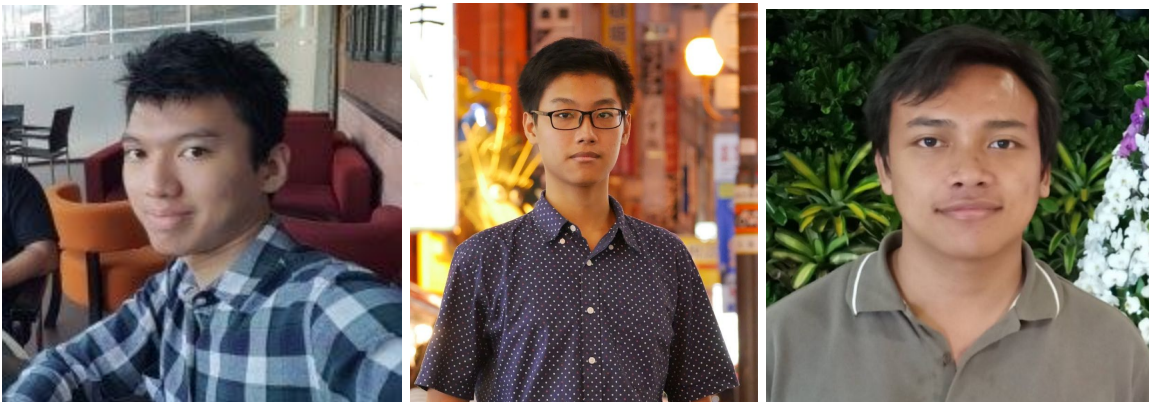


**LAPORAN TUGAS BESAR**  
**IF4020/KRIPTOGRAFI**  
**SEMESTER I 2020-2021**

**Steganografi**

Disusun oleh:

T. Antra Oksidian Tafly	13517020
Willsen Sentosa	13517036
Al Terra	13517145



**PROGRAM STUDI TEKNIK INFORMATIKA**  
**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**  
**INSTITUT TEKNOLOGI BANDUNG**  
**2020**

## DAFTAR ISI

<b>DAFTAR ISI</b>	<b>2</b>
<b>Teori Singkat</b>	<b>3</b>
Steganografi	3
Metode LSB	3
Metode BPCS	4
Image Steganography	6
Audio (.wav) Steganography	6
Video Steganography	6
<b>Perancangan dan Implementasi</b>	<b>6</b>
Steganografi Gambar	6
Steganografi Video	6
Steganografi Audio	6
<b>Pengujian program dan analisis hasil</b>	<b>7</b>
Steganografi Gambar	7
Steganografi Audio	9
Hasil steganografi	9
Kasus khusus	13
<b>Kesimpulan dari hasil implementasi</b>	<b>13</b>

## 1. Teori Singkat

### a. Steganografi

Steganografi berasal dari bahasa Yunani “steganos” yang berarti tersembunyi dan “graphien” yang berarti tulisan. Maka dari itu, steganografi dapat diartikan sebagai “Tulisan Tersembunyi”. Secara formal steganografi merupakan ilmu dan seni menyembunyikan pesan rahasia dengan suatu cara sedemikian sehingga tidak seorang pun yang mengetahui keberadaan pesan tersebut, dan berguna sebagai pendamping dari kriptografi. Dimana kriptografi bertujuan agar hanya orang tertentu yang mengetahui makna sebuah pesan, steganografi bertujuan agar hanya orang tertentu yang mengetahui keberadaan sebuah pesan. Steganografi bekerja dengan cara menyisipkan sebuah pesan ke dalam sebuah medium dengan suatu cara tertentu agar pesan tidak terlihat keberadaannya.

*Apparently neutral's protest is thoroughly discounted and ignored.  
Isman hard hit. Blockade issue affects pretext for embargo on by-  
products, ejecting suets and vegetable oils.*

Ambil huruf kedua setiap kata, diperoleh pesan berikut: *Pershing  
sails from NY June 1.*

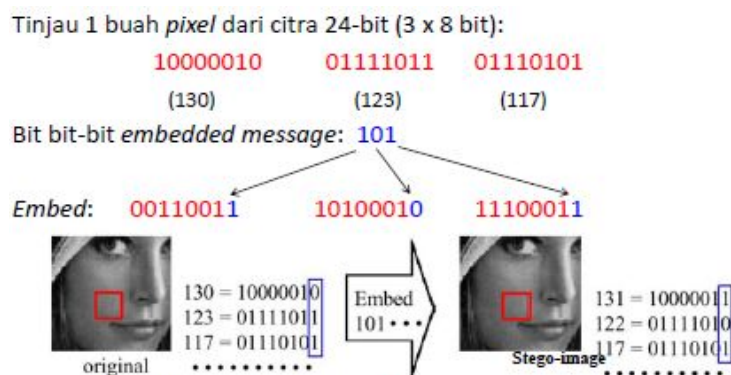
Gambar 1: Steganografi dengan menggunakan karakter kedua tiap kata.

Berikut beberapa istilah dalam steganografi:

- Embedded message / secret message : pesan yang disembunyikan
- Cover object : Media yang digunakan untuk menyembunyikan pesan
- Stego object : Media yang sudah disisipkan pesan
- Stego key : Kunci yang digunakan untuk ekstraksi pesan

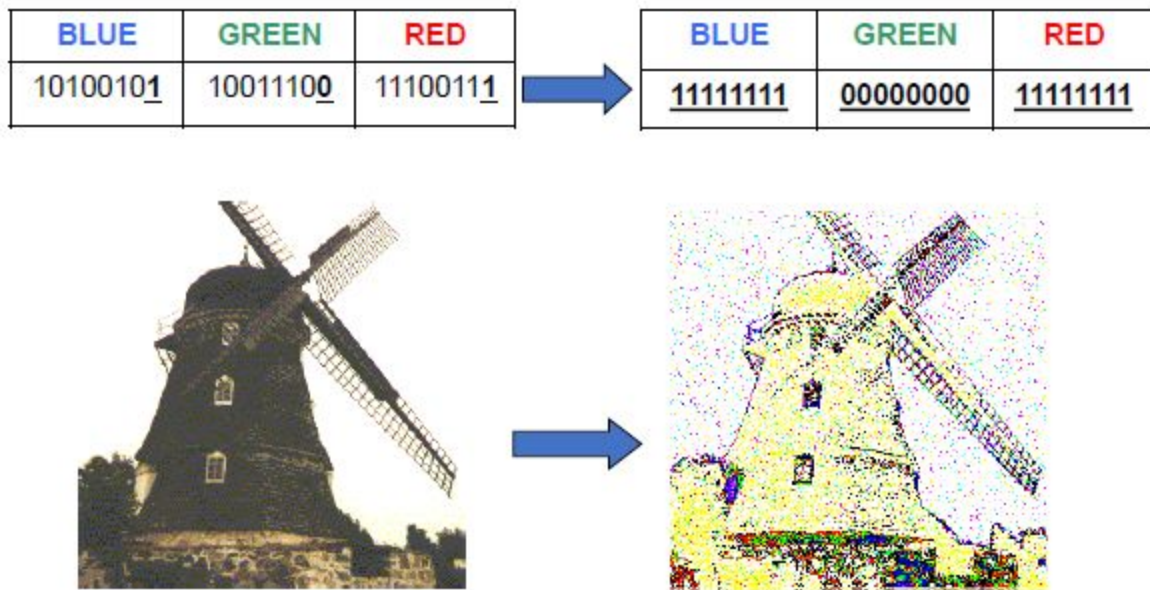
### b. Metode LSB

Metode *Least Significant Bit* (LSB) merupakan metode paling sederhana dan populer dalam steganografi media digital. Metode ini memanfaatkan kelemahan indra manusia yang tidak begitu peka pada perubahan kecil dalam media digital. Metode ini membagi bit-bit pesan dan menyisipkannya pada bit-bit citra dengan nilai paling kecil (paling insignifikan).



