan apapun pada pesan yang dapat menimbulkan kecurigaan. Secara umum, cara kerja aplikasi ini dapat diamati pada gambar 1 berikut ini.



Gambar 1. *Prinsip Kerja Steganografi File Teks*

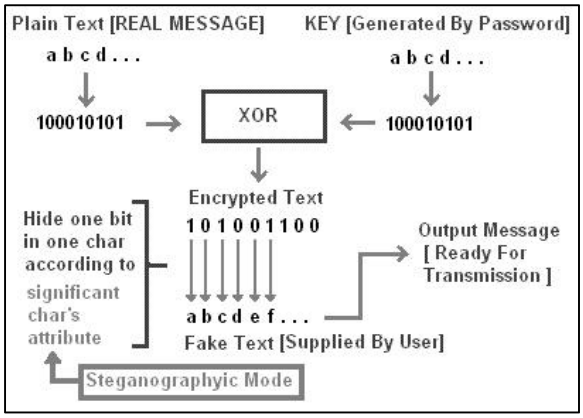
Pada gambar 1 terlihat bahwa pesan yang akan disisipkan (*Real Message*) disertai dengan *password* yang menjadi kata kunci sekaligus identitas data yang disisipkan tersebut. Apabila isi pesan akan dibaca kembali maka *password* tersebut akan diminta kembali dan disesuaikan dengan *password* yang dimasukkan oleh user. Formulanya dapat digambar sebagai berikut[2].

cover medium + embedded message + stegokey = stego-medium

Cover medium adalah teks asli yang akan disisipkan pesan, *embedde message* adalah pesan yang akan

disisipkan, *stegokey* adalah password yang digunakan untuk proses penyisipan sekaligus digunakan untuk membongkar pesan yang tersembunyi dan *stego-medium* adalah hasil akhir berupa teks yang telah disisipkan pesan.

Lebih rinci cara kerja aplikasi dapat diamati pada gambar 2 berikut[6].



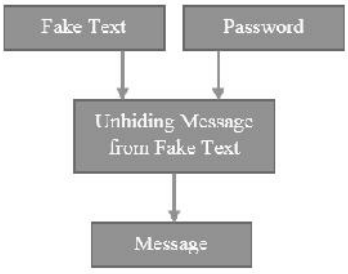
Gambar 2. *Rincian Proses Perhitungan Nilai Bit*

Pada dasarnya proses menyisipkan pesan kedalam file teks terdapat proses enkripsi atau penyandian pesan yang dilakukan dengan melakukan proses perhitungan XOR terhadap bit dari *plain text* dengan bit dari *key*. Hasil dari perhitungan nilai bit tersebut yang selanjutnya akan dimasukkan kedalam *fake text*.

Jumlah karakter yang dapat dimasukkan kedalam teks tergantung ukuran teks pembawa pesan (carrier). Pesan yang akan disisipkan tidak boleh melebihi ukuran file pembawanya. Artinya apabila teks pembawa pesan terdiri dari 100 karakter, maka pesan yang akan disisipkan tidak boleh melebihi 100 karakter.

2.3 Algoritma Ekstraksi Pesan

Untuk membongkar pesan yang telah dimasukkan kedalam *fake text* maka langkah yang dilakukan adalah terlebih dahulu mempersiapkan file teks yang telah disisipkan pesan. Hal yang tidak boleh dilupakan adalah password dari proses penyisipan pesan sebelumnya harus dengan password yang digunakan pada proses ekstraksi pesan. Algoritma yang digunakan dapat diamati pada gambar 3.



Gambar 3. *Algoritma Ekstraksi Pesan*

Yang perlu diperhatikan adalah pada saat ekstraksi pesan yang terdapat dalam *fake text* yang dibaca adalah nilai warna pada *fake text* tersebut yang selanjutnya akan dilakukan proses XOR dengan password sehingga dapat menghasilkan pesan yang tersembunyi didalamnya. Berikut formula yang digunakan.

