

MODUL PRAKTIKUM 2022

KEAMANAN SISTEM INFORMASI

MODUL 3

Hacking Linux System &
Escalating Priviledge

ASISTEN

Ananda Anggie Nur Aini	ENJI
Andi Sayid Muhammad Qoyyum	ANDI
Arya Bimo Bagas Penggalih	BIMO
Faris Aufar Putra	PTRA
Farras Naim	RAAS
Fie Alfain Nuril Haque	ALFA
Fitria Nikmatul Hidayah	WPIN
M. Alwi Zein	ZEIN
Maulana Malik Ibrahim	IBRA
Milenia Ari Oktaviana	ARII
Muhammad Hafiz Hawarizmi	VISS
Muliya Dewi	MYDE
Ni Made Meliana Listyawati	MELI
Nurul Annisaa	NASA
Rizal Indera	INRA
Ryan Supriadi Ramadhan	RYAN
Syarah Tazkiatun Nupus	AZKI
Wiratama Putra Prakosa	RAKO



MODUL 3

Steganografi

I. Tujuan Praktikum

- 1.1 Peserta praktikum dapat mengerti dan paham tentang Steganografi.
- 1.2 Peserta praktikum mengerti fungsi Steganografi.
- 1.3 Peserta praktikum dapat mempraktikan Steganografi.

II. Alat dan Bahan

- 2.1 Laptop/PC with Windows OS
- 2.2 StegHide UI
- 2.3 OpenPuff
- 2.4 Audacity

III. Landasan Teori

3.1 Steganografi

3.1.1 Definisi Steganografi

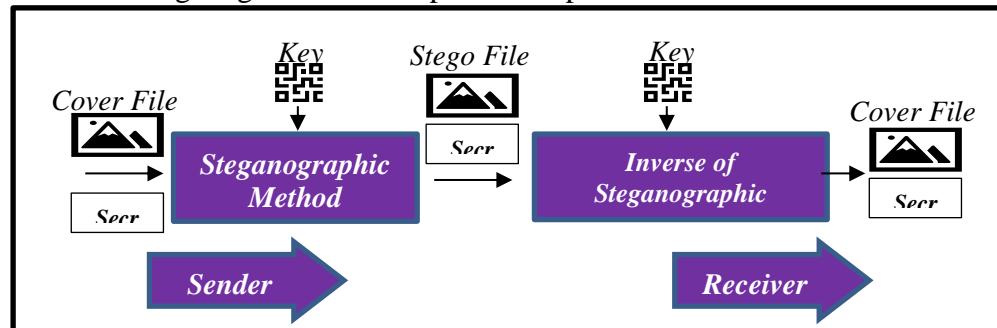
Steganografi adalah teknik menyembunyikan atau menyamarkan keberadaan informasi rahasia dalam suatu media sedemikian rupa sehingga keberadaan informasi tersebut tersembunyi dan tidak diketahui. Istilah steganografi berasal dari bahasa Yunani yang terdiri dari kata steganos yang berarti "tersembunyi" dan graphien berarti "tulisan", yang secara harfiah berarti "tulisan tersembunyi".

Steganografi telah digunakan sejak sekitar 2.500 tahun yang lalu untuk kepentingan politik, militer, diplomatik dan pribadi sebagai sarana penyampaian pesan. Catatan pertama tentang Steganografi ditulis oleh Herodotus, seorang sejarawan Yunani. Herodotus mengirim pesan rahasia menggunakan kepala budak atau tentara sebagai media, dengan menulis pesan di kepala budak yang telah botak, ketika rambut budak telah tumbuh, budak dikirim untuk membawa pesan rahasia di balik rambutnya.

Teknik ini mencegah timbulnya kecurigaan pihak luar tentang adanya informasi rahasia karena media yang telah disisipkan informasi rahasia (stego file) memiliki perbedaan dengan media aslinya (cover file) sehingga tidak dapat disadari secara langsung oleh manusia. Ada beberapa format media dalam Steganografi dalam teknik penyembunyian file yaitu melalui format gambar, teks, video dan audio, namun yang paling populer digunakan adalah format gambar dan audio.

3.1.2 Mekanisme Steganografi

Untuk menyisipkan data yang ingin Anda sembunyikan memerlukan dua elemen. Elemen pertama adalah media penampung seperti gambar, audio, video dan sebagainya yang terlihat tidak mencurigakan untuk menyimpan pesan rahasia. Elemen kedua adalah pesan yang ingin disembunyikan yaitu media penampung yang disebut dengan file sampul dan objek yang telah disisipkan pesan tersebut disebut dengan file stego. Struktur Steganografi sendiri dapat dilihat pada Gambar III.1 di bawah ini.



Gambar III. 1 Skema Steganografi

Catatan:

- Pesan Rahasia informasi rahasia dirahasiakan.
- File Stego, versi modifikasi dari file yang berisi informasi rahasia di dalamnya.
- Cover file / Digital Medium, file untuk menyembunyikan informasi rahasia.h
- Key, Data rahasia yang dibutuhkan dalam proses embedding dan extracting yang harus diketahui oleh pengirim dan penerima. Tombol berfungsi untuk dapat membuka informasi yang tersembunyi di dalam file stego.
- Metode Steganografi, sebuah fungsi Steganografi yang mengambil pesan rahasia, file sampul, dan kunci sebagai parameter dan menghasilkan file stego sebagai output.
- Kebalikan dari Metode Steganografi, Sebuah fungsi Steganografi yang memiliki file stego dan kunci sebagai parameter. Kebalikan dari metode yang digunakan dalam proses penyembunyian berarti hasil dari proses ekstraksi identik dengan input dari proses embedding.

3.1.3 Proses Embedding dan Extraction

Dalam Steganografi, ada dua proses dalam menyembunyikan informasi rahasia, yaitu :

- a. Embedding/Encoding Process

Proses Embedding merupakan langkah untuk menyembunyikan

informasi rahasia pada cover file. Hasil yang Anda dapatkan dari fungsi Embedding adalah versi modifikasi dari file yang memiliki informasi rahasia di dalamnya. File ini biasanya disebut file stego.

b. Extracting/Decoding Process

Proses extracting merupakan kebalikan dari proses Embedding yaitu proses memisahkan file stego antara secret file dan cover file. Proses ini memiliki dua parameter yang diperlukan untuk dapat memulai proses ekstraksi file rahasia, yaitu file stego dan kunci. Jika kunci yang digunakan penerima sama dengan kunci yang digunakan pengirim untuk menyembunyikan informasi rahasia dalam file stego, maka penerima dapat membuka file stego dan menemukan pesan tersembunyi tersebut.

3.1.4 Format dalam Steganografi

Ada empat format file yang biasa digunakan dalam Steganografi, yaitu:

a. Text Steganografi

Steganografi Teks adalah format yang paling sulit dan merupakan media digital yang jarang digunakan, karena file teks memiliki jumlah data redundant yang sangat kecil dibandingkan dengan media digital lainnya seperti gambar, audio dan video. Sedangkan dalam penyimpanan file teks, Steganografi teks hanya memakan sedikit memori.

b. Images Steganografi

Penggunaan gambar sebagai objek untuk menyimpan informasi rahasia sangat populer. Informasi yang disematkan pada citra digital menggunakan algoritma Embedding dan menggunakan kunci rahasia. Kemudian file gambar tersebut diproses dengan algoritma ekstraksi untuk mendapatkan informasi yang tertanam di dalamnya. Selama transmisi gambar stego, orang yang tidak berwenang menganggap gambar itu hanya gambar biasa dan tidak memiliki informasi apa pun tentangnya.

c. Audio/Video Steganografi

Audio atau video Steganografi memanfaatkan sifat indera pendengaran dan penglihatan manusia dalam menyembunyikan informasi tanpa disadari saat memutar audio atau video. Informasi disimpan dengan memilih saluran terpisah untuk menyembunyikan informasi. Kemudian gunakan media digital khusus untuk mendapatkan informasi tersebut.

d. Protokol Steganografi

Protokol Steganografi menyimpan informasi rahasia ke dalam protokol jaringan, yaitu TCP/IP. Metode ini dilakukan dengan menyembunyikan informasi di header paket TCP/IP. Salah satu contoh penempatan informasi pada nomor port TCP sumber dan nomor port TCP tujuan.