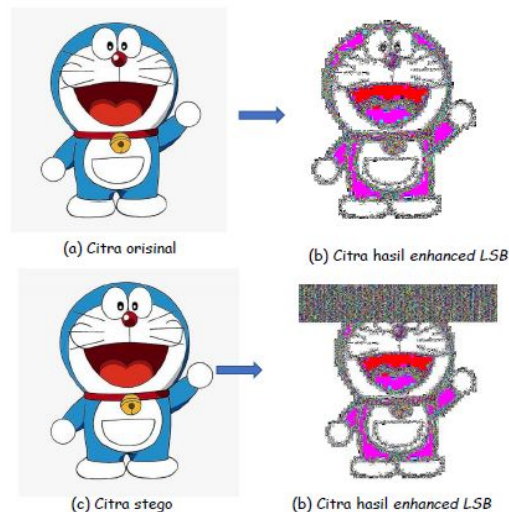
Gambar 2: Visualisasi metode LSB

Metode LSB ini walau tidak dapat dideteksi secara kasat mata, metode ini dapat dideteksi dengan metode Enhanced LSB. Dimana semua bit dari sebuah byte diubah menjadi sama dengan bit terakhir (bit paling insignificant).



Gambar 3: Visualisasi metode Enhanced LSB

Jika diaplikasikan ke sebuah stego-image, metode enhanced LSB ini akan menunjukkan bagian yang memiliki pesan sebagai bagian yang memiliki banyak noise. Hal ini dapat berarti pendeteksian pesan jika bit-bit pada cover-image pada awalnya tidak memiliki banyak noise.

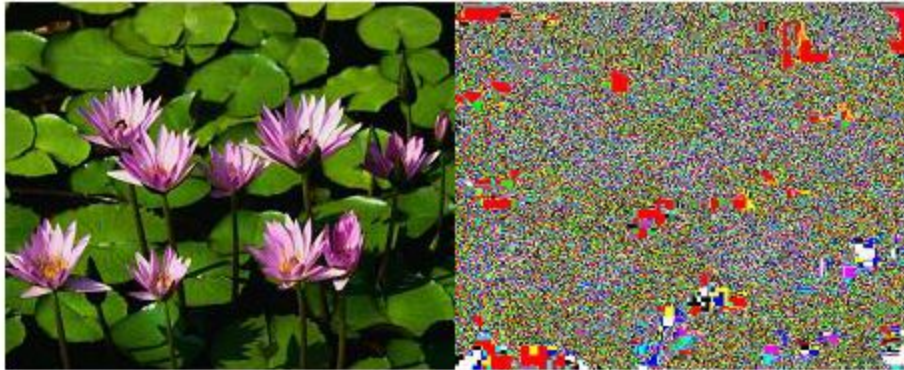


Gambar 4: Pendeteksian pesan melalui Enhanced LSB

### c. Metode BPCS

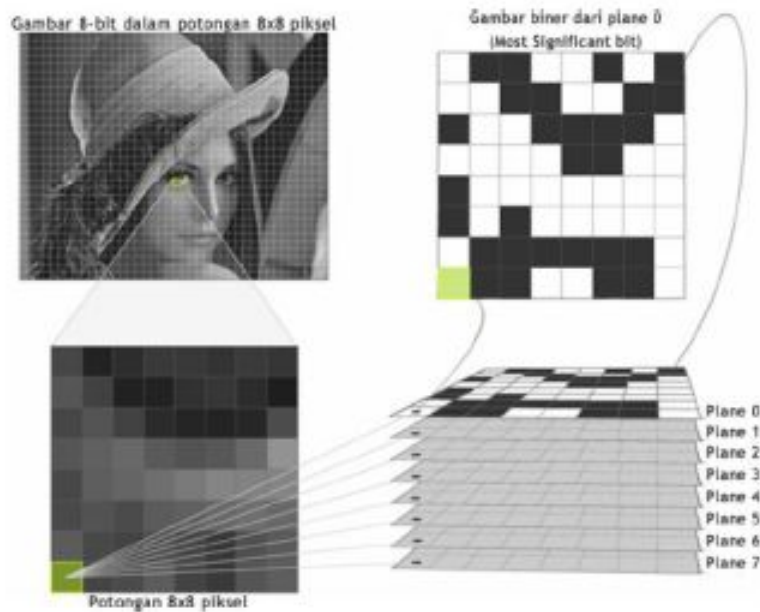
Metode Bit Plane Complexity Segmentation adalah sebuah metode steganografi yang dikembangkan oleh Eiji Kawaguchi dan R.O. Eason pada tahun 1997. Metode ini

memanfaatkan kelemahan yang dimiliki metode Enhanced LSB dimana jika gambar pada awalnya memiliki banyak noise pada hasil Enhanced LSB nya, maka pesan tidak akan terdeteksi.



Gambar 5: Contoh gambar yang memiliki banyak noise

Metode BPCS memanfaatkan kelemahan ini dengan memecah gambar menjadi bagian-bagian  $8 \times 8$  pixel lalu mengidentifikasi bagian yang memiliki banyak noise dan memasukkan pesan hanya pada bagian-bagian tersebut. Pengidentifikasian bagian yang memiliki banyak noise tersebut dilakukan dengan memecah bagian itu menjadi 8 bit-plane dan menghitung jumlah perubahan bit.



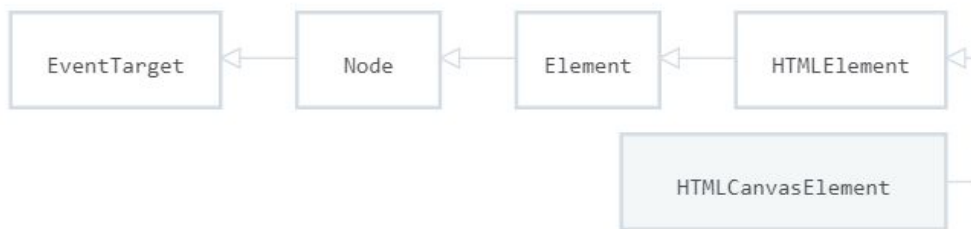
Gambar 6: Segmentasi gambar dalam metode BPCS



#### d. Image (.png dan .bmp) Steganography

Steganografi dalam file image (disini file image tersebut adalah file .png dan .bmp) menggunakan Web API HTMLCanvasElement yang akan memanipulasi dan menampilkan gambar (baik sebelum disisipkan pesan maupun sesudah).

The **HTMLCanvasElement** interface provides properties and methods for manipulating the layout and presentation of **<canvas>** elements. The **HTMLCanvasElement** interface also inherits the properties and methods of the **HTMLElement** interface.

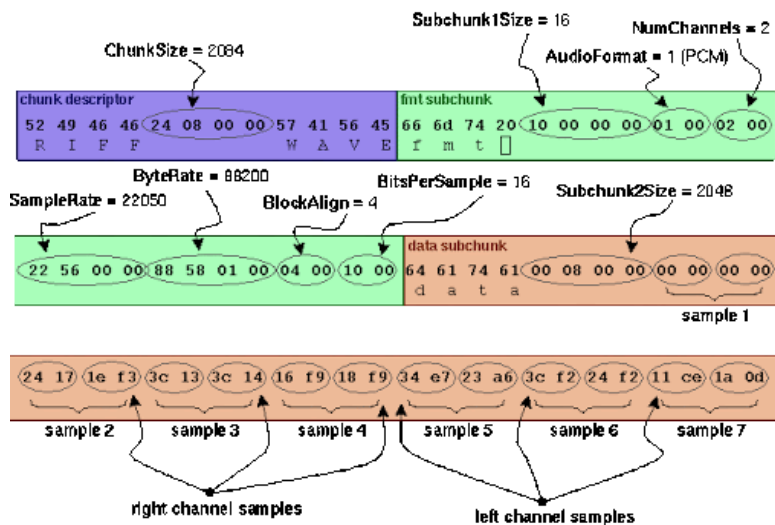


Gambar 7: Penjelasan dan visualisasi HTMLCanvasElement

Di dalam API tersebut terdapat sebuah method getImageData() dan putImageData() yang dapat mengambil dan menaruh bit-bit data RGB dari pixel-pixel tertentu.

#### e. Audio (.wav) Steganography

Steganografi dalam file audio (disini file audio tersebut adalah file .wav) menggunakan standar struktur data yang dimiliki oleh semua file .wav untuk menentukan bagian yang dapat diubah lalu menggunakan metode LSB untuk menyisipkan pesan (metode BPCS tidak perlu diimplementasikan karena Enhanced LSB tidak bisa digunakan untuk file audio).



Gambar 8: Struktur data file .wav

Dilihat dari struktur data tersebut, dapat disimpulkan bahwa 5 byte setelah byte “data” merupakan byte-byte isi dari .wav yang bisa dimanipulasi. Oleh karena itu, pesan disisipkan di bagian tersebut.