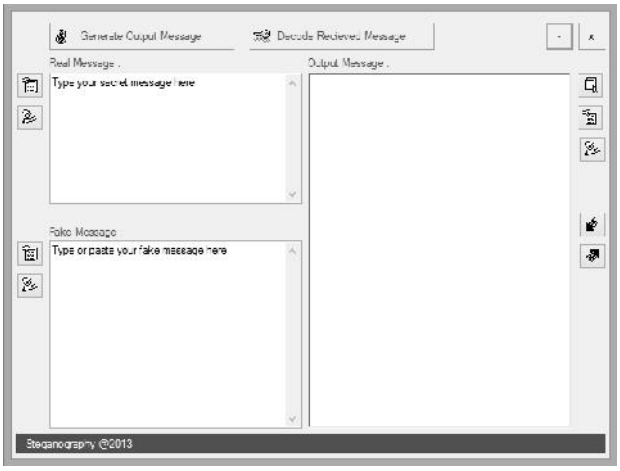
Stego-medium + stegokey = secreet message

2.4 Desain Interface

Berikut ini adalah beberapa desain tampilan (*interface*) dari aplikasi steganografi untuk menyisipkan pesan kedalam file teks yang telah dibuat.

a. Tampilan Awal

Pada saat pertama kali menjalankan aplikasi maka akan didapatkan hasil *output* seperti pada gambar 4 berikut ini. terlihat ada 3 (tiga) bagian penting dalam form yaitu 1) *Real Message* yang merupakan tempat untuk mengetikkan pesan rahasia yang akan disisipkan kedalam file teks. 2) *Fake Message* yang merupakan pesan palsu yang digunakan sebagai media pembawa (*Carry*) untuk pesan rahasia. 3) *Output Message* yang merupakan hasil akhir dari proses memasukkan pesan kedalam media file teks tersebut. Kemudian terdapat 2(dua) tombol yaitu *Generate Ouput Message* yang berfungsi untuk memasukkan pesan rahasia kedalam pesan palsu, *Decode Received Message* yang berfungsi untuk membaca pesan yang terdapat dalam *fake message* (pesan palsu).

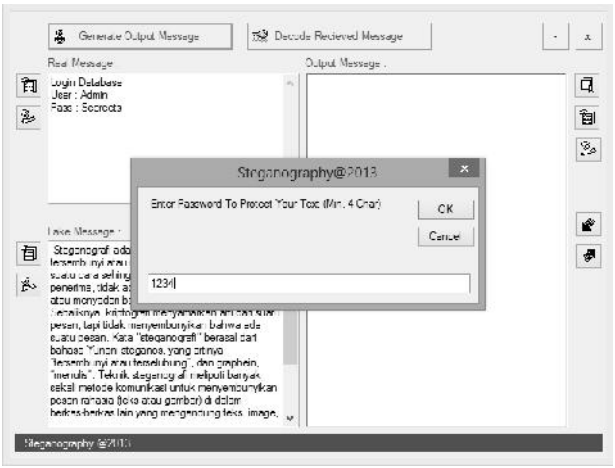


Gambar 4. *Tampilan Awal*

b. Operasi Program

Tampilan pada gambar 5 merupakan proses menjalankan aplikasi dengan mencoba memasukkan pesan kedalam sebuah teks yang nantinya dilengkapi dengan *password.* Pada saat tombol *Generate Output Message* diklik maka sebuah input box untuk permintaan password akan tampil. Password yang dimasukkan nantinya akan digunakan untuk menyisipkan pesan kedalam teks pembawa (carrier) sekaligus juga akan digunakan untuk mengekstrak pesan yang tersebunyi didalam *fake text*

apabila ingin menetahui informasi yang ada didalamnya. Jadi pengirim dan penerima pesan harus sama-sama mengerahui password tersebut dan jangan sampai hilang, karena bisa berakibat pesan tidak akan dapat diekstrak dari *fake text.*