3.1.5 Aplikasi Steganografi

Teknik steganografi yang sering dilakukan pada media digital biasanya membutuhkan dukungan suatu perangkat lunak. Perangkat lunak yang digunakan dapat membantu pengguna untuk menyimpan informasi rahasia atau membuka informasi rahasia dalam data. Beberapa contoh software ini antara lain Secret Layer, OpenPuff, Audacity, Camouflage, SilentEye, StegHide, Our Secret, JSteg, Sonic Visualizer and others.

a. OpenPuff

OpenPuff Steganography and Watermarking, atau biasa disingkat OpenPuff atau Puff, adalah alat Steganografi gratis untuk sistem operasi Windows yang dibuat oleh Cosimo Oliboni dan masih dikembangkan sebagai perangkat lunak independen (tidak terikat pada perusahaan atau organisasi mana pun). Program ini tercatat sebagai alat Steganografi pertama yang dirilis pada bulan Desember 2004. OpenPuff memungkinkan penggunanya untuk menyembunyikan file stego dalam file stego. Selain itu, OpenPuff juga menyediakan pilihan tingkat enkripsi dan menyarankan pengguna untuk menggunakan tiga kata sandi.

Versi terbaru OpenPuff mendukung berbagai format seperti berikut:

- *Image*: bmp, jpg, png, tga
- *Audio*: aiff, mp3, wav
- *Video*: 3gp, mp4, mpeg I, mpeg II, vob
- *Flash-Adobe*: flv, pdf, swf

b. StegHide UI

StegHide UI adalah program Steganografi Opensource yang ditulis oleh Drunken Canadian di situs sourceforge.net dan digunakan untuk mengenkripsi dan menyembunyikan data dalam gambar (format .jpeg, .bmp) dan file audio (.wav, au). Ada tab yang dapat digunakan untuk melakukan Steganografi dalam antarmuka pengguna grafis (GUI) dan baris perintah. Di UI StegHide, pengguna dapat mengubah metode

enkripsi default, mengubah folder keluaran file stego atau mengubah warna latar belakang baris perintah, font dan warna font.

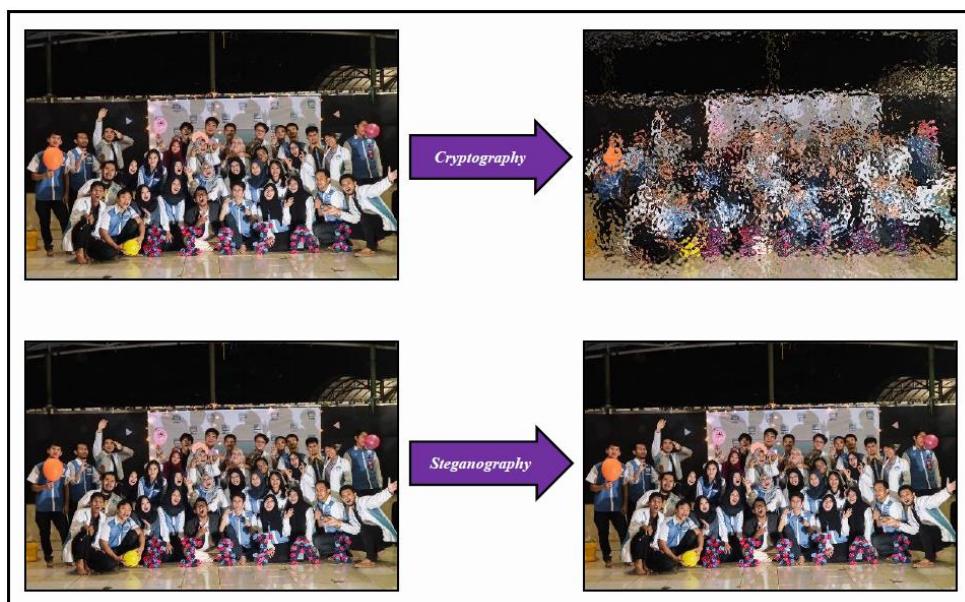
Tampilan GUI memudahkan proses Steganografi. Pengguna dapat menyembunyikan dan mengenkripsi file pada tab "Embed", memilih gambar atau suara operator untuk menyembunyikan data, memasukkan kata sandi dan memilih algoritma enkripsi dan metode lain yang ingin mereka gunakan. Untuk mendekripsi pesan Steganografi, pengguna dapat menggunakan tab "Extract", memasukkan Kata Sandi dan memilih file output dengan format ekstensi (mis. File_Dekrip.txt).

c. Audacity

Audacity adalah perangkat lunak audio Opensource yang mudah digunakan untuk mengedit, mengonversi, dan merekam audio di Windows, Mac OS X, GNU / Linux, dan sistem operasi lainnya. Audacity dapat membantu proses Steganografi dengan menyembunyikan pesan rahasia berupa Kode Morse, Gambar, atau pola sederhana lainnya dengan fitur Spectrogram dan bantuan software Coagula.

3.1.6 Perbedaan Steganografi dan Cryptography

Steganografi dan Kriptografi memiliki keterkaitan yang erat, namun keduanya merupakan dua hal yang berbeda. Letak perbedaan yang mendasar adalah hasil tampilan pesan ketika pesan rahasia telah disisipkan. Hasil Kriptografi biasanya berupa data yang berbeda dengan data saat sebelum menyisipkan pesan rahasia dan biasanya data tersebut terkesan berantakan (tetapi dapat dikembalikan ke bentuk semula), untuk membuka file tersebut harus mendekripsi file tersebut sendiri sedangkan keluaran dari Steganografi berupa persepsi yang terlihat sama dengan data yang belum disisipi pesan rahasia. Untuk lebih jelasnya lihat Gambar III.2



Gambar III. 2 Steganografi and Cryptography

Agar lebih mudah dipahami, dapat dilihat pada Tabel III.1 Perbedaan Steganografi dan Kriptografi.

Tabel III. 1 Perbedaan Steganografi dan Kriptografi.

Steganografi	Kriptografi
Informasi yang dikirim tidak diketahui.	Informasi yang dikirim dapat diketahui.
Mencegah untuk menemukan komunikasi antara pengirim dan penerima secara langsung	Enkripsi mencegah pihak yang tidak bertanggung jawab mengetahui isi dari suatu komunikasi.
Teknologi yang sedikit diketahui.	Teknologi yang banyak diketahui.
Teknologi terus dikembangkan untuk format file tertentu.	Sebagian besar algoritma yang digunakan secara umum telah diketahui
Sekali terdeteksi, isi pesannya dapat diidentifikasi.	Algoritma yang digunakan tahan terhadap serangan, membutuhkan kemampuan komputasi yang baik untuk cracking.
Steganografi tidak mengubah struktur dan informasi yang dirahasiakan.	Kriptografi mengubah struktur informasi yang dirahasiakan.

IV. Practice Lab

4.1 Steganografi with Image

4.1.1 Menyembunyikan Pesan dalam Image menggunakan OpenPuff

- a. Sebelum masuk ke tahap hiding *file* dengan menggunakan *software* OpenPuff, persiapkan terlebih dahulu 2 *file* berupa *image file* dan *text file*. Pada kasus ini *text file* bernama “SISJAR.txt” akan disembunyikan pada *image file* bernama “Gambar Jurnal.png”.