#### f. Video Steganography

Steganografi dalam file video (format berupa .avi) menggunakan struktur data video avi secara umum untuk menyisipkan text ke dalam video. Penyisipan tersebut dengan menggunakan metode *Least Significant Bit* (LSB), yaitu memasukkan bit pesan pada bit video yang paling tidak mengubah kualitas video, yaitu 1 bit terakhir pada tiap pixel.

Offset	0	1	2	3	4	5	6	7	-	8	9	A	B	C	D	E	F	ASCII
00000000	52	49	46	46	D2	7A	00	00		41	56	49	20	4C	49	53	54	RIFFTz..AVI LIST
00000010	D2	04	00	00	68	64	72	6C		61	76	69	68	38	00	00	00	T...hdrlavih8...
00000020															10	08	00	PT...0.....
00000030															BA	05	00	b.....e...
00000040	AS	00	00	00														r...a.....
00000050	00	00	00	00						4C	49	53	54	86	04	00	00	.....LIST†...
00000060	73	74	72	6C	73	74	72	68		38	00	00	00	76	69	64	73	strlstrh8...vids
00000070	6D	72	6C	65	00	00	00	00		00	00	00	00	00	00	00	00	mrle.....
00000080	05	00	00	00	64	00	00	00		00	00	00	00	62	00	00	00	....d.....b...
00000090	BA	05	00	00	10	27	00	00		00	00	00	00	00	00	00	00	e....'.....

Berikut adalah sekilas struktur data dari video avi jika dilihat menggunakan hex editor. Data dari video akan ditandai dengan sebuah tag MOVI, lalu akan dimasukkan bit pesan dengan algoritma LSB setelah tag MOVI tersebut

## 2. Perancangan dan Implementasi

### a. Steganografi Gambar

Penyisipan data diawali dengan menyisipkan delimiter untuk menandai tempat untuk meletakkan extension file dan akhir dari pesan. Mulainya extension file diawali dengan ( ) dan akhir dari pesan ditandai dengan { }. Setelah itu pesan akan dimasukkan karakter per karakter ke suatu array dan akan diubah ke nilai karakternya. Setiap nilai karakter akan diubah ke representasi biner.

Setelah nilai biner didapatkan maka pengguna dapat memasukkan stego key untuk melakukan pengacakan atau dapat melakukannya secara berurut sekuensial.

Setelah itu dilakukan pemilihan metode apakah akan menggunakan LSB atau BPCS. Dalam metode LSB sekuensial teks akan dienkripsi dan bit-bit pesan akan disisipkan ke Least Significant Bit dari message tersebut. Dalam metode LSB acak maka angka akan diacak dari 0, panjang carrier-1 dan akan disimpan di pixel tersebut.

Dalam metode BPCS pengguna kembali lagi dapat memilih apakah metoda yang digunakan sekuensial atau acak. Dalam BPCS sekuensial program akan mengecek setiap bitplane, jika ditemukan bitplane yang kompleks maka akan diganti dengan 8 byte pesan. Apabila hasil

perubahan tersebut memiliki kompleksitas rendah maka diperlukan konjugasi dengan checkerboard dan pencatatan pada conjugation map. Conjugation map digunakan dengan mencatatnya pada bidang melebihi tengahnya. Contohnya apabila ada 10 bit plane dan data yang tidak kompleks disimpan pada bit plane ke 1 maka conjugation map disimpan pada bit plane ke 6, jika ke 2 maka ke 7 dan seterusnya. Apabila dilakukan secara random maka tidak dilakukan pengecekan apakah plane kompleks atau tidak namun tetap dilakukan pencatatan apabila bit pesan tidak kompleks.

Pada proses dekripsi maka akan dicek setiap bit plane dan dibandingkan dengan mirrornya dan apabila kembar maka akan dikonjugasikan dengan checkerboard. Setelah data didapatkan maka akan diambil datanya melalui bitplane tersebut.

### **b. Steganografi Video**

Implementasi dalam steganografi video avi dilakukan dengan menghitung max size file/text yang dapat dimasukkan dalam sebuah video. Setelah itu, pesan akan dienkripsi menggunakan Vigenère Cipher sesuai dengan kunci yang dimasukkan pengguna dan diubah menjadi binary data. Lalu akan dicari tag MOVI pada file video AVI. Setelah itu, akan dilakukan iterasi untuk memasukkan bit-bit pesan pada data video yang merupakan Least Significant Bit, yaitu bit terakhir pada setiap byte pixel.

Untuk pengambilan data pada video sendiri, aplikasi akan menghitung berapa banyak data yang sudah disisipkan pada video, lalu akan ditelusuri dan diambil bit" pada video tersebut yang merupakan bit terakhir setiap byte pixel. Setelah itu, pesan akan di dekripsi oleh Vigenère Cipher decryption sehingga menghasilkan text yang dimasukkan pada video.

### **c. Steganografi Audio**

Ada 3 hal yang harus diperhatikan dalam steganografi ini, yaitu:

- Apakah pesan bersifat teks atau file?
- Apakah pesan diacak atau sekuensial?
- Jika pesan berupa file, apa saja metadata nya?

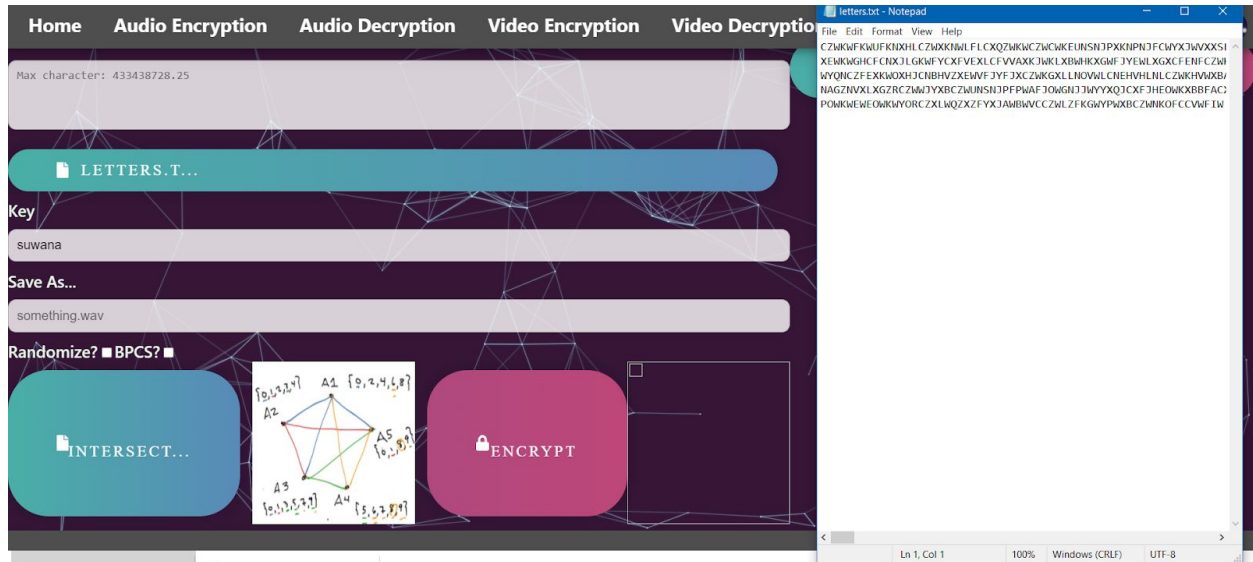
3 hal ini diselesaikan dengan cara memberikan sebuah standar format dalam steganografi audio, yaitu:

- Pada byte pertama, diisi 1 atau 0. 1 jika pesan berupa file, 0 jika pesan berupa teks.
- Jika byte pertama 1 (file berupa teks), 4 byte berikutnya menentukan berapa panjang nama file (anggap panjang nama file = n), n byte berikutnya merupakan nama file, 4 byte berikutnya menentukan berapa byte panjang pesan.
- Untuk pesan berupa teks, akhir dari pesan ditandai dengan byte bernilai 0 atau 1.
- Baik untuk pesan file / teks, byte berakhiran 0 berarti pesan ditulis secara sekuensial. Byte berakhiran 1 berarti pesan ditulis secara acak.