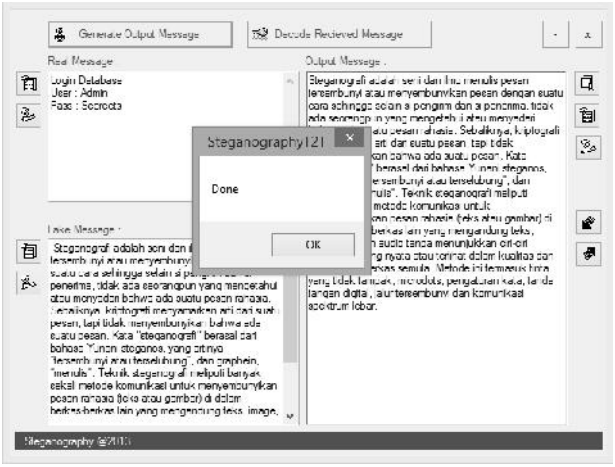
Gambar 5. *Operasi Program*

Pada saat tombol *Generate Output Message* diklik maka akan tampil *DialogBox* permintaan *password* sebagai pengaman terhadap pesan yang disembunyikan sekaligus sebagai kunci pembuka untuk membaca pesan rahasia yang telah disisipkan sebelumnya.

Dalam kasus ini, pengguna akan menyisipkan sebuah informasi rahasia yaitu berupa *user* dan *password* untuk login database dan dianggap sangat berbahaya apabila diketahui oleh orang lain. Hal ini dilakukan karena informasi ini akan dikirimkan via email dan sangat rawan penyadapan ataupun pencurian data. Oleh karena itu informasi rahasia tersebut dimasukkan kedalam media tulisan yang berisi informasi artikel sederhana yang dianggap sebagai tulisan biasa dan tidak terlalu berarti.

c. Hasil Akhir Pesan

Tampilan berikut ini merupakan hasil akhir dari proses menyisipkan pesan kedalam file teks yang dapat mengamankan informasi rahasia.



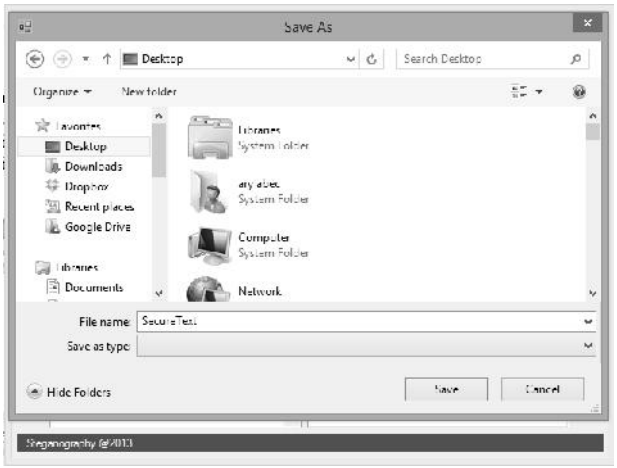
Gambar 6. *Hasil Akhir Pesan*

Hasil yang tampak pada gambar 6 pada bagian *output message* adalah tulisan atau teks yang sudah mengandung isi pesan rahasia yang berhasil disisipkan. Hasil teks tersebut selanjutnya dibisa di-copy dan di-paste pada file *.doc atau dapat juga disimpan dalam bentuk file *.rtf dengan mengklik tombol save disebelah kanan *output message.*

d. Menyimpan Pesan Rahasia

Untuk hasil akhir teks yang mengandung pesan rahasia selanjutnya dapat disimpan dalam bentuk file *.rtf yang selanjutnya dapat ditransmisikan menggunakan media apapun dengan aman. Untuk menyimpan dapat dilakukan dengan mengklik tombol *save*. File dengan ekstensi *.rtf ini selanjutnya dapat dibuka kembali untuk mengambil isi atau teks tulisan yang berisi pesan rahasia tadi untuk selanjutnya dibongkar untuk membaca isi pesan yang tersembunyi didalamnya.

Catatan, pada saat menyimpan teks yang sudah disisipkan pesan ataupun meng-copy isi tulisan kedalam file *.doc atau media file teks lainnya, jangan sampai melakukan perubahan warna tulisan atau mengganti format tulisan yang ada, karena hal ini akan mempengaruhi isi dari pesan rahasia yang terkandung didalamnya. Setiap nilai warna yang terkandung didalam tulisan tersebut mempunyai nilai yang merupakan hasil perhitungan dari aplikasi.