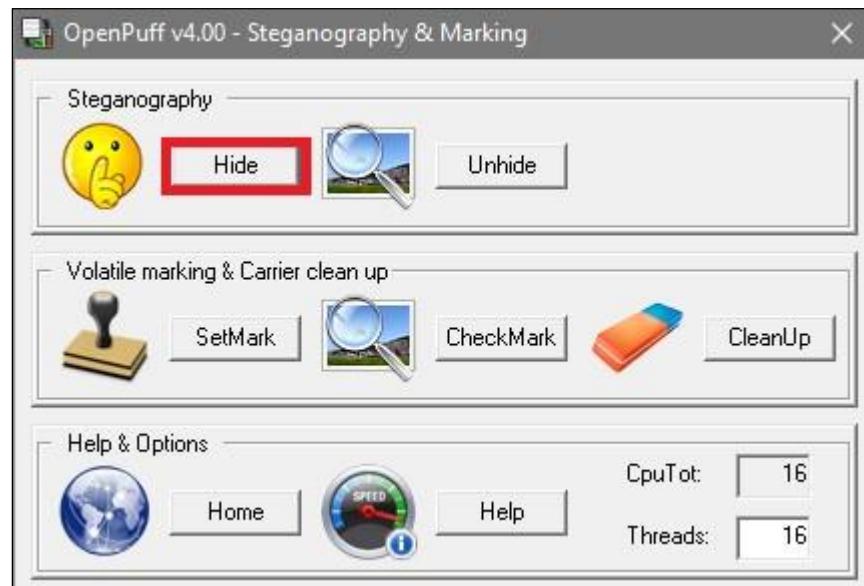


Gambar III. 2 Raw File



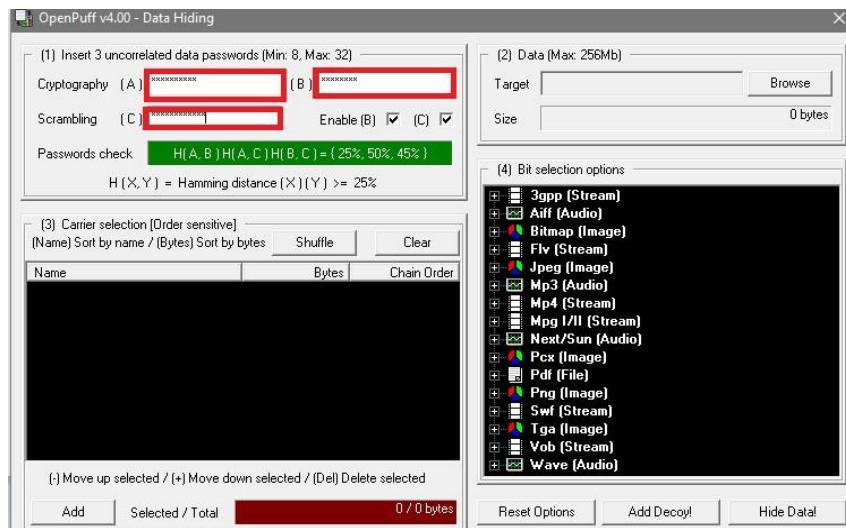
Gambar III. 3 Pesan Rahasia

- b. Buka *software* OpenPuff, kemudian pada bagian *Steganography* klik tombol “*Hide*”.



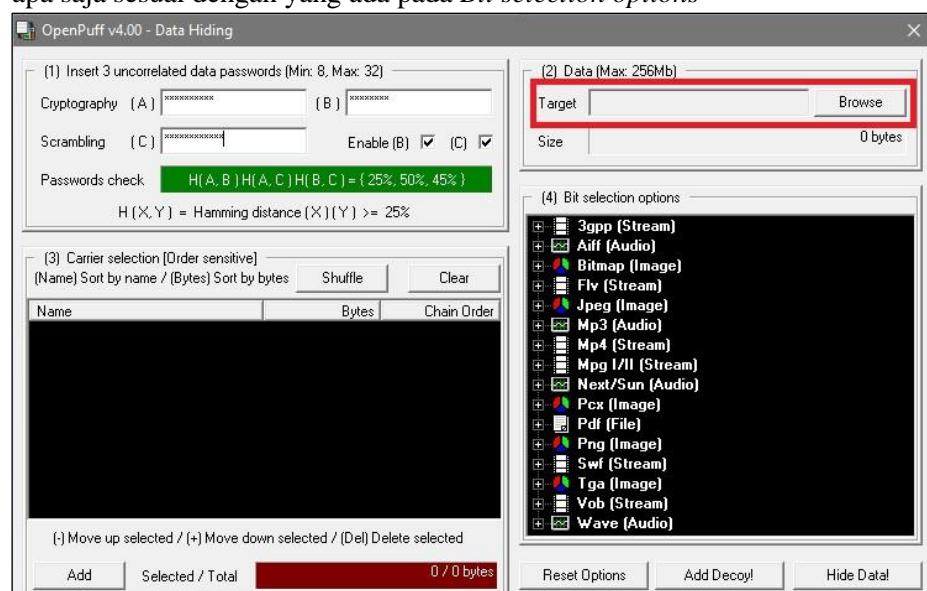
Gambar III. 4 OpenPuff

- c. Langkah selanjutnya, isi 3 *Password* yang berbeda pada tiap kolom [A], [B], dan [C]. *Password* yang diisi harus memiliki minimal 8 karakter, dan maksimal 32 karakter. (Tidak wajib untuk mengisi tiga - tiganya, boleh satu atau dua).



Gambar III. 5 Insert Key

- d. Kemudian pada kolom target, pilih *file* yang akan disembunyikan dengan mencarinya menggunakan tombol “Browse”. *File* tersebut dapat berformat apa saja sesuai dengan yang ada pada *Bit selection options*



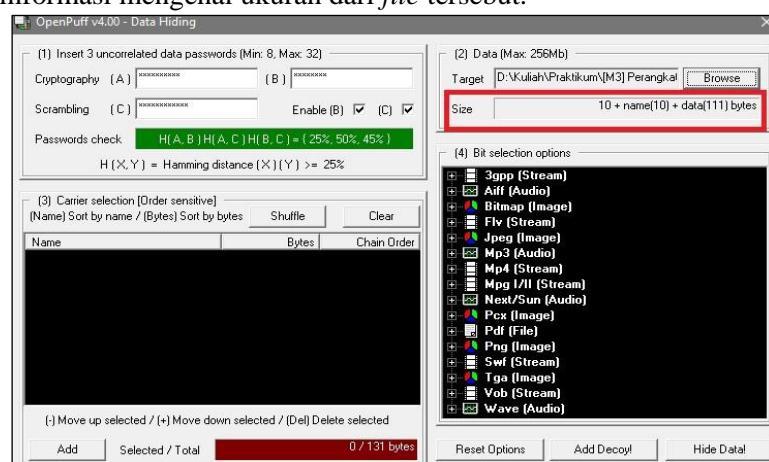
Gambar III. 6 OpenPuff Hide Menu

- e. Pilih *file* yang akan disembunyikan, pada kasus ini menggunakan *file* dengan format .txt, isi *file* teks tersebut dengan kalimat yang telah ditentukan, kemudian pilih “Open”.



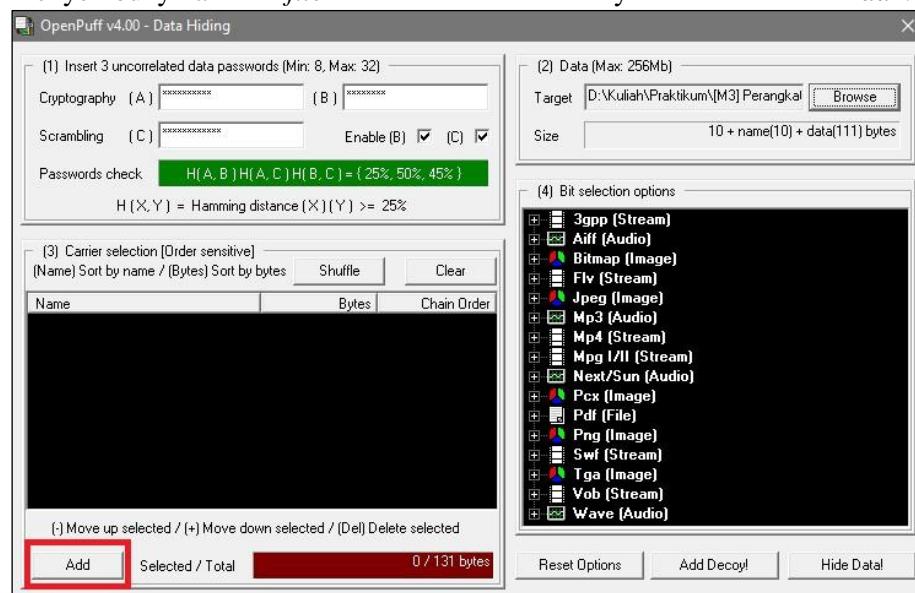
Gambar III. 7 Browse Message File

- f. telah memilih *file* yang akan disembunyikan, aplikasi akan memberikan informasi mengenai ukuran dari *file* tersebut.



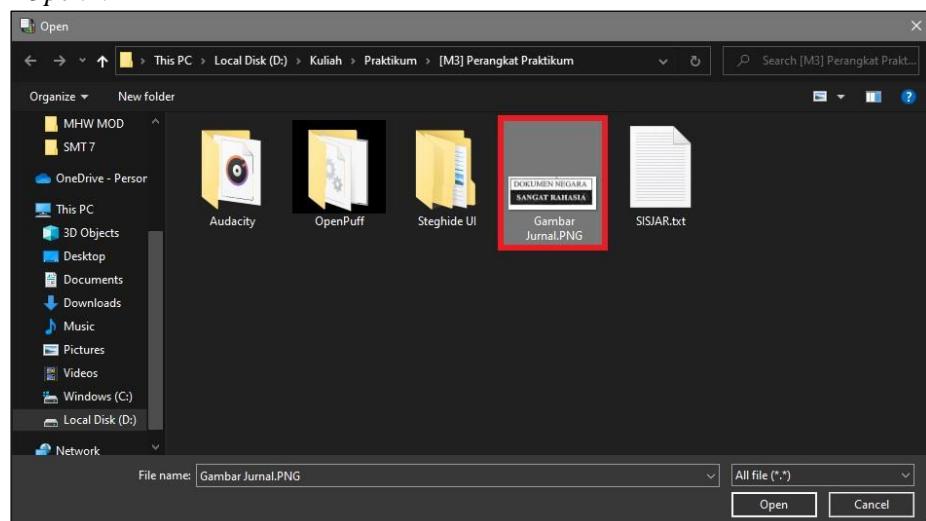
Gambar III.8 File Information

- g. Lalu pilih *file image* yang akan dijadikan *file* utama sebagai tempat menyembunyikan *file* teks sebelumnya. Klik “Add”.



