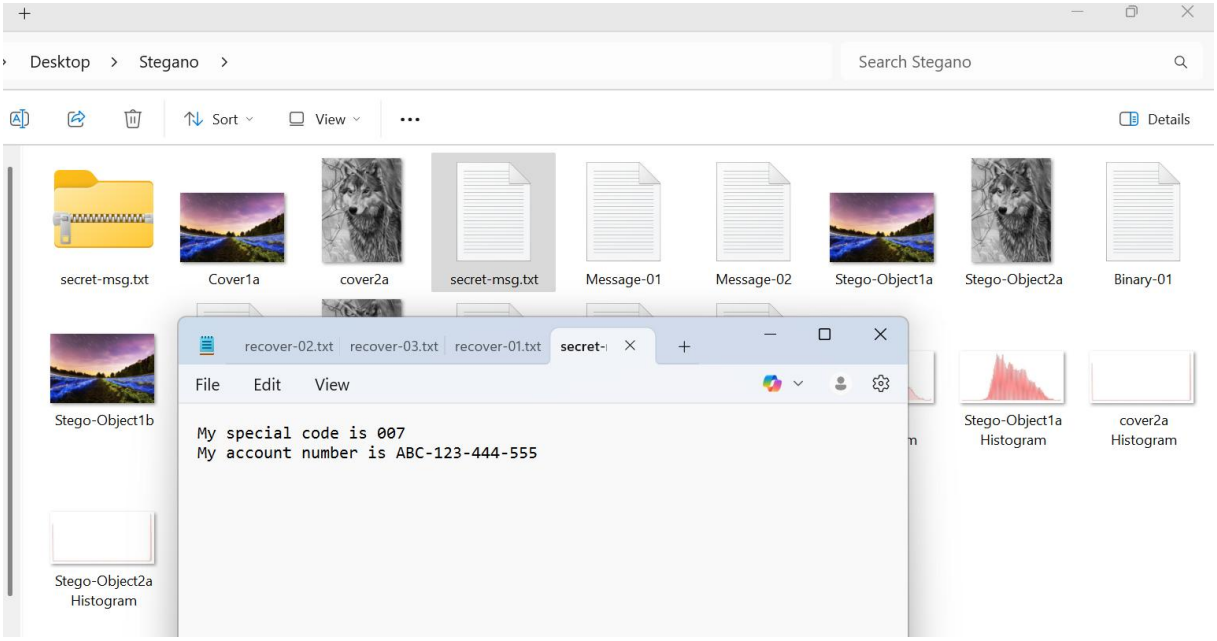
Cryptographic protocol L1
Project
Aghamir Ahmadov
ID 49729

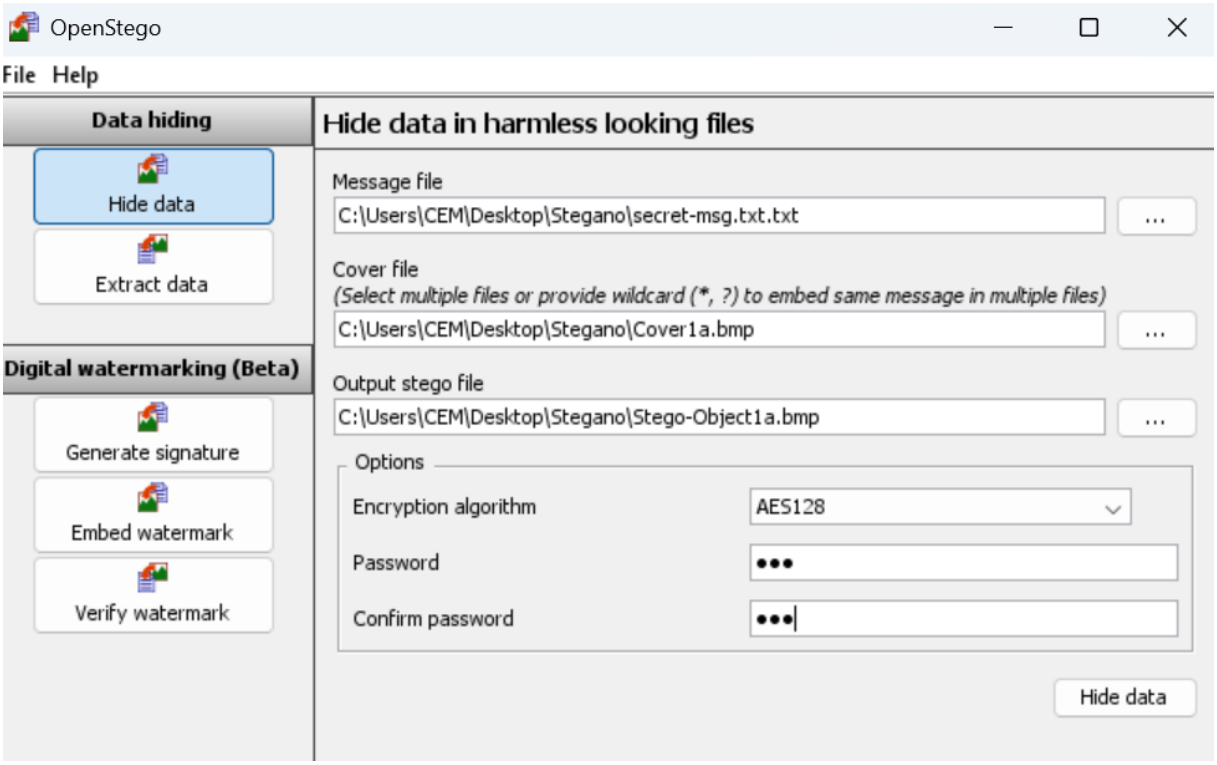
Step#	Submission	File Name	Points	Report
#1	02 Information files (01 text + 01 image) 02 Cover-files 02 Stego-files (files containing hidden message)	secret-msg.txt, Cover1a Cover1a Cover2a Stego-Object1a Stego-Object2a	02	
#2	01 Information files (01 text) 01 Cover-files (original files hiding information) 02 Binary-Representation files (for hidden messages) 03 recovered messages	Message-01 Message-02 Cover1a Cover2a Binary-01 Binary-02 recover-01 recover-02 recover-03	01 02 03	
#3	03+03 difference files 03 Comparisons	text-diff-01 text-diff-02 text-diff-03 image-diff-01 image-diff-02 image-diff-03 compasirion-01 compasirion-02 compasirion-03	03 03 02	
#4	03+03 histogram files	cover1a Histogram Stego-Object1a Histogram cover2a Histogram Stego-Object2a Histogram cover2b Histogram Stego-Object2b Histogram	06	
#5	03+03 HEX-map files	Original-HEX-map-01 stego-HEX-map-01.hex	06	
#6	03+03 DESC Stats	desc-orig-01 desc-stego01 desc-orig-02 desc-stego02 desc-orig-03 desc-stego03	06	