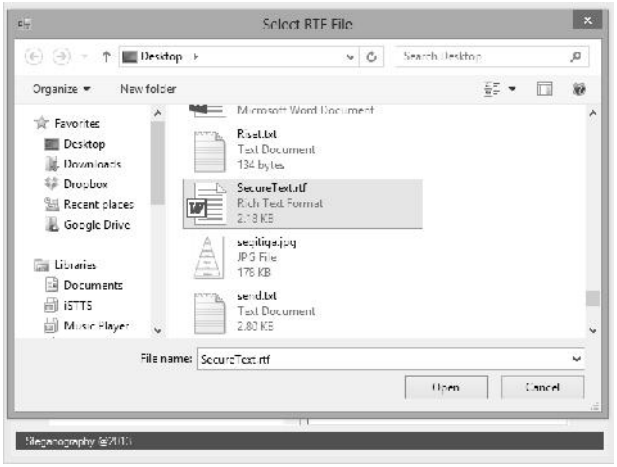
Gambar 7. *Menyimpan Pesan Rahasia*

Sebagai contoh teks hasil dari penyisipan pesan rahasia lewat aplikasi tadi disimpan dengan nama "SecureText.rtf" yang diletakkan di desktop komputer seperti tampak pada gambar 7. File ini selanjutnya dapat diolah tanpa menimbulkan kecurigaan karena bentuk dan strukturnya sama seperti file *.rtf pada umumnya.

e. Membuka Isi Pesan Rahasia

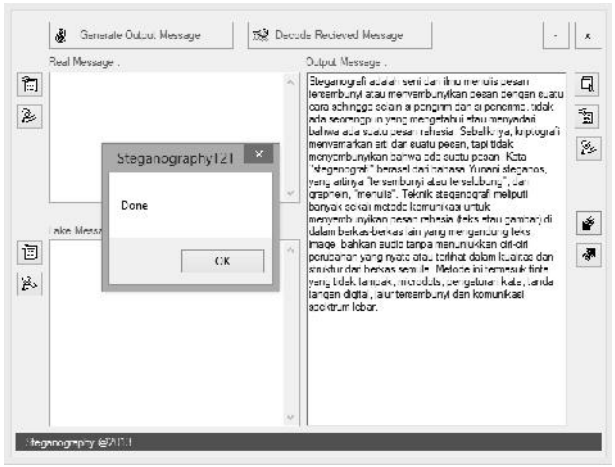
Untuk membuka isi pesan rahasia yang terkandung dalam file "SecureText.rtf" tadi dapat dilakukan dengan melakukan *import file* dengan cara klik tombol *import file* yang ada disebelah kanan kemudian cari file yang disimpan tadi.perhatikan gambar 8. Pada saat tombol

import file diklik maka akan tampil sebuah open dialog box yang akan mengarahkan pengguna aplikasi untuk mencari file teks yang berisi pesan rahasia yang nantinya akan diekstrak. Setelah file teks yang dimaksud ditemukan dan dibuka maka seluruh isi dari file tersebut ditampilkan di bagian *output message.*



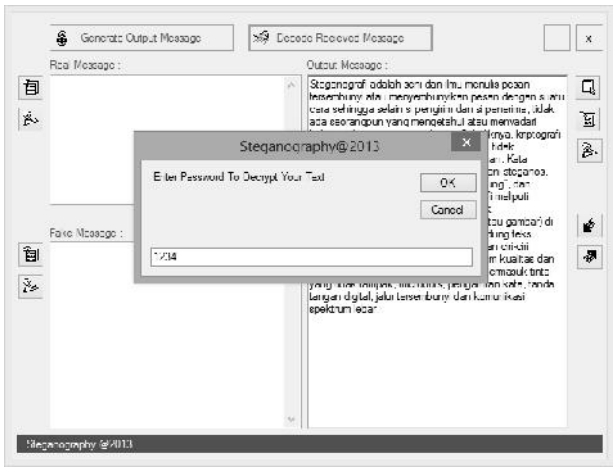
Gambar 8. *Membuka File RTF*

Selanjutnya setelah tombol "Open" diklik maka isi dari file RTF tersebut secara otomatis akan masuk kedalam *output message.* Perhatikan gambar 9 berikut ini.