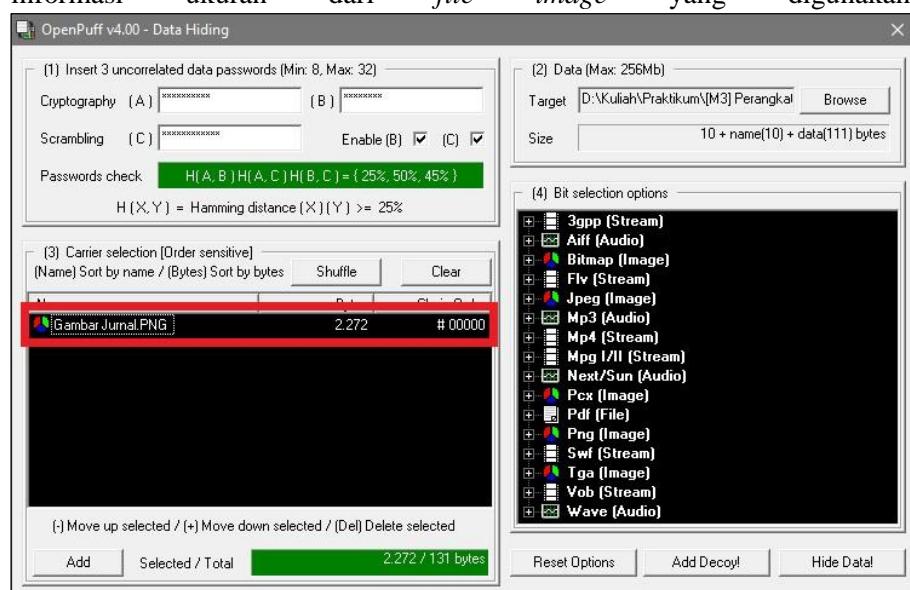
Gambar III. 9 Add Image

- h. Selanjutnya akan muncul *dialog box* untuk mencari *file* yang ingin dimasukan, pilih *file image* sesuai dengan yang telah disiapkan, lalu klik “*Open*”.



Gambar III. 10 Select Image File

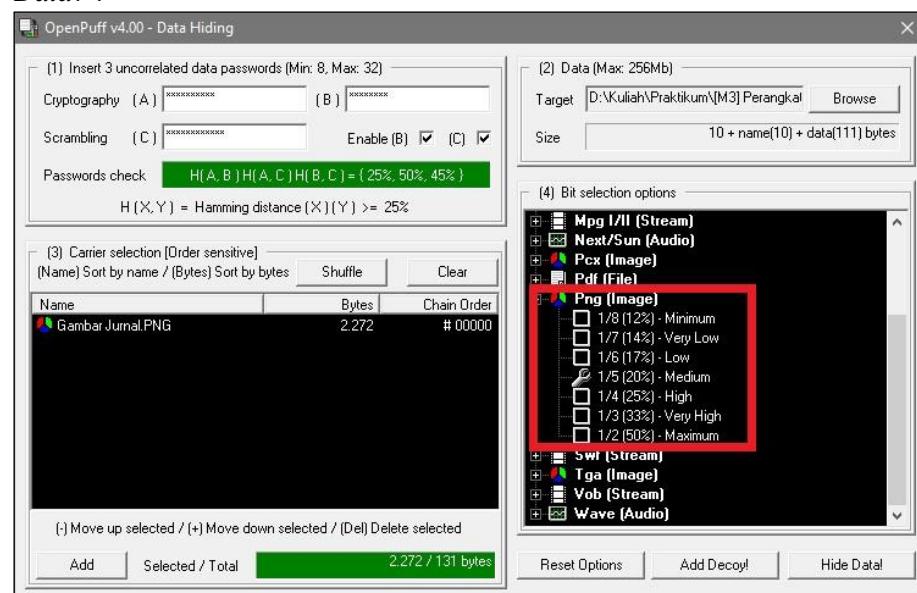
- i. Sama dengan jenis *file* yang disembunyikan, OpenPuff juga memberikan informasi ukuran dari *file image* yang digunakan



Gambar III. 11 Image File Information

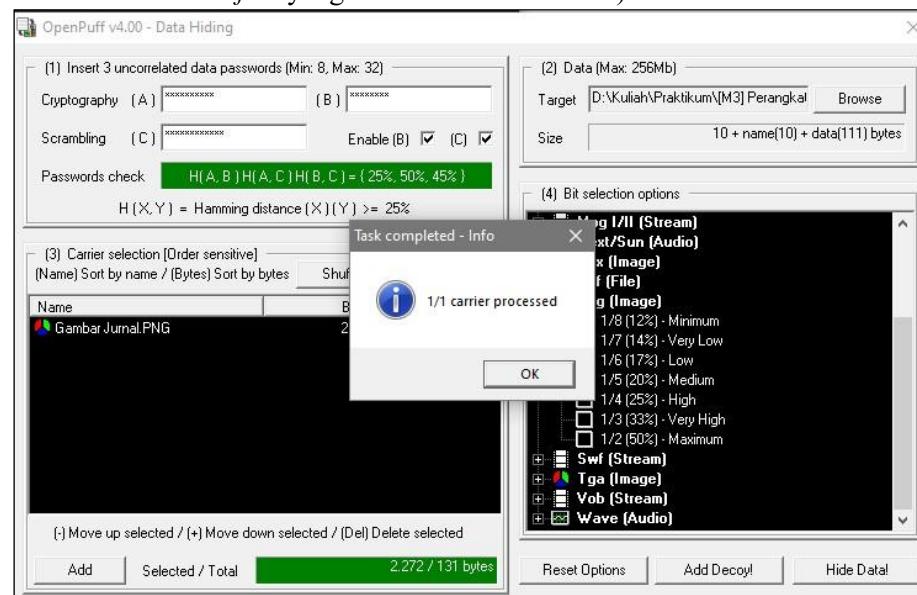
- j. Selanjutnya pilih jenis keluaran gambar yang akan dihasilkan, lalu *expand* dan pilih kualitas dari gambar tersebut (opsional), karena secara tidak langsung, *software* akan menentukan secara otomatis jenis gambar yang digunakan, berdasarkan kategori gambar tersebut. Kemudian klik “*Hide*

Data!“.



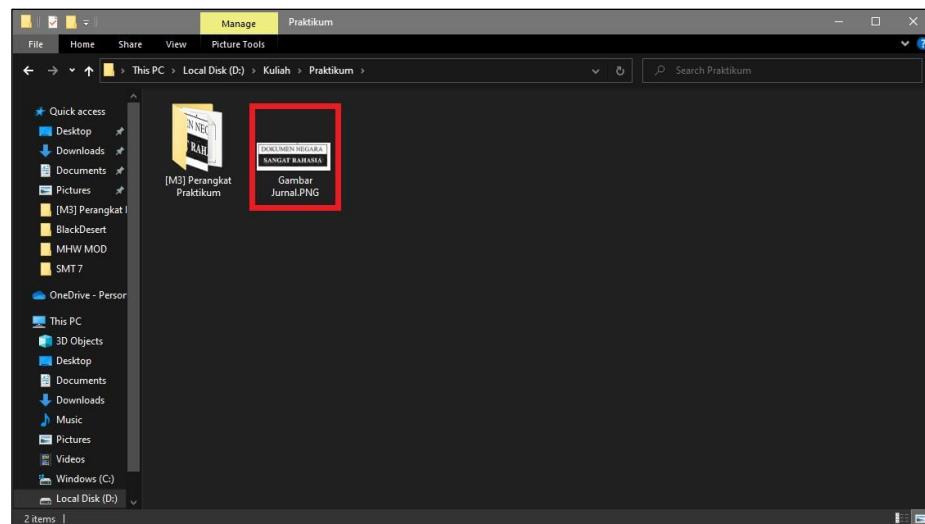
Gambar III. 12 Hiding Message

- Selanjutnya pilih direktori untuk menyimpan hasil dari *Steganography* (Jangan menyimpan di dalam *folder* yang sama dengan *file* sebelumnya, karena nama dari *file* yang dihasilkan akan sama). Kemudian klik “OK”.



Gambar III. 13 Task Complete Info

- Proses pembuatan *image Steganography* berhasil dilakukan, lakukan pengecekan hasil gambar tersebut.