Step #1

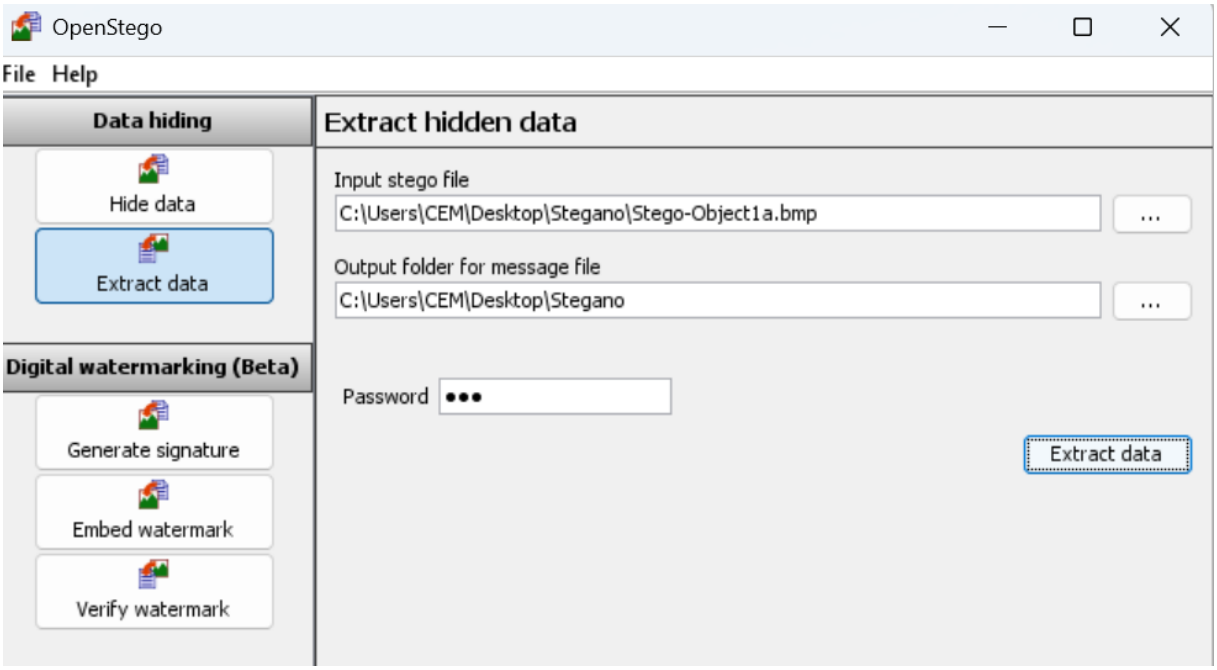
Steganography

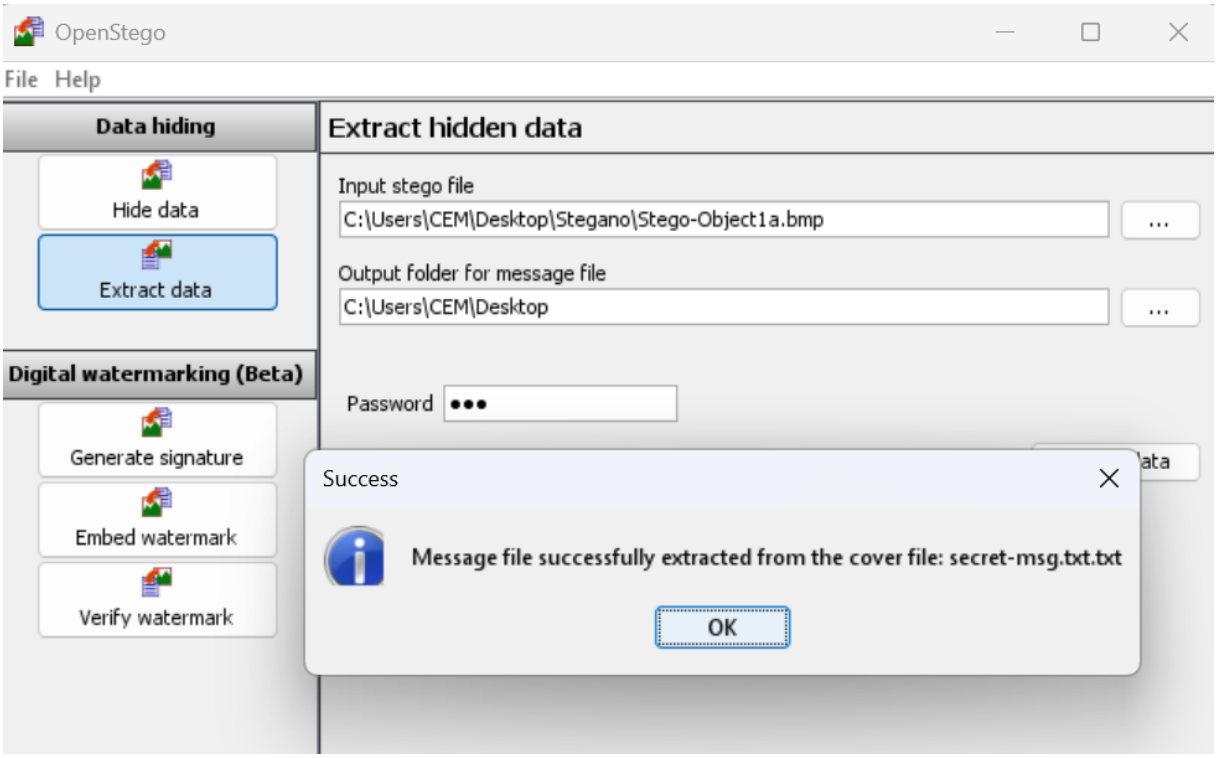
A A text message within image 3 point



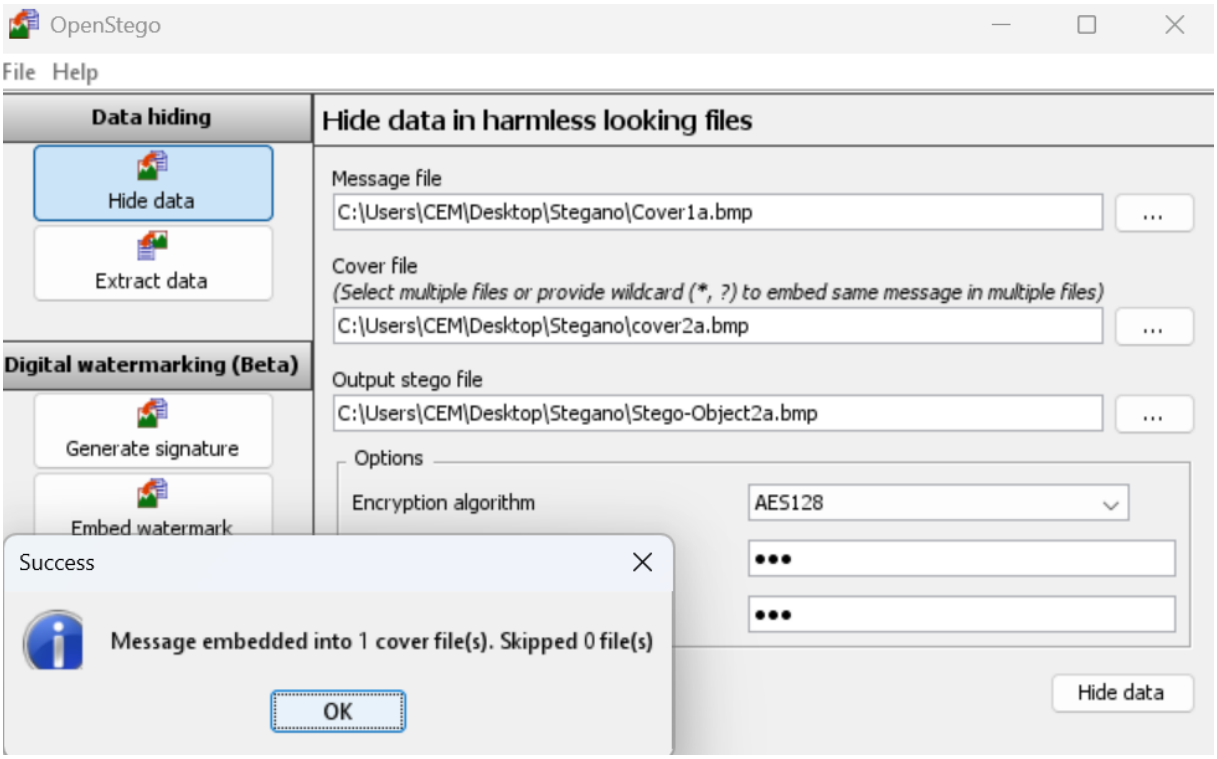


EXTRACTING

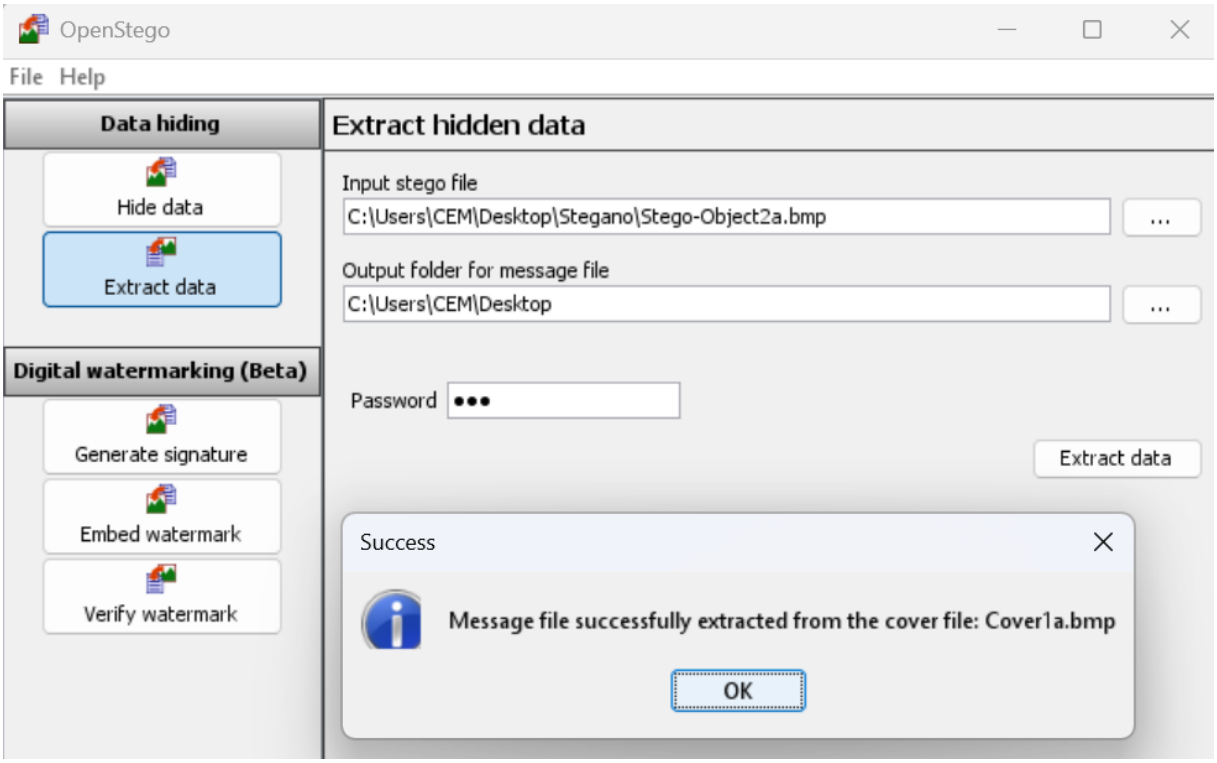

OpenStego
File Help
Data hiding
Hide data
Extract data
Digital watermarking (Beta)
Generate signature
Embed watermark
Verify watermark
Extract hidden data
Input stego file
C:\Users\CEM\Desktop\Stegano\Stego-Object1a.bmp
Output folder for message file
C:\Users\CEM\Desktop\Stegano
Password
Extract data


OpenStego
File Help
Data hiding
Hide data
Extract data
Digital watermarking (Beta)
Generate signature
Embed watermark
Verify watermark
Extract hidden data
Input stego file
C:\Users\CEM\Desktop\Stegano\Stego-Object1a.bmp
Output folder for message file
C:\Users\CEM\Desktop
Password
Success
Message file successfully extracted from the cover file: secret-msg.txt.txt
OK

B. An image within another image [3 point]



Extract hidden data.