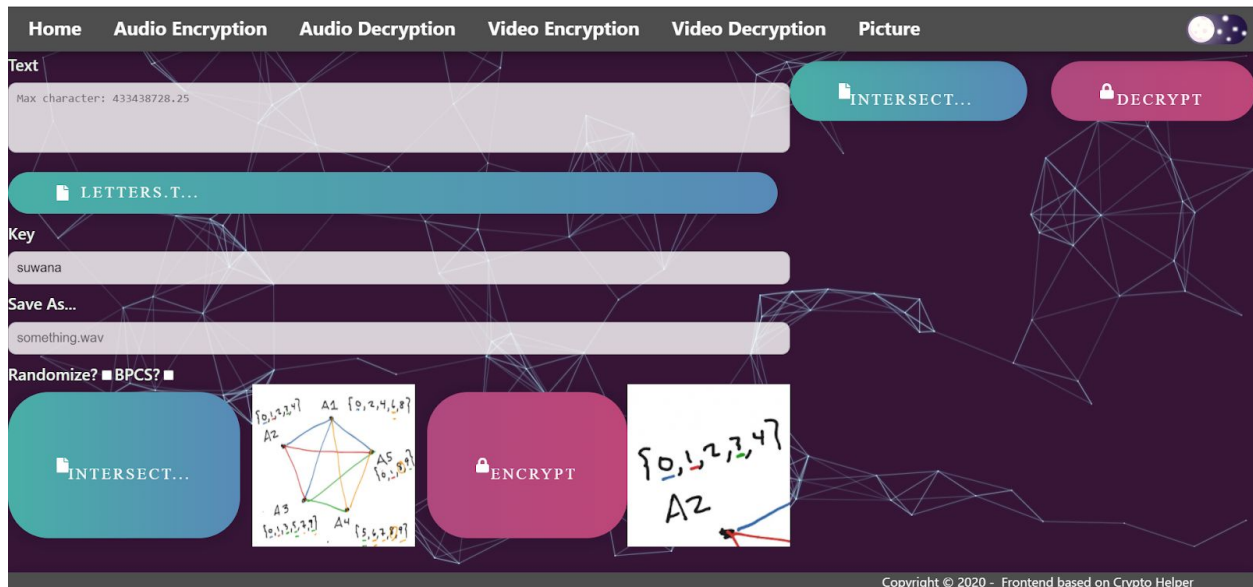


### 3. Pengujian program dan analisis hasil

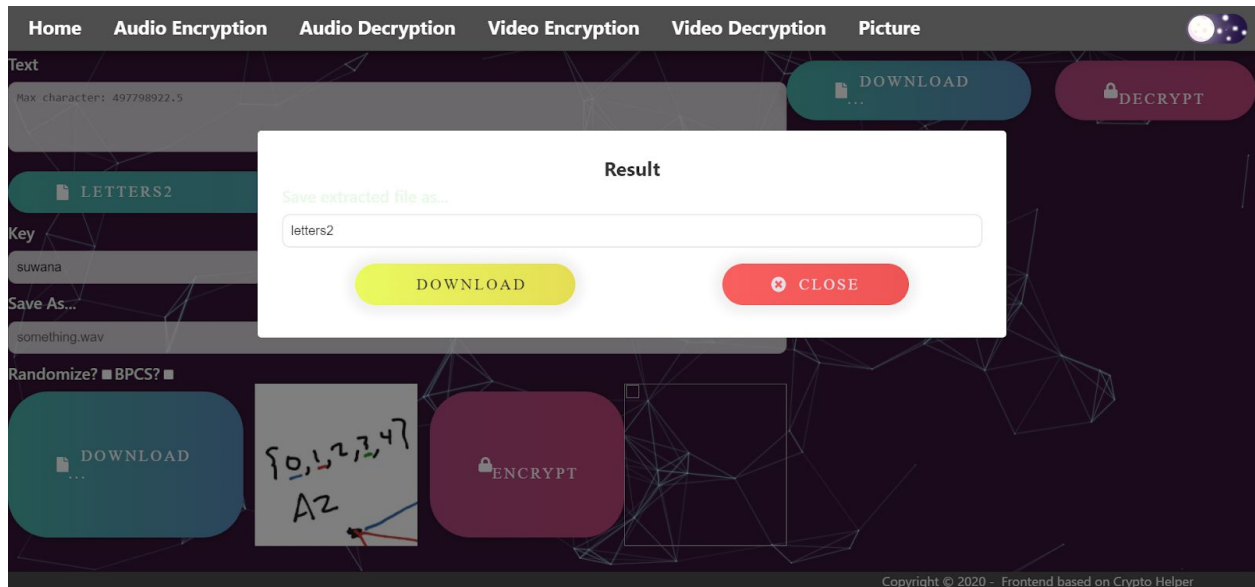
#### a. Steganografi Gambar



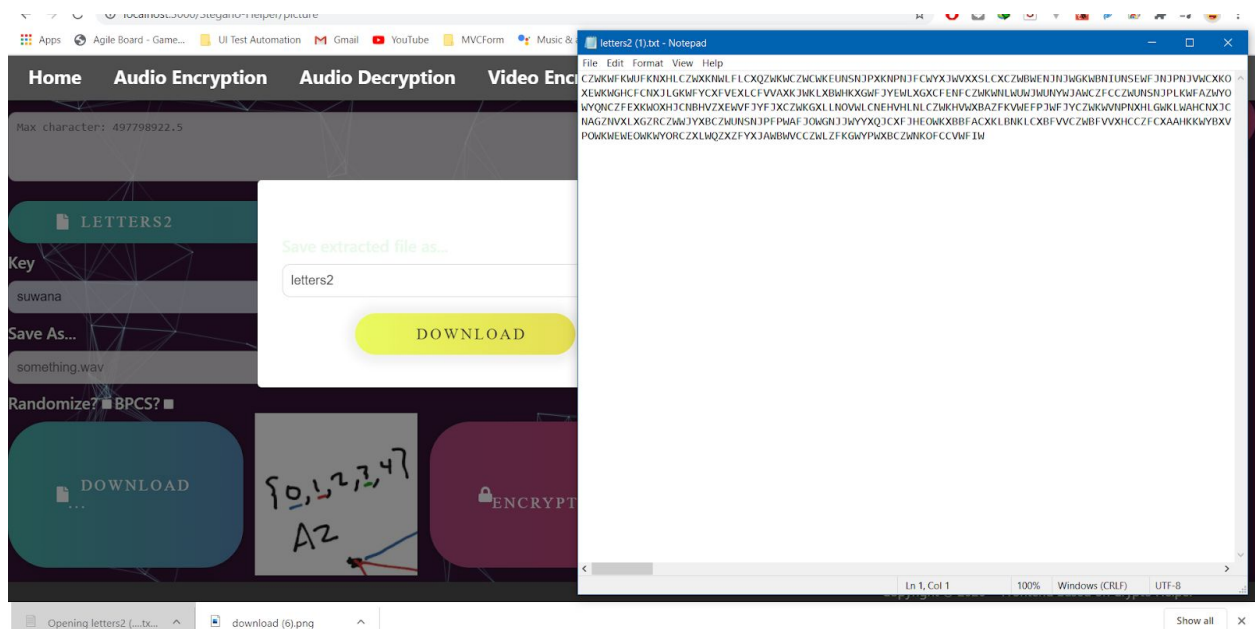
Injeksi "Letters.txt" ke gambar



Hasil injeksi dapat diambil dari gambar di kanan



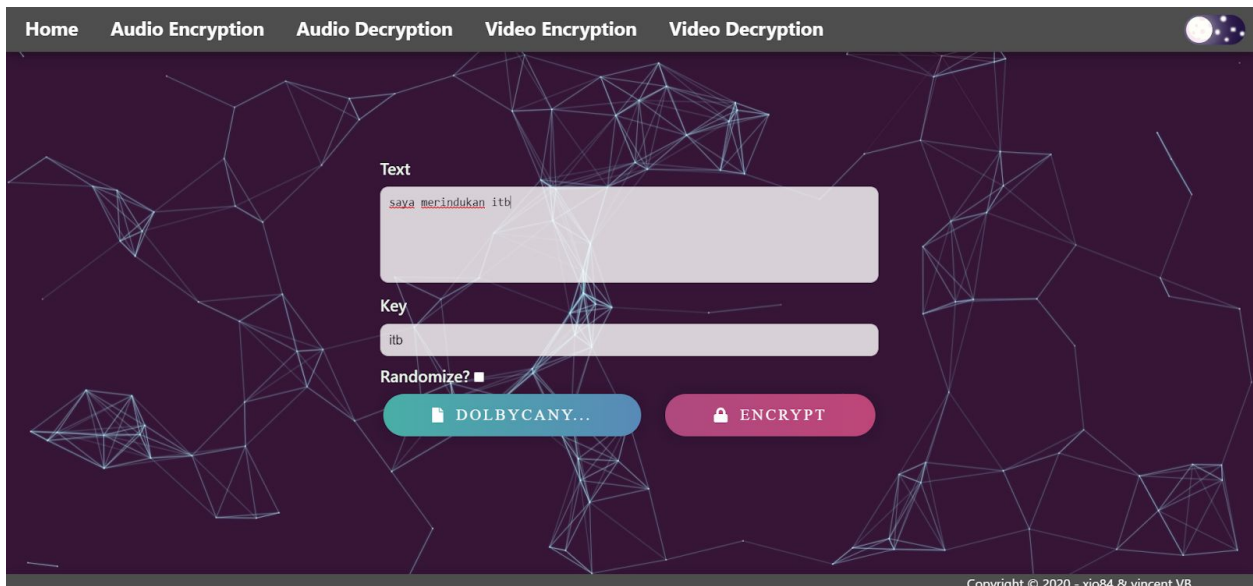
Hasil ekstraksi dapat disimpan dengan nama tertentu