Gambar III. 14 Checking Stego File

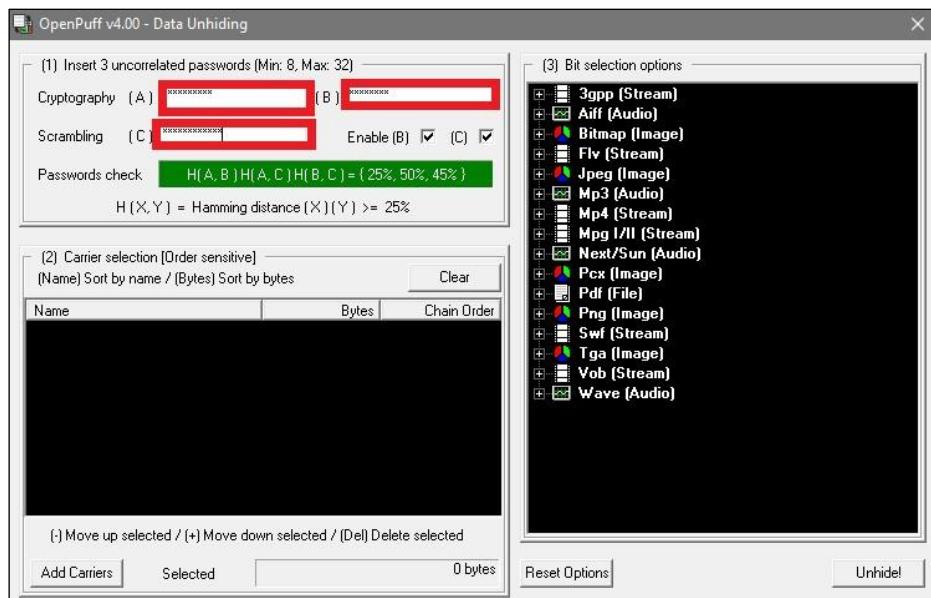
4.1.2 Menampilkan Pesan tersembunyi dalam *Image* menggunakan OpenPuff

- Buka *software* OpenPuff, pada bagian *Steganography* pilih tombol “*UnHide*”.



Gambar III. 15 OpenPuff Menu

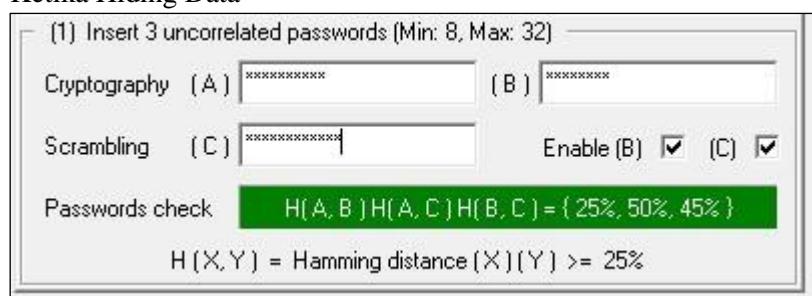
- Masukkan 3 *Password* yang telah ditentukan sebelumnya



Gambar III. 16 Insert Key

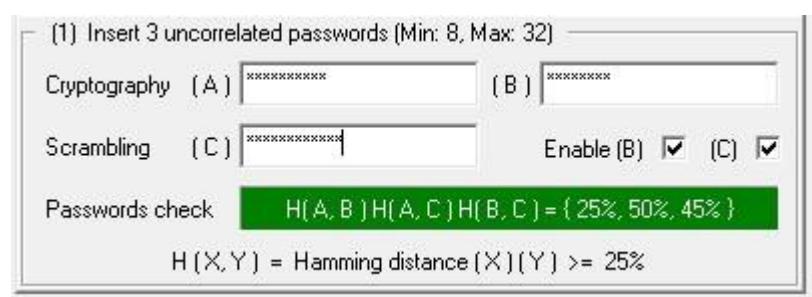
- c. Pastikan persentase dari *Password check* antara *Hiding Data* dan *Unhiding Data* telah sesuai dan cocok. Cara pengecekannya adalah dengan membandingkan persentase dari keterkaitan antara 3 *Password* tersebut seperti di bawah:

- Ketika Hiding Data



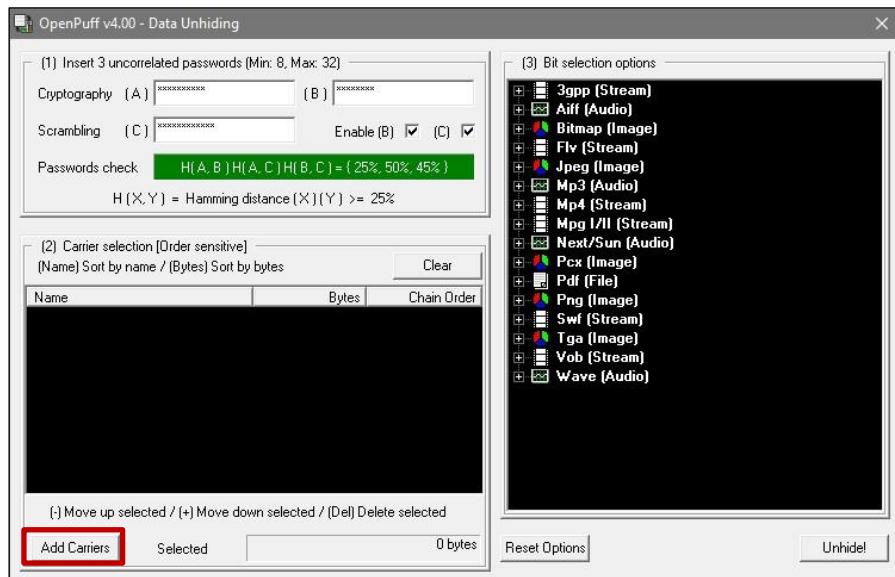
Gambar III. 17 Hiding Password Check

- Ketika Unhiding Data



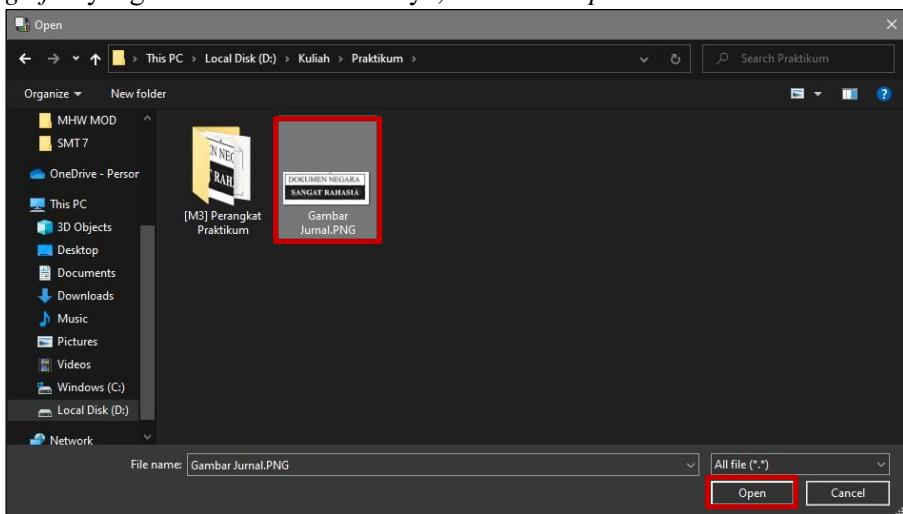
Gambar III. 18 Unhiding Password Check

- d. Selanjutnya pilih *stego file* yang telah dibuat sebelumnya dengan cara klik tombol “Add Carriers”.



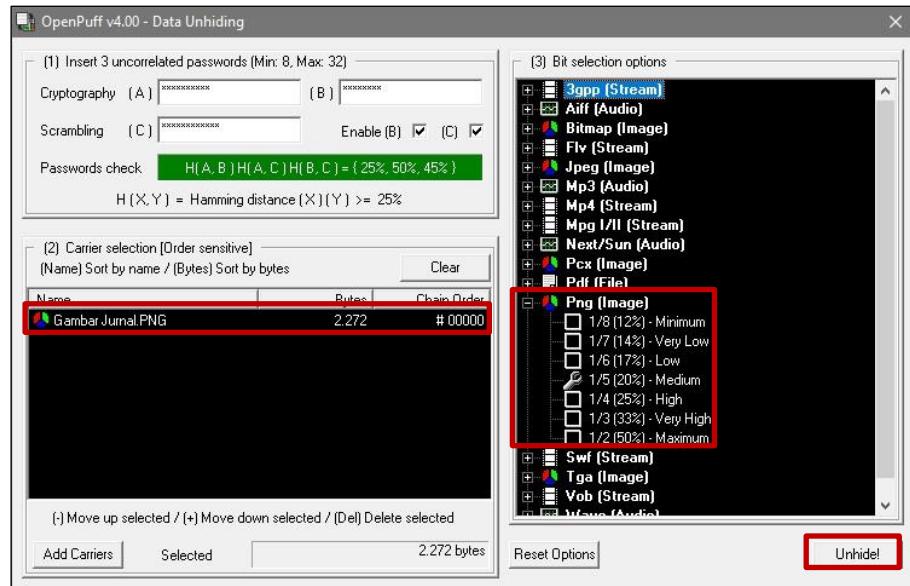
Gambar III. 19 Add Carriers

- e. Pilih *stego file* yang telah dibuat sebelumnya, lalu klik “Open”.