Report – Step #1
Steganography

Objective

Apply steganography using OpenStego to hide:

- A **text message** within an image
- An **image** within another image

Files Created & Used

A. Information Files (01 Text + 01 Image)

- `secret-msg.txt` — Text message to be hidden
- `Cover1a` — Image used to hide the text

B. Cover Files (Original images before embedding)

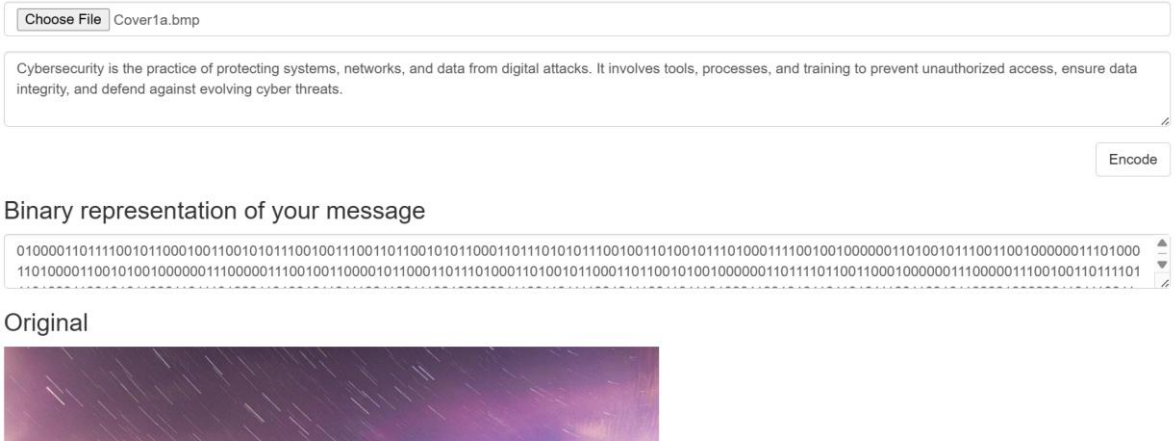
- `Cover1a` — Used for hiding the text message
- `Cover2a` — Used for hiding the image

C. Stego-Files (Files containing hidden information)

- `Stego-Object1a` — Image with embedded text (`secret-msg.txt`)
- `Stego-Object2a` — Image with another image hidden inside

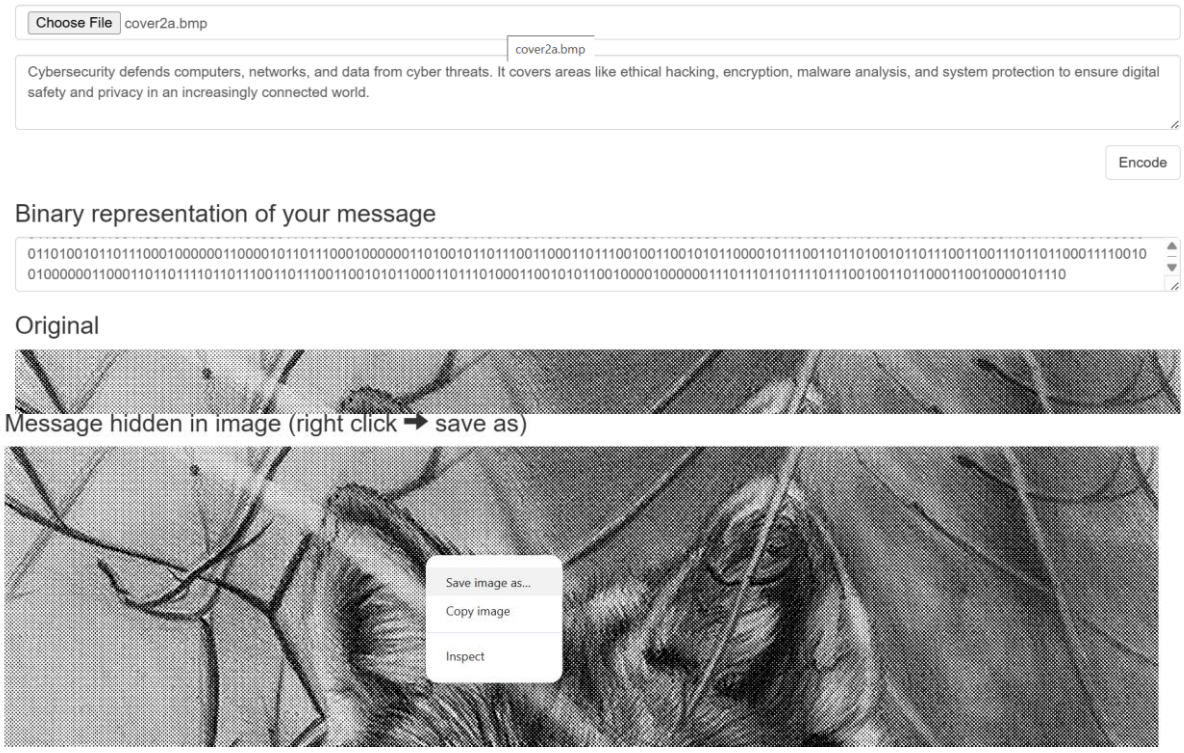
#Step2.

A. Use https://stylesuxx.github.io/steganography/ to encode a message in an image (your picture or other), the message should consist of a large text (at least 200 characters). [2 points]

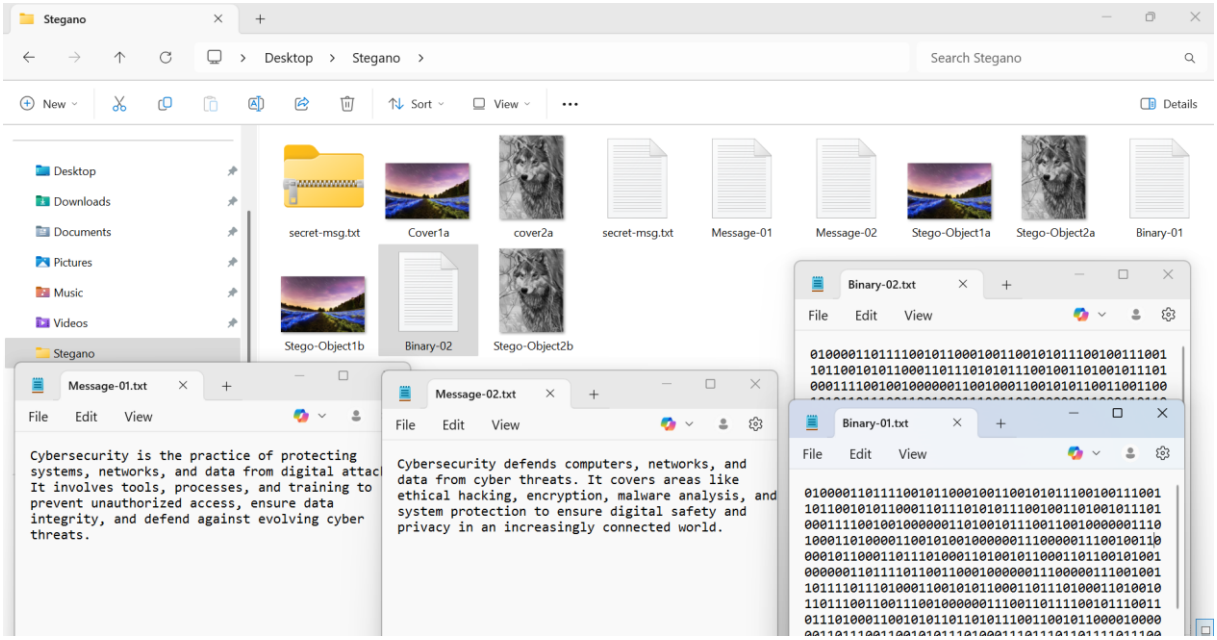


Message hidden in image (right click ➜ save as)