Hasil ekstraksi bersifat lossless

## b. Steganografi Video

### I. Hasil Stegano pada video



Home Audio Encryption Audio Decryption Video Encryption Video Decryption

Text  
saya merindukan itb

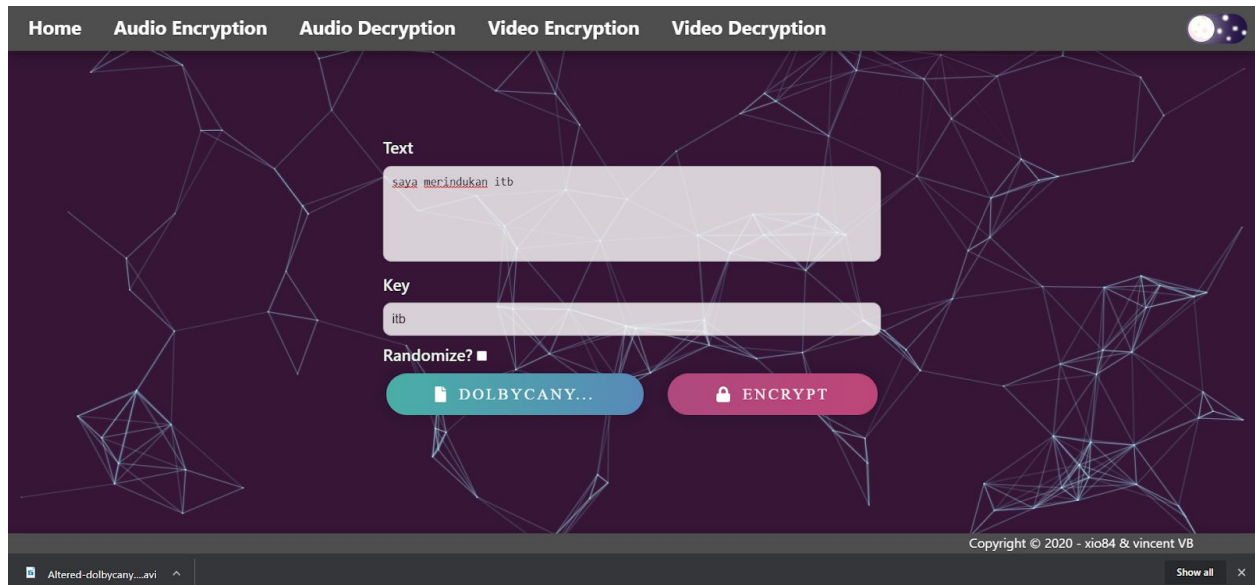
Key  
itb

Randomize? ☐

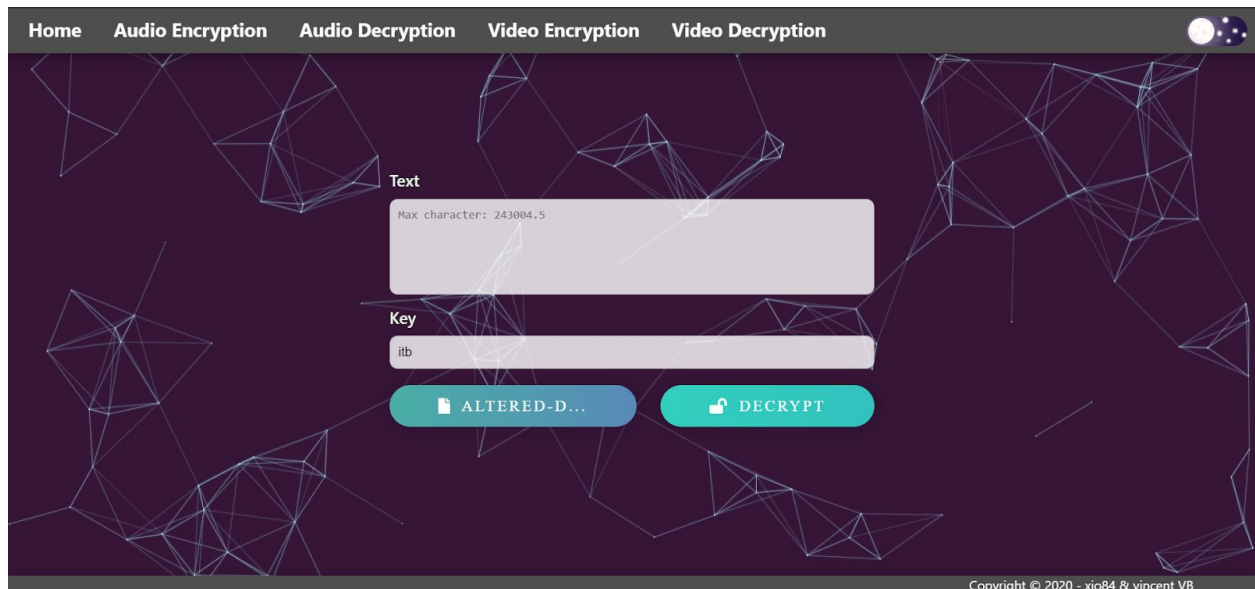
DOLBYCANY... ENCRYPT

Copyright © 2020 - xio84 & vincent VB

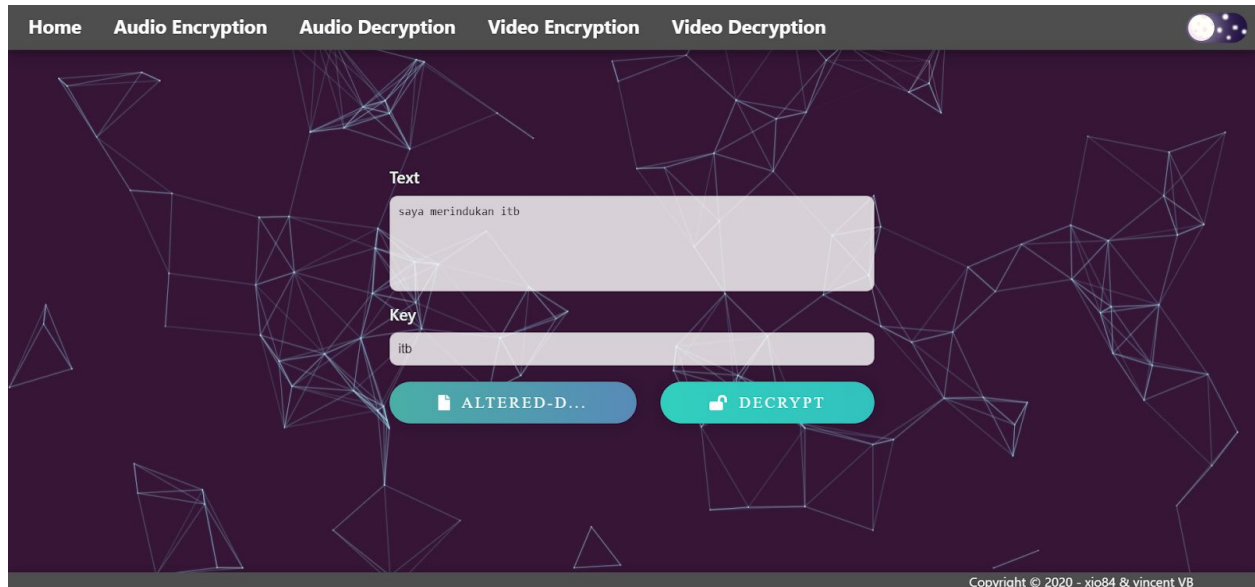
Injeksi dan enkripsi teks “saya merindukan itb” ke dalam file “dolbycanyon.avi”