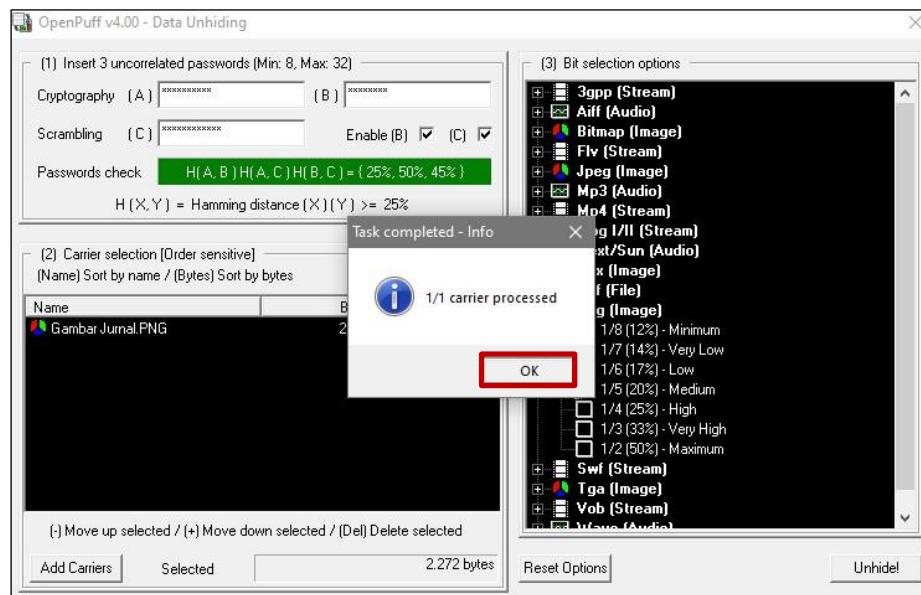
Gambar III. 20 Open Stego File

- f. Sama seperti pembuatan *stego file* pada tahap praktikum sebelumnya, *software* OpenPuff akan memberikan informasi untuk ukuran *file* gambar tersebut. Pemilihan kategori gambar disesuaikan dengan jenis gambar (opsional). Kemudian pilih “*UnHide!*”.



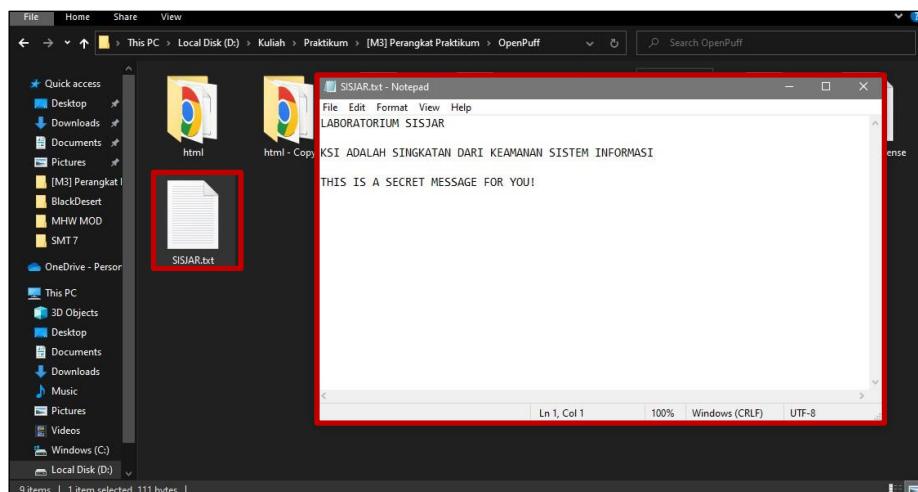
Gambar III. 21 Stego File Informamtion

- g. Pilih folder sebagai tempat penyimpanan hasil *extracting* dari *stego file* tersebut (pilih direktori baru diluar direktori hasil *stego file* sebelumnya) kemudian klik “OK”.



Gambar III. 22 Task Completed

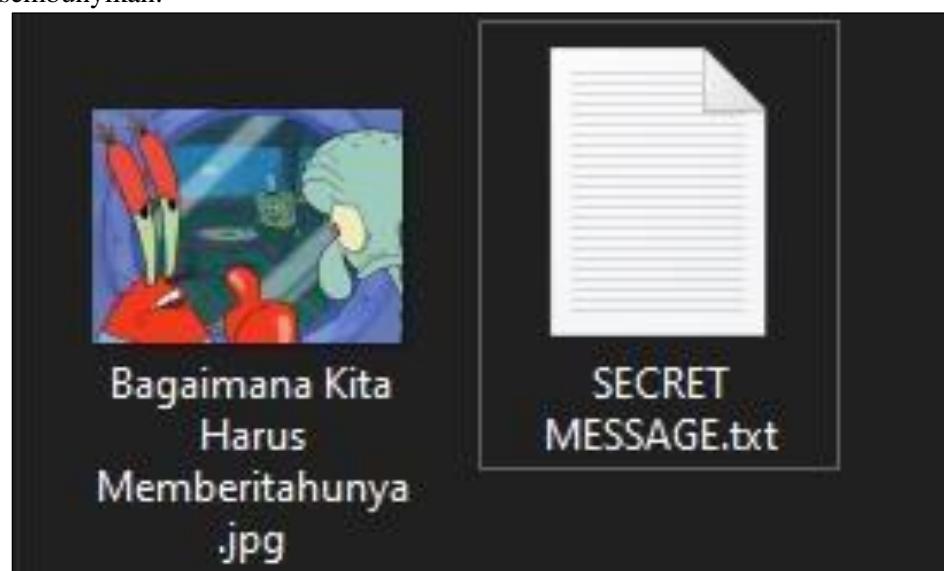
- h. Berikut adalah hasil dari proses *extracting* dari *stego file* jika berhasil dilakukan



Gambar III. 23 Output

4.1.3 Menyembunyikan Pesan dalam *Image* menggunakan StegHide UI

- Sebelum memulai tahapan untuk menyembunyikan pesan menggunakan StegHide UI, persiapkan dua buah *file* yang berupa *file image* dan *file text* yang akan disembunyikan.



Gambar III. 24 Raw File

Catatan : format *cover file* pada StegHide UI hanya mendukung .au, .bmp, .jpg, dan .wav.

- Setelah *file* sudah siap, buka *software* StegHide UI, kemudian pilih tab *Embed* untuk menyembunyikan *file text* ke dalam *file image*.



Gambar III. 25 StegHide UI Menu

- c. Pada section *Files* bagian *Cover file*, klik “*Browse...*” untuk mencari serta menambahkan direktori tempat *file image* yang akan dijadikan *stego file*. kemudian pada bagian *Embed File*, klik “*Browse...*” untuk mencari serta menambahkan direktori tempat *file* yang ingin disembunyikan, pada Lab Praktik ini. Jangan lupa pada bagian *Output File*, cari dan tambahkan direktori untuk tempat menyimpan *stego file*.



Gambar III. 26 Insert Files

- d. Untuk menambahkan *Password* pada *stego file*, silakan cek pada section Encryption bagian *Password*. Masukkan *Password* sesuai keinginan Anda.