2.a

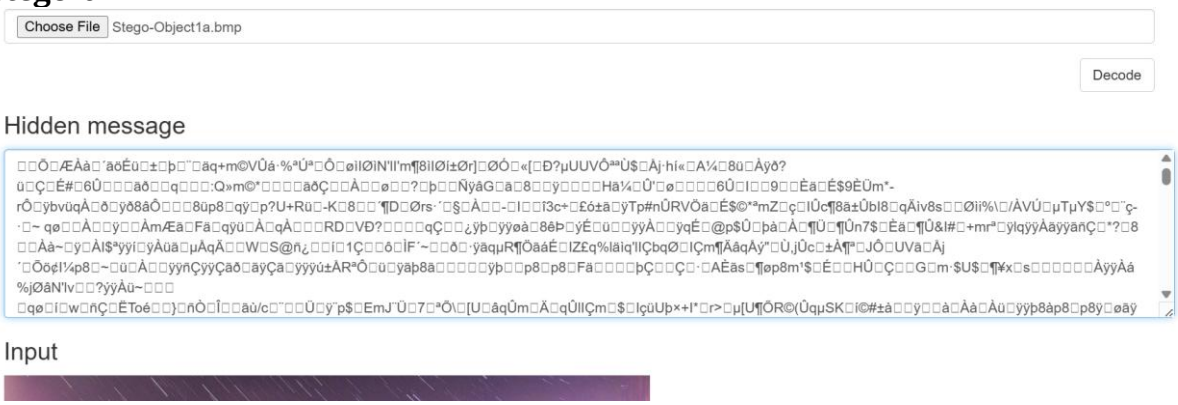


2.B. Analyse the binary representation of the Information-messages (for all the 02 text messages, this step will not be applicable on image-information file (until you convert it into text) , save all binary representations (for each Message-01, Message-02 (text only) as Binary-01 and Binary-02, respectively). [2 points]

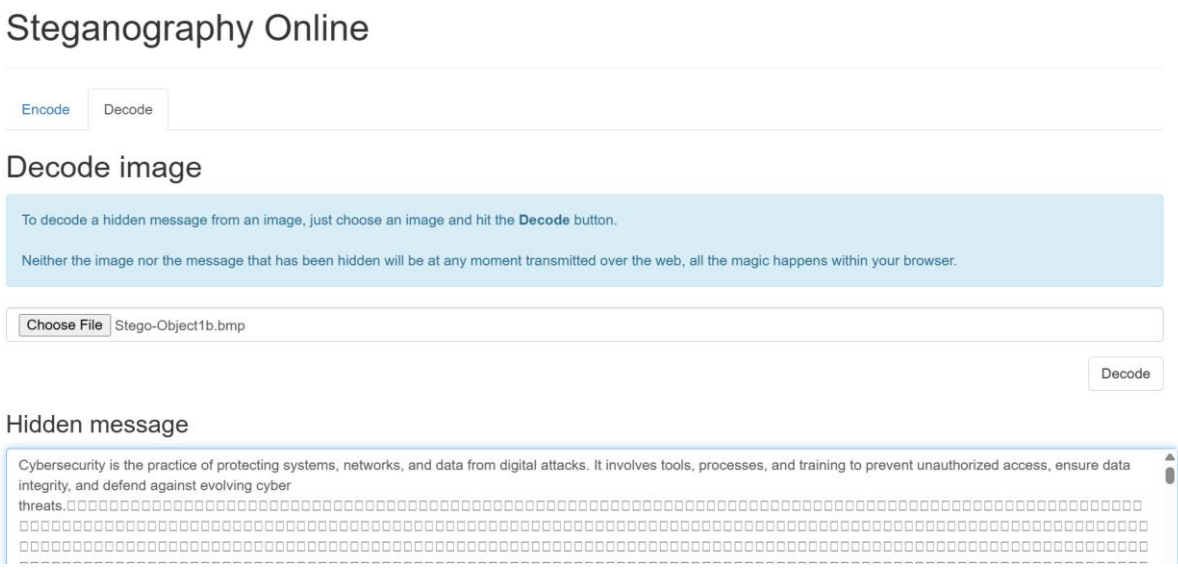


2C. Decode the Messages from the Stego-Object files from the same web site https://stylesuxx.github.io/steganography/ , click the tab Decode, upload your stego-object images (Stego-01, Stego-02 and Stego-03), click Decode button and recover the hidden messages (note that in case of the hidden image you will recover its textual representation-unknown characters), copy the messages and Save these recovered messages for each case i.e. recover-01, recover-02 and recover-03, etc. [2 points]