Hasil stegano "dolbycanyon.avi" terdownload menjadi "Altered-dolbycanyon.avi"



Pengambilan teks dalam file "Altered-dolbycanyon.avi"



Hasil teks yang diambil dari file "Altered-dolbycanyon.avi"

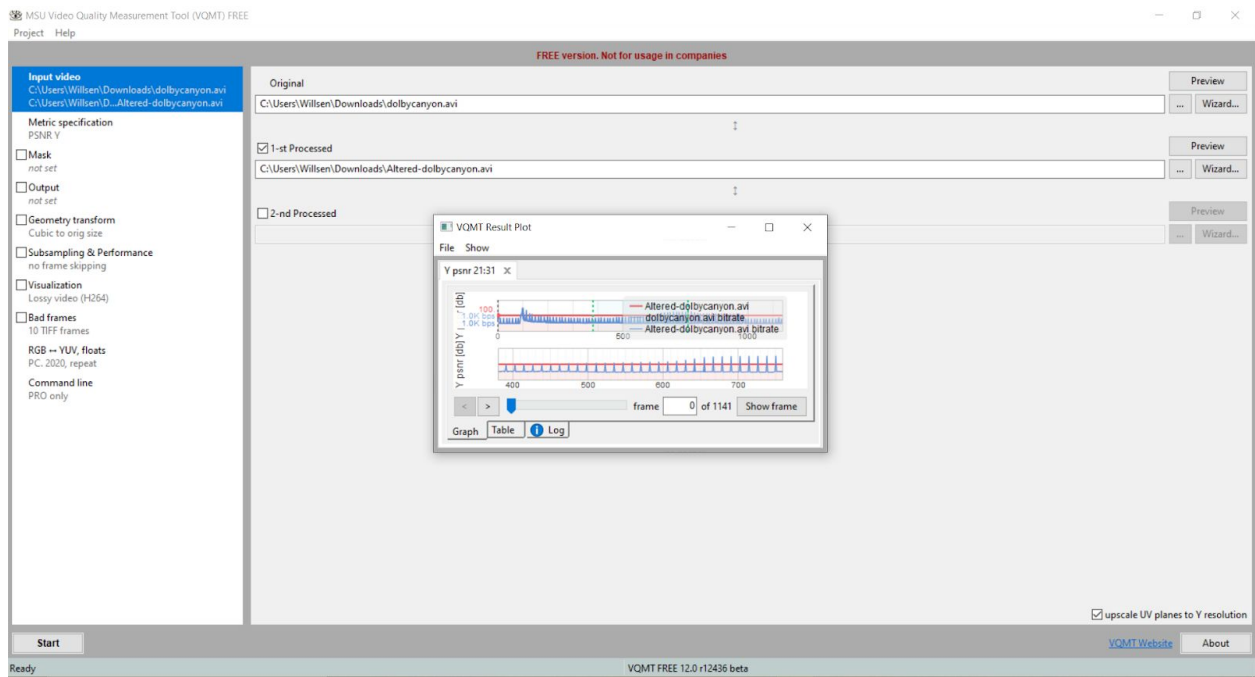


Video dolbycanyon.avi

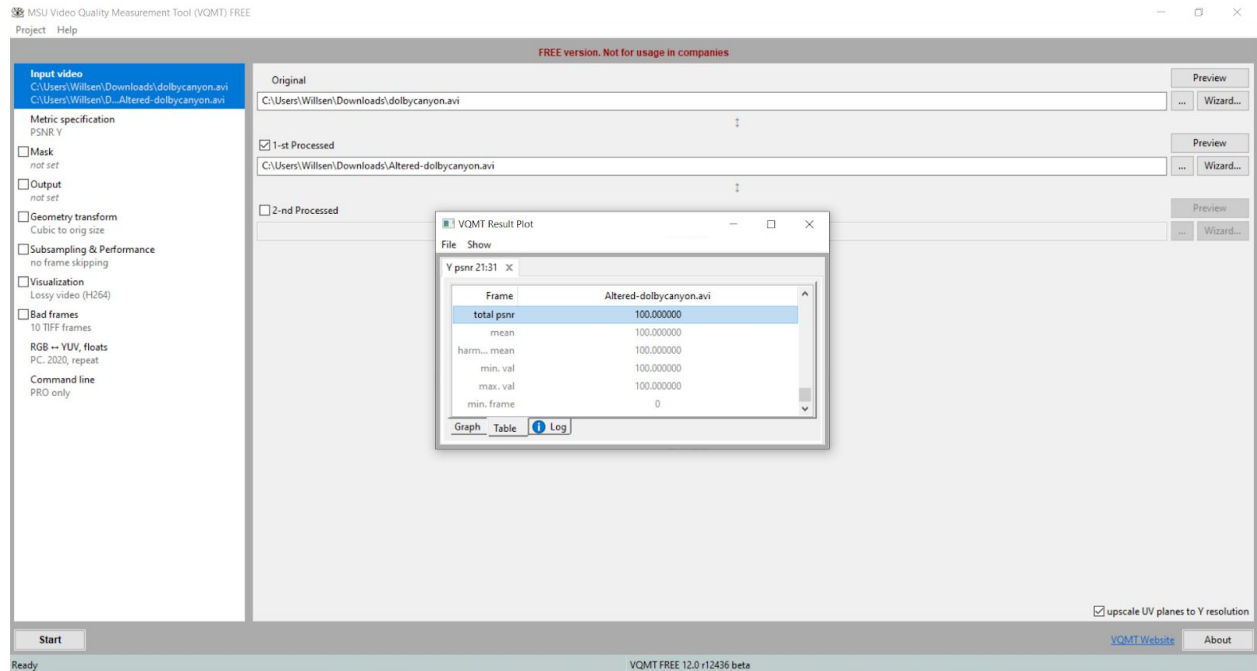




Video Altered-dolbycanyon.avi

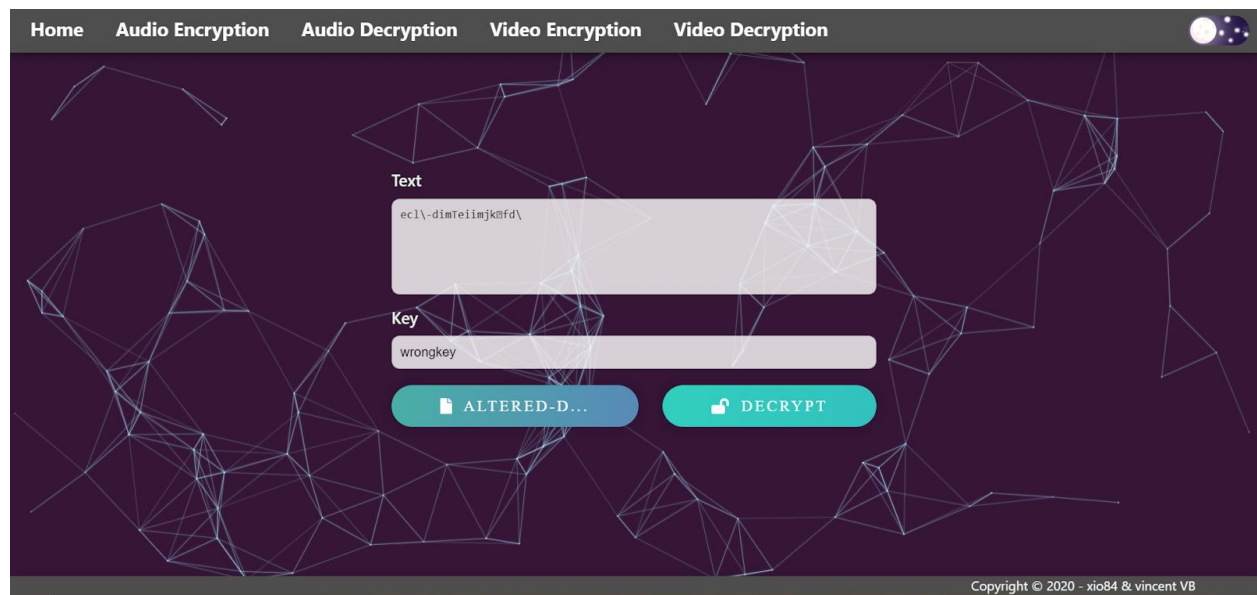


Hasil perhitungan PSNR