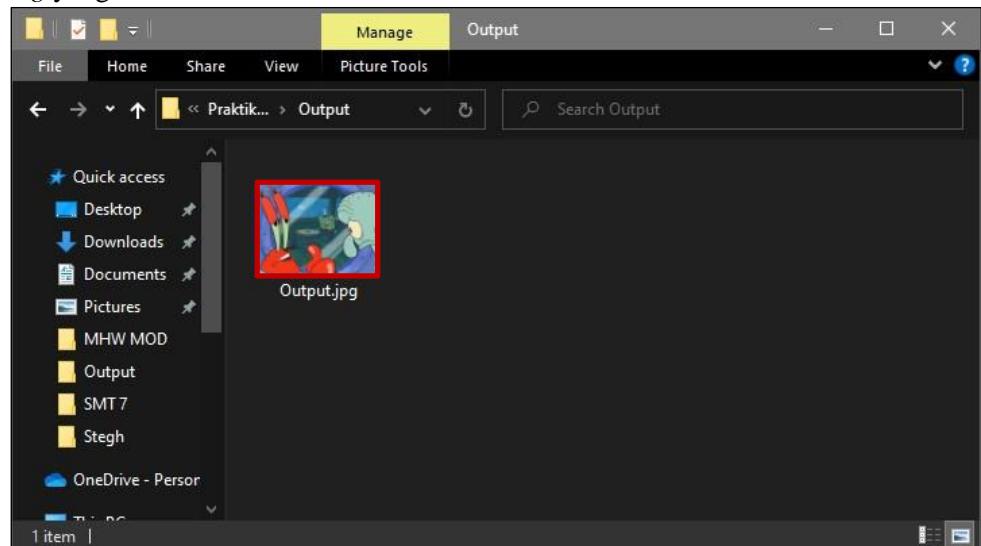
Gambar III. 27 Encryption

- e. Ketika sudah selesai semua, klik tombol “*Embed*” pada bagian bawah halaman agar *software* StegHide UI memulai proses *Embedding File*. Jika berhasil maka akan muncul sebuah pemberitahuan, lalu klik “OK”.



Gambar III. 28 Encrypted

- f. Periksa *stego file* yang telah dibuat. File “Output.jpg” merupakan hasil proses *Embedding* yang sudah berhasil dibuat.



Gambar III. 29 Stegofile

4.1.4 Menampilkan Pesan tersembunyi dalam *Image* menggunakan StegHide UI

- Buka *software StegHide UI*, kemudian pilih tab Extract untuk menampilkan pesan tersembunyi yang ada pada suatu *file image*.



Gambar III. 30 StegHide UI Menu

- Pada section *Files* bagian *Cover file*, klik “Browse...” untuk mencari serta menambahkan direktori tempat *stego file* disimpan yaitu “Output.jpg”. kemudian pada bagian *Output File*, cari dan tambahkan direktori yang digunakan untuk menyimpan hasil ekstrak pesan tersembunyi dari *stego file*.



Gambar III. 31 Choose Stego File and Output Folder

- Pada section *Encryption* bagian *Password*, masukkan *Password* yang sudah Anda ketahui. Kemudian klik tombol “Extract” untuk memulai proses *Extracting File*.

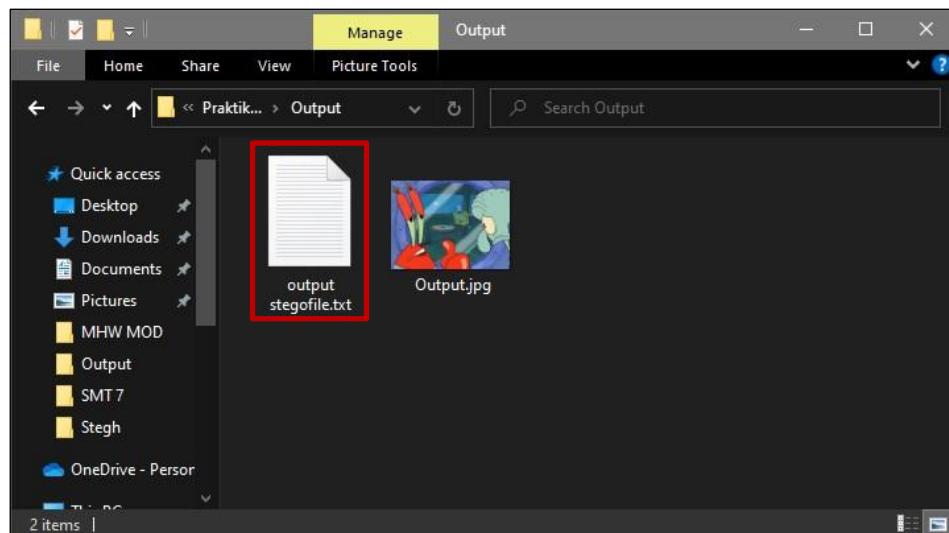


- d. Jika berhasil maka akan menampilkan hasil seperti pada gambar dibawah ini, kemudian klik “OK”.



Gambar III. 33 Stego File Extracted

- e. Periksa file hasil ekstrak dari stego file, berikut adalah hasil dari *extracting stego file* jika berhasil dilakukan



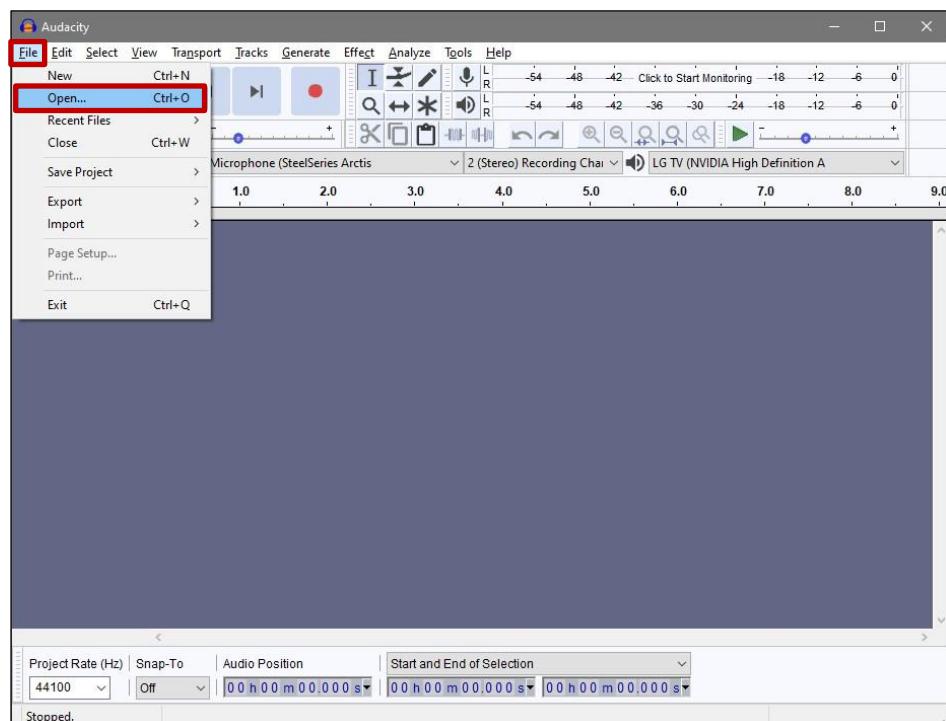
Gambar III. 34 StegHide UI Output

4.2 Steganography with Audio

4.2.1 Menampilkan Pesan tersembunyi dalam *Audio* menggunakan Audacity.

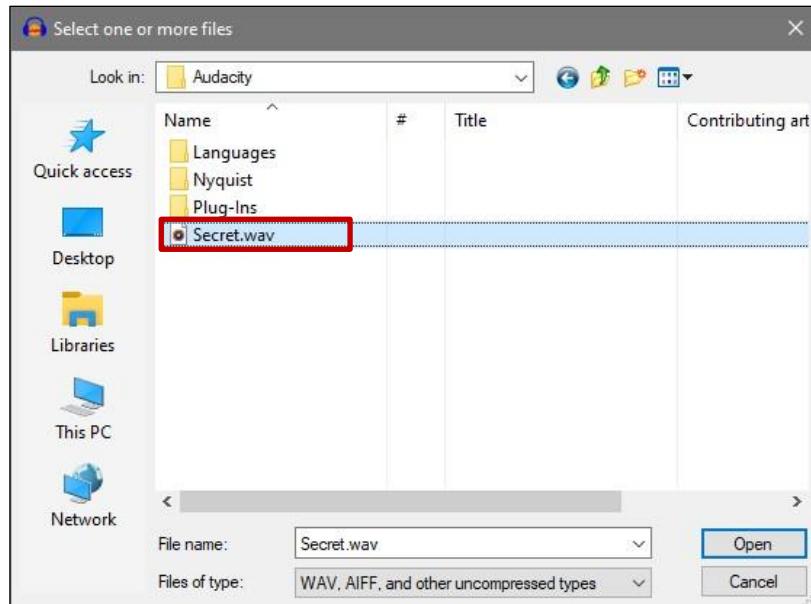
Berikut merupakan cara untuk menampilkan pesan tersembunyi dalam *audio* menggunakan Audacity.

- Buka aplikasi Audacity, pada tab *File*, pilih “Open...”.