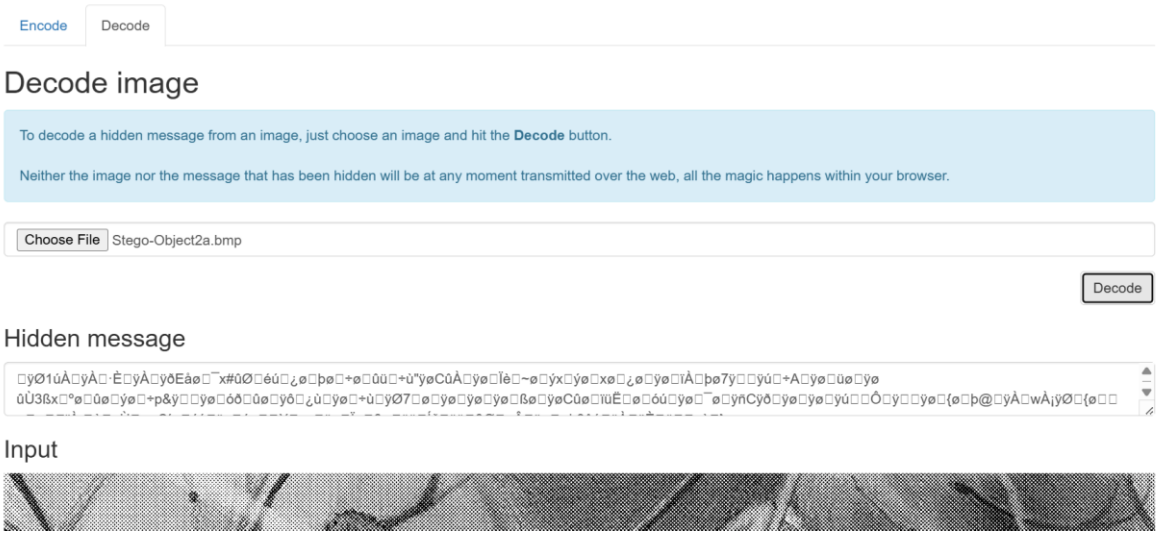
stego-01



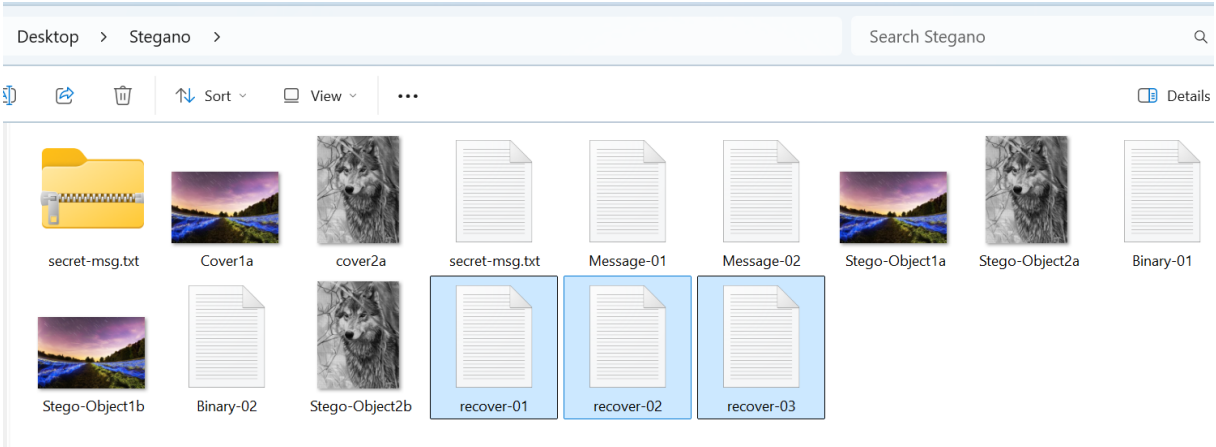
stego-02



stego-03



recover hidden text here



Report Step #2
Steganography

Objective

Use an online tool to hide and recover messages using steganography and analyze the binary representation of hidden text data.

Files Created & Used

A. Information File (01 Text Message)

- `Message-01` — Long text message (≥200 characters) hidden in an image

B. Cover Files (Original images used to hide data)

- `Cover1a` — Cover image used for hiding `Message-01`
- `Cover2a` — Cover image used in previous task (image-in-image steganography)

C. Binary Representation Files (Converted from Messages)

- `Binary-01` — Binary version of `Message-01`
- `Binary-02` — Binary version of `Message-02` (text extracted or used if applicable)

D. Recovered Messages (From Stego-Files)

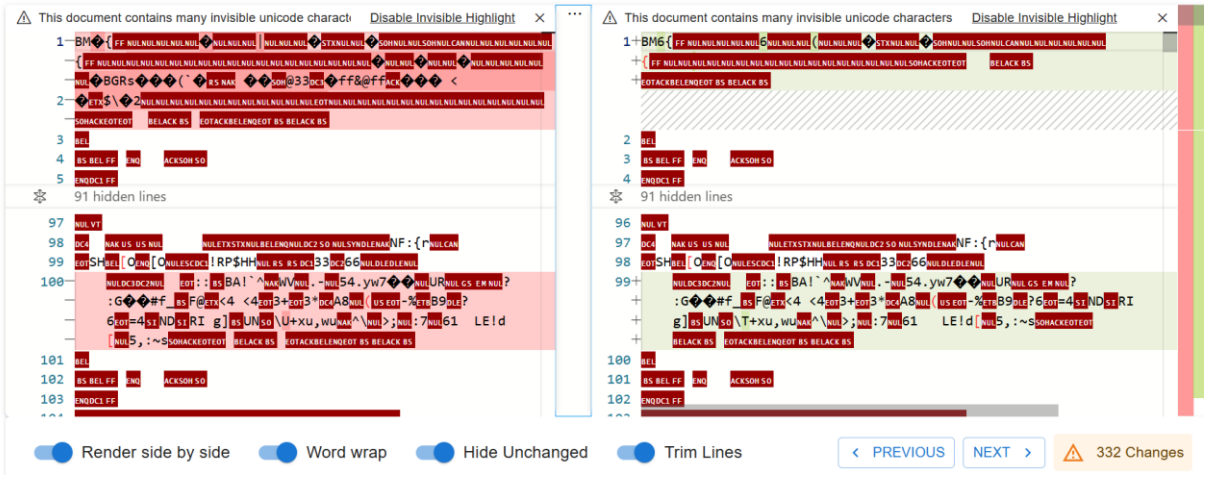
- `recover-01` — Message recovered from `Stego-Object1a`
- `recover-02` — Message/image content recovered from `Stego-Object2a`

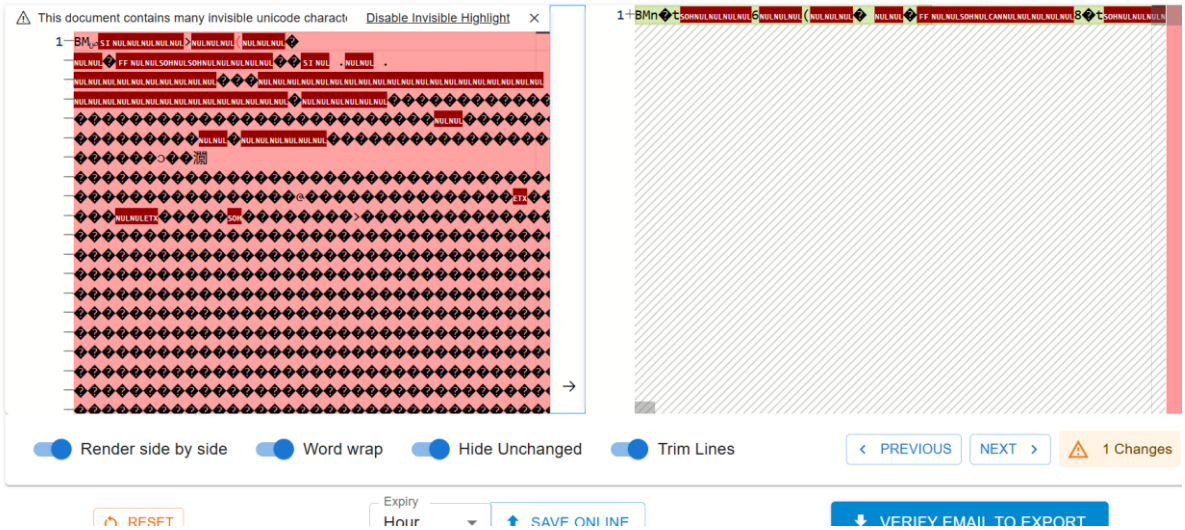
Step#3
Steganalysis (comparison and text-map) [12 points]