Hasil perhitungan PSNR, dapat dilihat bahwa PSNR > 40

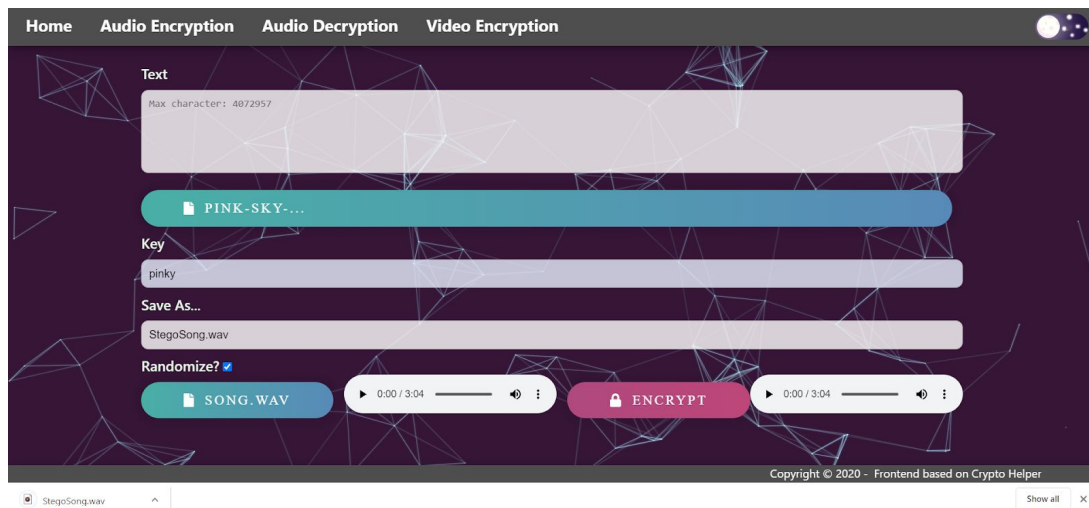
## II. Kasus Khusus



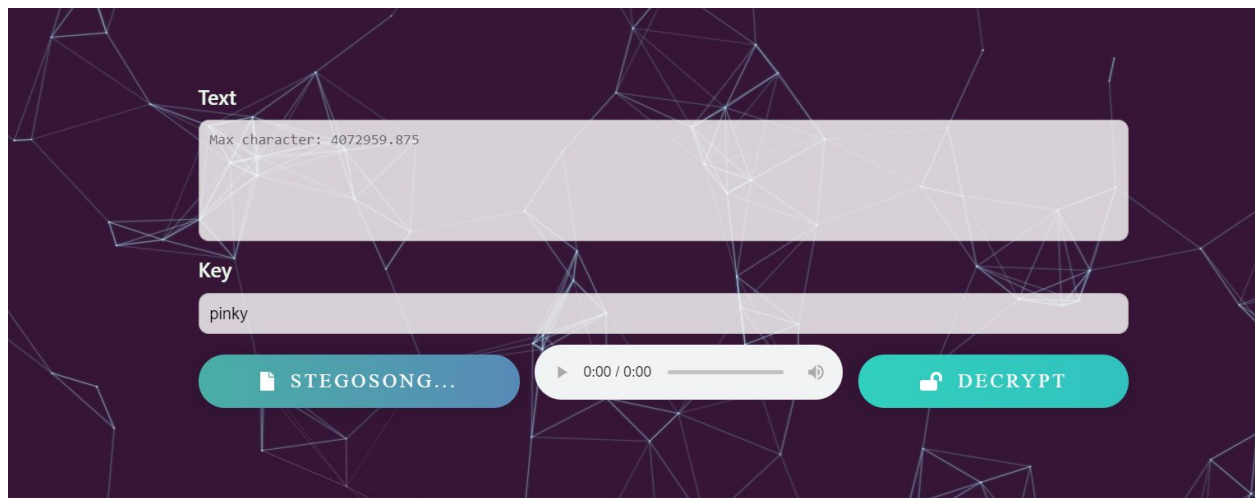
Salah Kunci

### c. Steganografi Audio

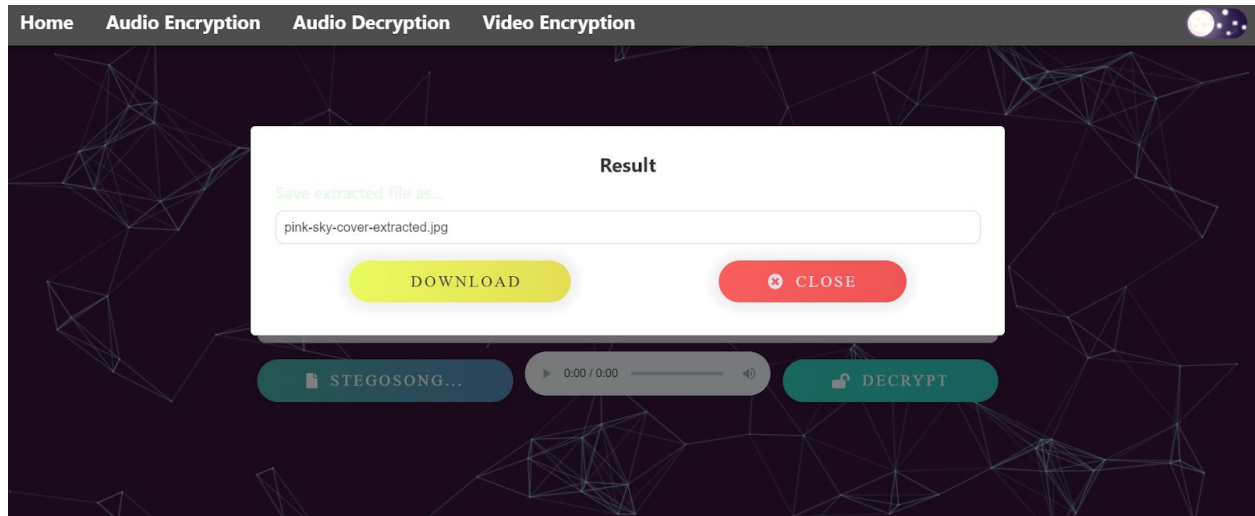
#### i. Hasil steganografi



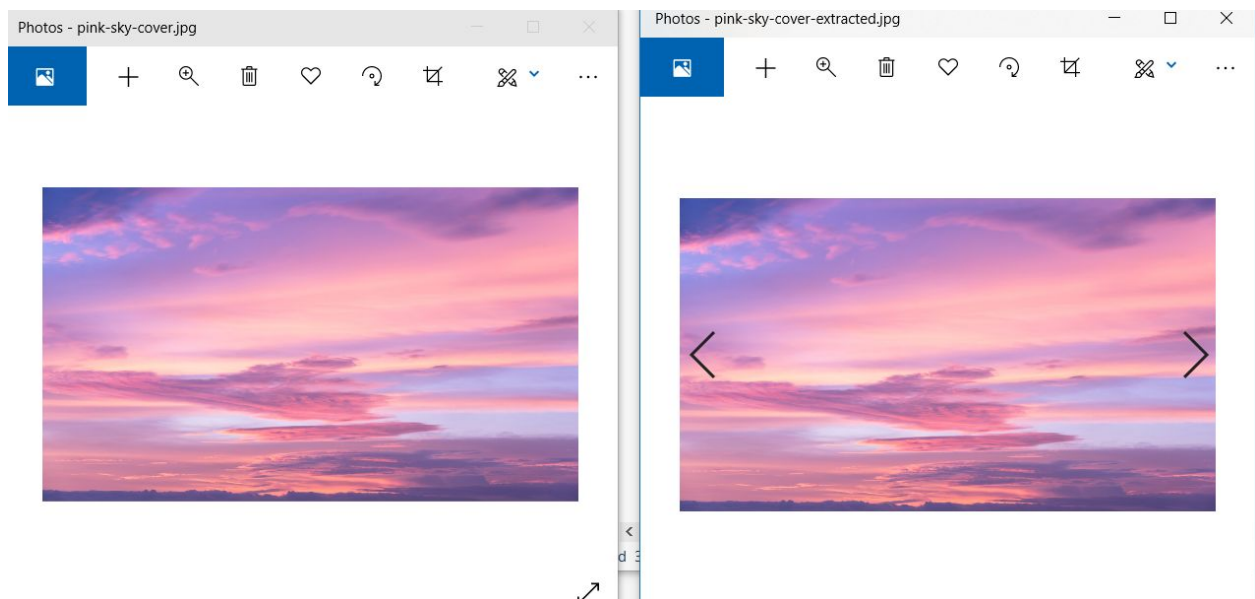
Injeksi dan enkripsi gambar “pink-sky.jpg” ke dalam file “song.wav, Stego-file bernama StegoSong.wav (nama dapat ditentukan sendiri) telah di simpan