Gambar 9. *Mengambil Isi File RTF*

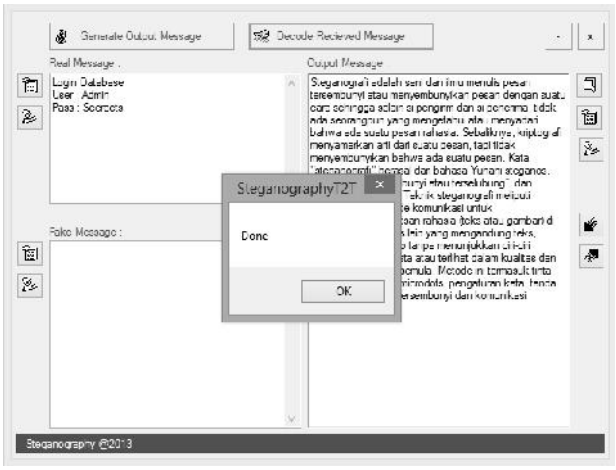
Proses berikutnya adalah membuka isi pesan tersembunyi dengan cara mengklik tombol *Decoded Receive Message.*



Gambar 10. *Permintaan Password Verifikasi*

Pada saat tombol *Decode Received Message* diklik, selanjutnya aplikasi akan meminta *password* pengaman untuk memastikan bahwa file teks tersebut sudah mendapatkan ijin untuk dibuka. Proses dapat diamati pada gambar 10.

Setelah *password* dimasukkan dengan benar maka isi pesan rahasia akan segera ditampilkan pada bagian *Real message.* Hasilnya dapat diamati pada gambar 11 berikut ini.



Gambar 11. *Hasil Pembacaan Pesan*

3. Kesimpulan

Penggunaan media file teks sebagai *carry* (pembawa) terhadap pesan rahasia sangat stabil dan aman. Hasil tidak menunjukkan adanya pesan rahasia yang tersembunyi didalamnya karena teks yang dihasilkan sama seperti teks biasa. Menyembunyikan pesan atau informasi rahasia dengan teknik steganografi merupakan cara yang sangat efektif dan cukup terjamin keamanannya, karena dengan permainan warna yang tepat, maka akan didapatkan hasil yang luar biasa.

Daftar Pustaka

[1] S.R. Govada, B.S. Kumar, M. Devarakonda, M.J. Stephen, "Text Steganography with Multi Level Shielding", *IJCSI International Journal of Computer Science Issue*, Vol.9, No.3, July 2012
[2] K. Rabah, "Steganography-The Art of Hiding Data", *Information Technology Journal (245-269)*, Vol. 3, No. 3, 2004.
[3] S.K.Bandyopadhyay, I.K.Maitra, "An Application of Palette Based Steganography", *International Journal of Computer Application (0975-8887)*, Vol. 6, No. 4, September 2010.
[4] T. Penvy, J. Fridrich, "Determining the Stego Algorithm for JPEG Images", *in IEEE Proc.-inf.Secure.,Vol. 153, No.3, September 2006*
[5] Admin, "Steganografi", Wikipedia, [online]. Tersedia: http://id.wikipedia.org/wiki/Steganografi [Diakses 15 November 2013].
[6] Admin, "Text 2 Text Steganography", Codeproject, [online]. Tersedia: http://www.codeproject.com/Articles/19260/Text-2Text-Steganography-Part-2 [Diakses 15 November 2013].

Biodata Penulis

Haryansyah, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Teknik Informatika STMIK PPKIA Tarakanita Rahmawati Tarakan, lulus tahun 2011. Saat ini menjadi Dosen di STMIK PPKIA Tarakanita Rahmawati Tarakan.

Dikky Praseptian M, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Sistem Informasi STMIK PPKIA Tarakanita Rahmawati Tarakan, lulus tahun 2013. Saat ini menjadi Dosen di STMIK PPKIA Tarakanita Rahmawati Tarakan.