A. Use https://www.textcompare.org/ tool to compare both types of the images created in Step# 1 and #2 (Covers and Stegos), use the tools mentioned in step B. [3 points]
B. Compare the differences of (all Covers and Stegos) images using Image Diff and Text Diff tools [3 points]
C. Repeat the Step 3B, exercise by choosing Image Diff for all of your cover and stego files (imagediff-01, 02, 03 etc) [3 points]
D. Compare the files sizes, Check the number of lines in text-map of each image, use in-line difference (highlight comparisons) [3 points]

Online Text Compare Tool

Find difference between 2 text files. Just input or paste original and modified text and click Compare button. Fast, Private & Unlimited.





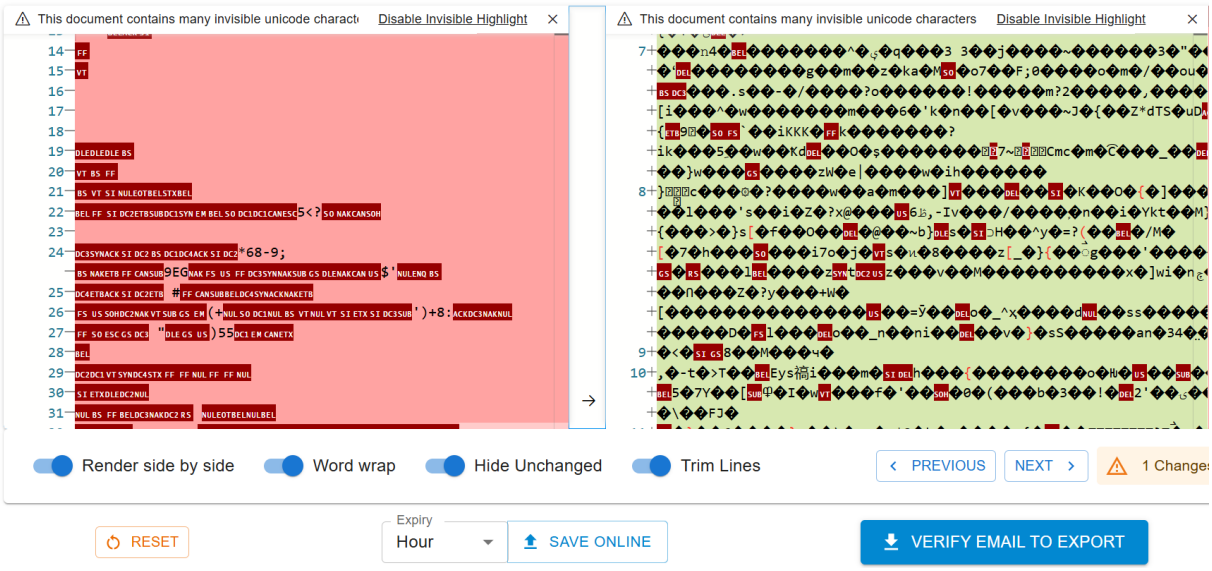


Image Compare Tool Online

Compare and find difference in 2 images or photos easily. Choose images to view changes instantly. Unlimited usage and share online or offline.

CHOOSE ORIGINAL FILE

CHOOSE MODIFIED FILE

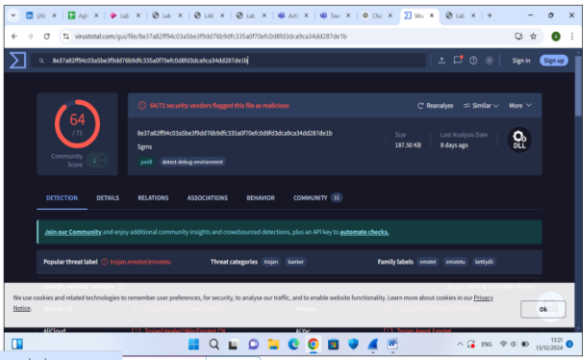
ADD SAMPLE | CLEAR | SWAP

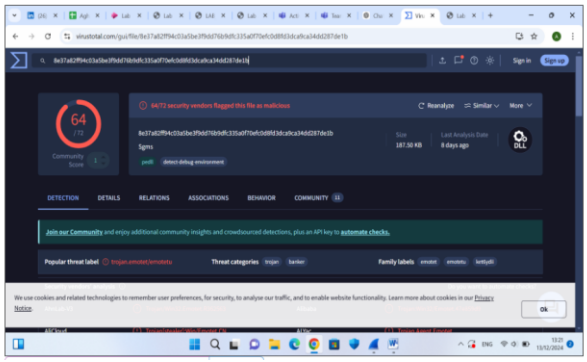
COMPARE

TextCompare Pro TEXT SPREADSHEET DOCUMENT IMAGE CSV PRICING DOWNLOAD

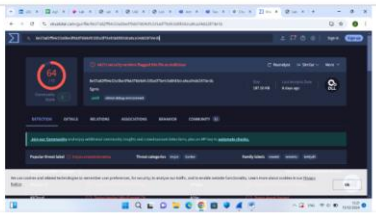
Online Image Comparison Tool - Compare Two Images Side by Side

Compare and find differences between two images online. Features include visual diff highlighting, slider comparison, fade transition, and pixel-perfect difference detection. Support for PNG, JPG, WEBP and many image formats.