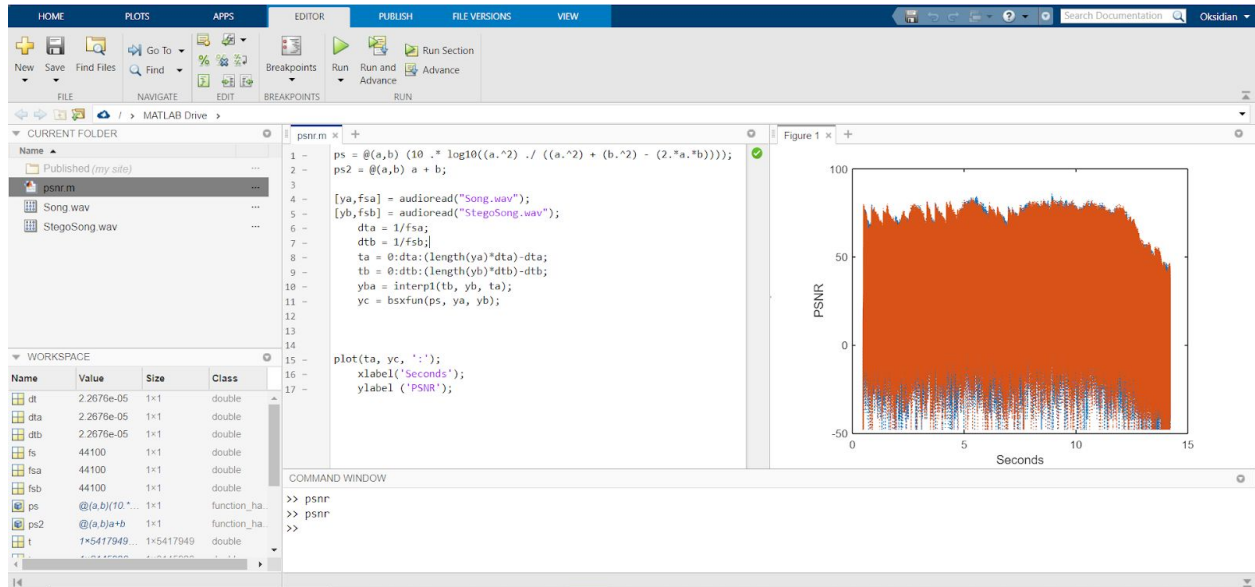
Ekstraksi dan dekripsi pesan di dalam StegoSong.wav



Hasil ekstraksi dan pemilihan nama file hasil ekstraksi (disini dipilih nama file ekstraksi : pink-sky-cover-extracted.jpg)



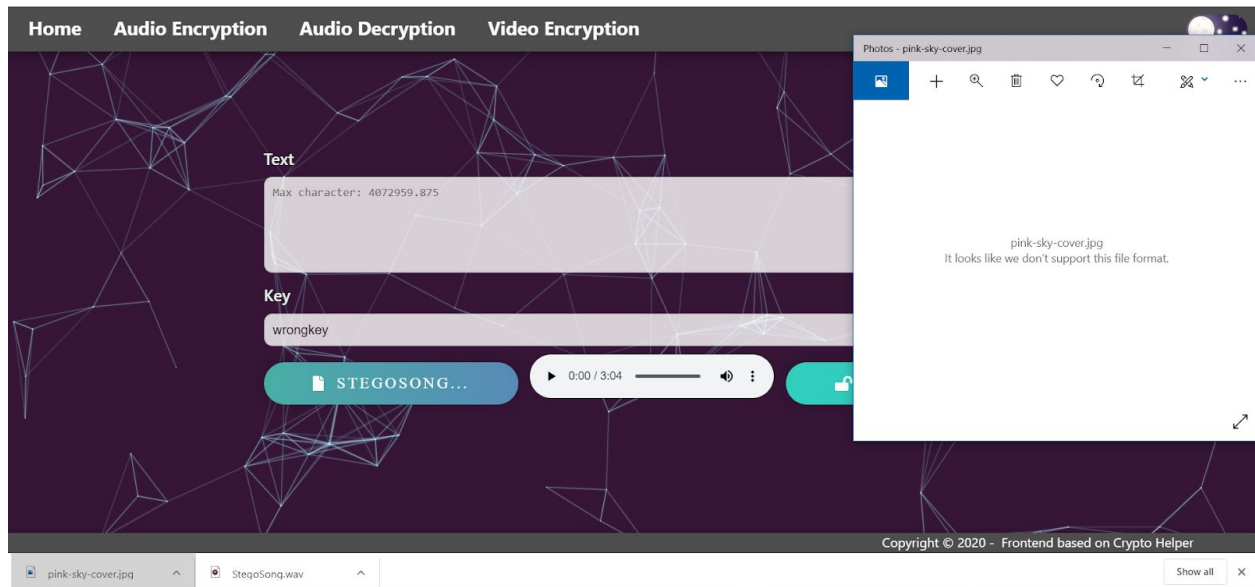
Dapat dilihat pesan sebelum dan sesudah ekstraksi bersifat sama (lossless)



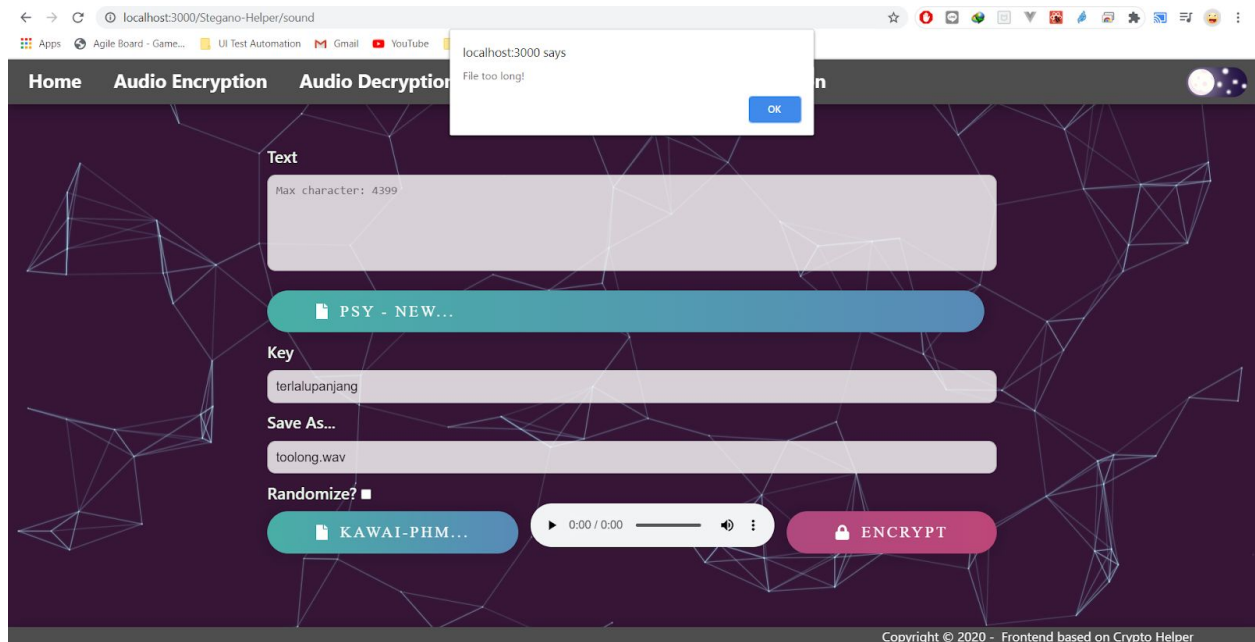
Hasil perhitungan PSNR dari kedua file, dapat dilihat bahwa PSNR tidak jatuh lebih dari 40



## ii. Kasus khusus



Hasil ekstraksi dan dekripsi jika menggunakan kunci yang salah



Error yang muncul jika muatan pesan melebihi kapasitas

#### **4. Kesimpulan dari hasil implementasi**

Dapat disimpulkan bahwa proses enkripsi, dekripsi, dan steganografi (baik injeksi maupun ekstraksi) telah sukses dengan tidak rusaknya gambar (PSNR diatas ambang) dan tidak rusaknya pesan (pesan yang terkandung masih sama).