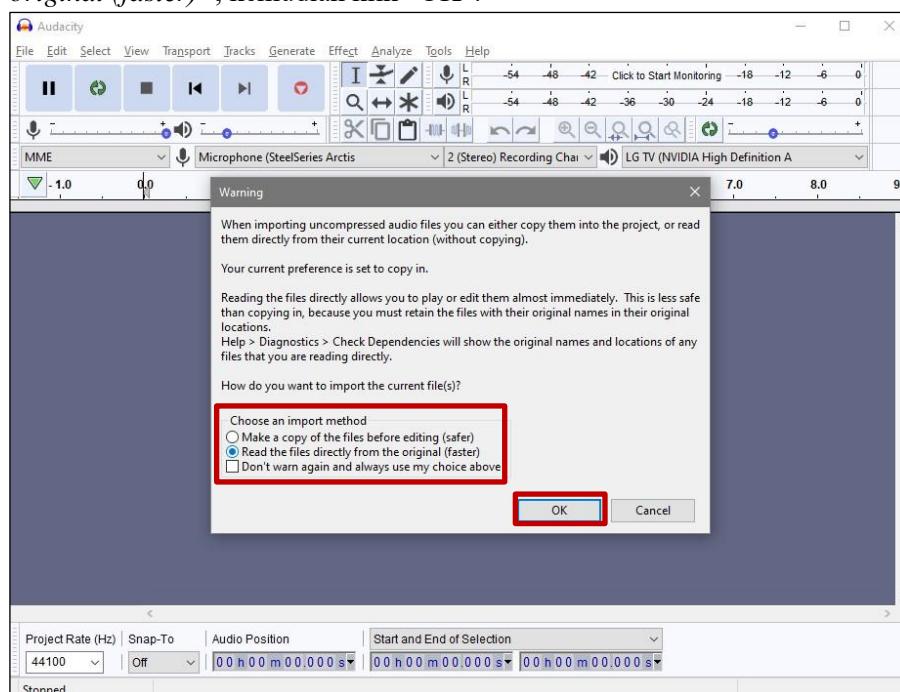
Gambar III. 36 Audacity File Tab

- Kemudian, pilih *audio* yang akan dianalisa, apakah memiliki pesan rahasia atau tidak. Klik “Open”.



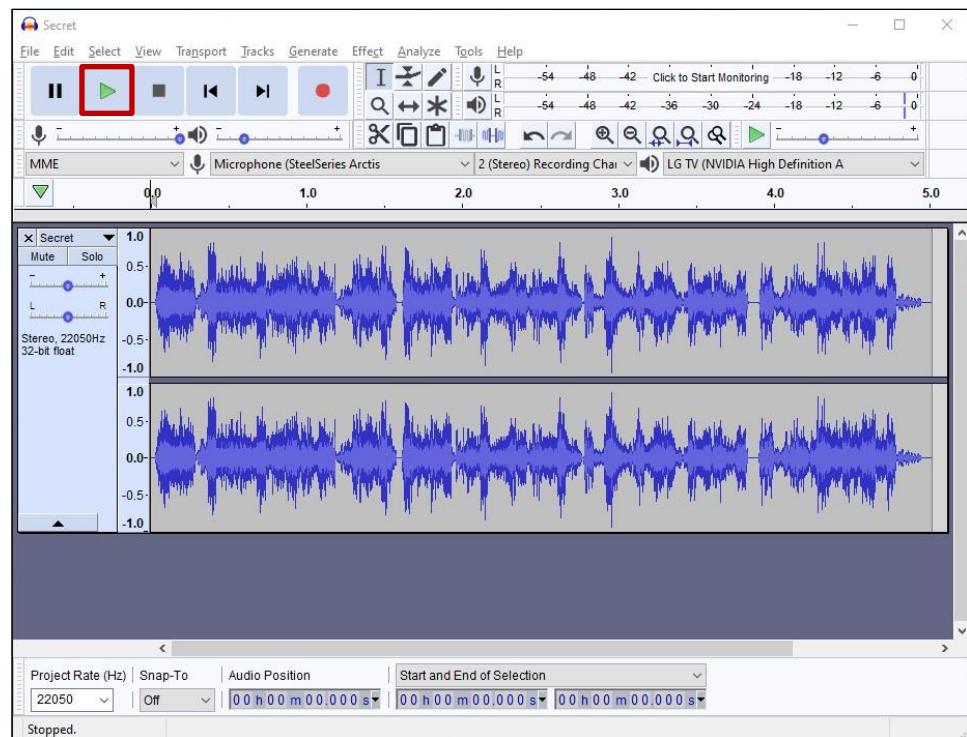
Gambar III. 37 Choosing Audio File

- c. Pada saat muncul Warning window seperti pada gambar di bawah. Pada bagian *Choose an import method*, pilih radio button “*Read the files directly from the original (faster)*”, kemudian klik “OK”.



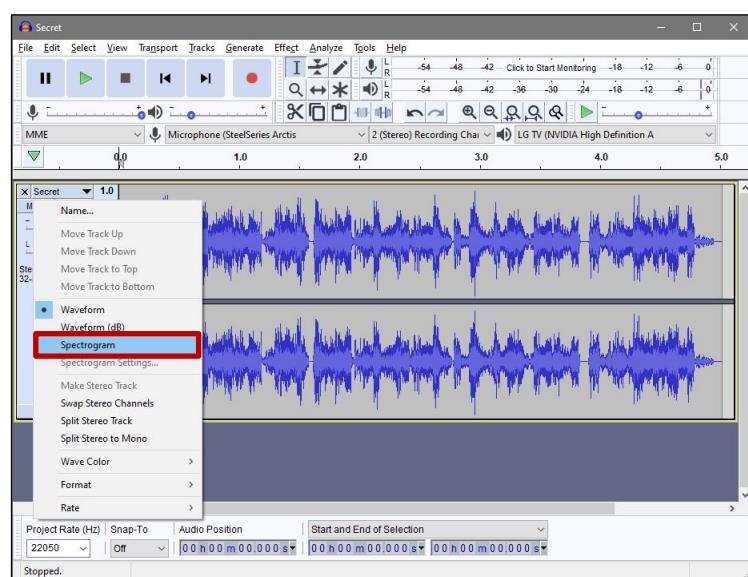
Gambar III. 38 Choosing Method

- d. Klik ikon play untuk memastikan apakah ada pesan rahasia yang tersembunyi pada file audio.



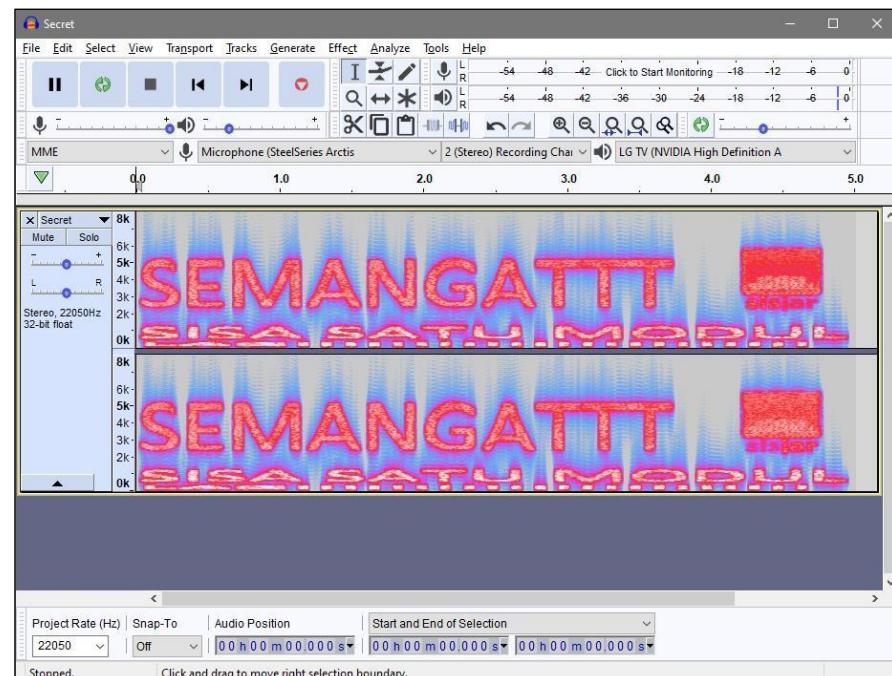
Gambar III. 39 Checking Audio

- e. Selanjutnya, pada layer *audio SISJAR* (tergantung nama file) , klik ikon “” dan pilih *Spectrogram*.



Gambar III. 40 Choosing Spectrogram

- f. Jika terdapat pesan rahasia, *audio* tersebut akan menampilkan spectrogram seperti gambar berikut.



Gambar III. 41 Message Output

Keterangan: Jika dilihat lebih lanjut spectrogram membentuk sebuah tulisan yaitu “SEMANGATT”.

V. Daftar Pustaka

1. Laboratorium Sistem Operasi dan Jaringan Komputer. (2021). Modul Praktikum Keamanan Sistem Informasi 2021. Bandung, Laboratorium Sistem Operasi dan Jaringan Komputer.