DIFF SLIDER FADE DIFFERENCE DETAILS



0.00% Changes

Resize to same size

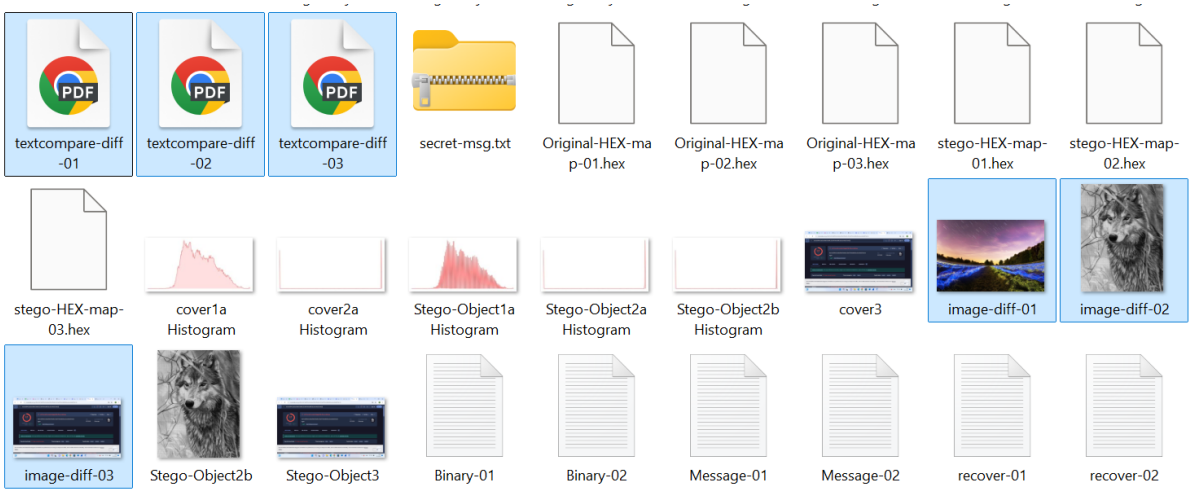
RESET

Expiry Hour

SAVE ONLINE

VERIFY EMAIL TO EXPORT

▶ Give Rating



Report Step 3
Steganalysis

In this step, I performed a detailed steganalysis to compare the original cover images with the corresponding stego images generated in the previous steps.

A. Text-Based Comparison

Using TextCompare.org's **Text Diff** tool, I compared the text representations (hex/image maps) of each pair of cover and stego images. The results were saved as:

- `text-diff-01.txt`
- `text-diff-02.txt`
- `text-diff-03.txt`

Each file highlights the specific differences in characters and lines, confirming subtle changes introduced by the hidden data.

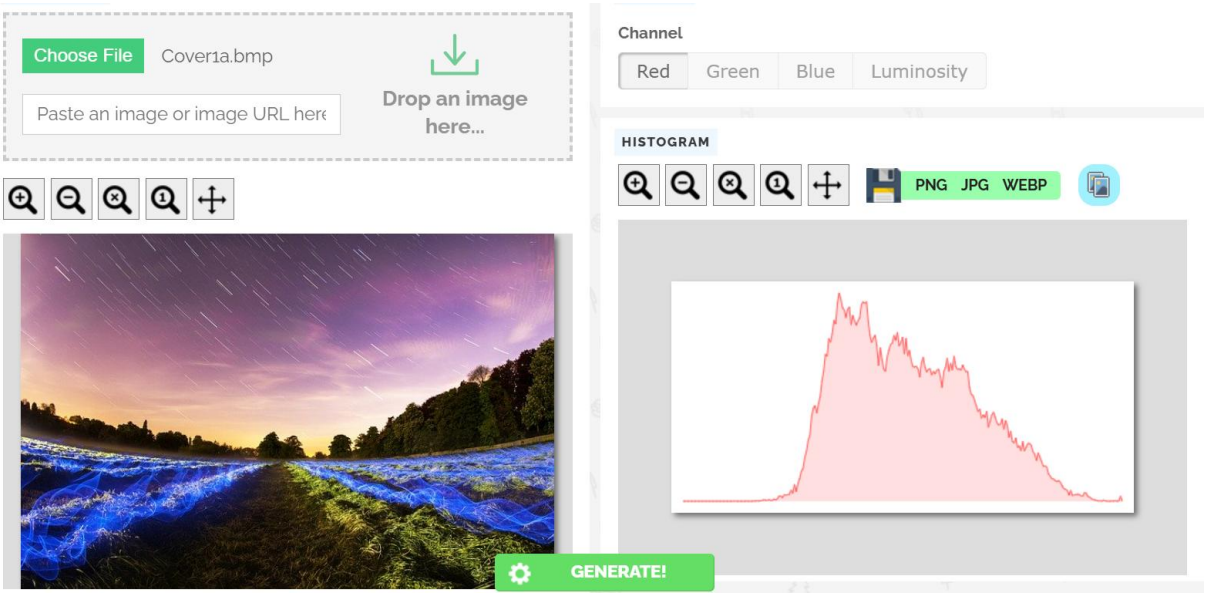
B. Image-Based Comparison

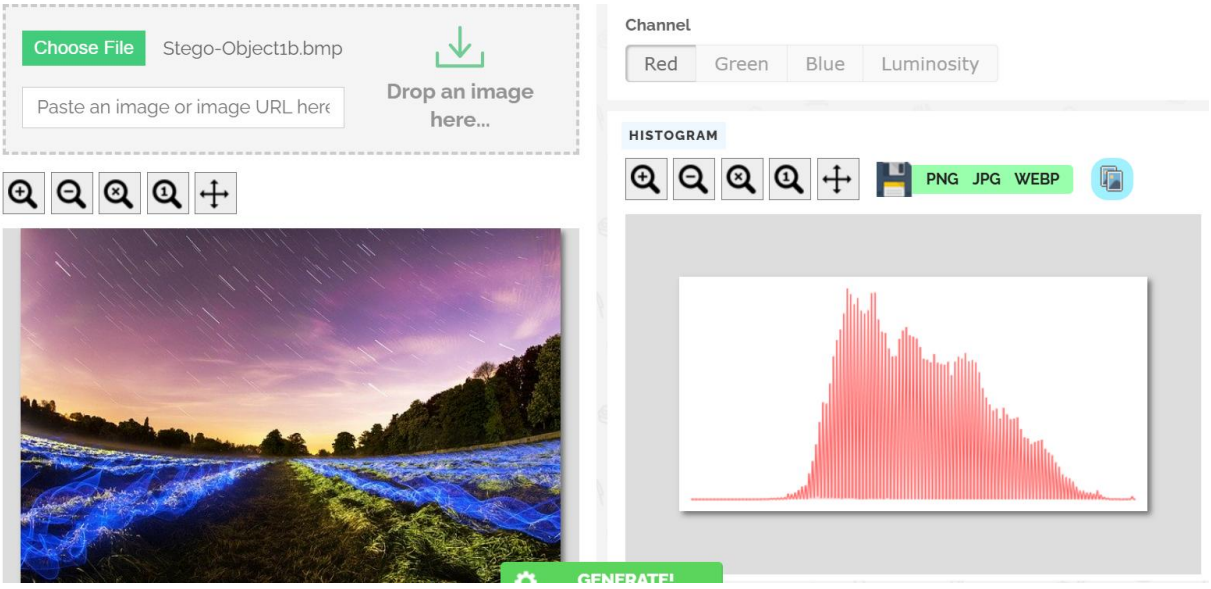
I used the **Image Diff** tool to visually compare all cover and stego image pairs. The differences were exported as:

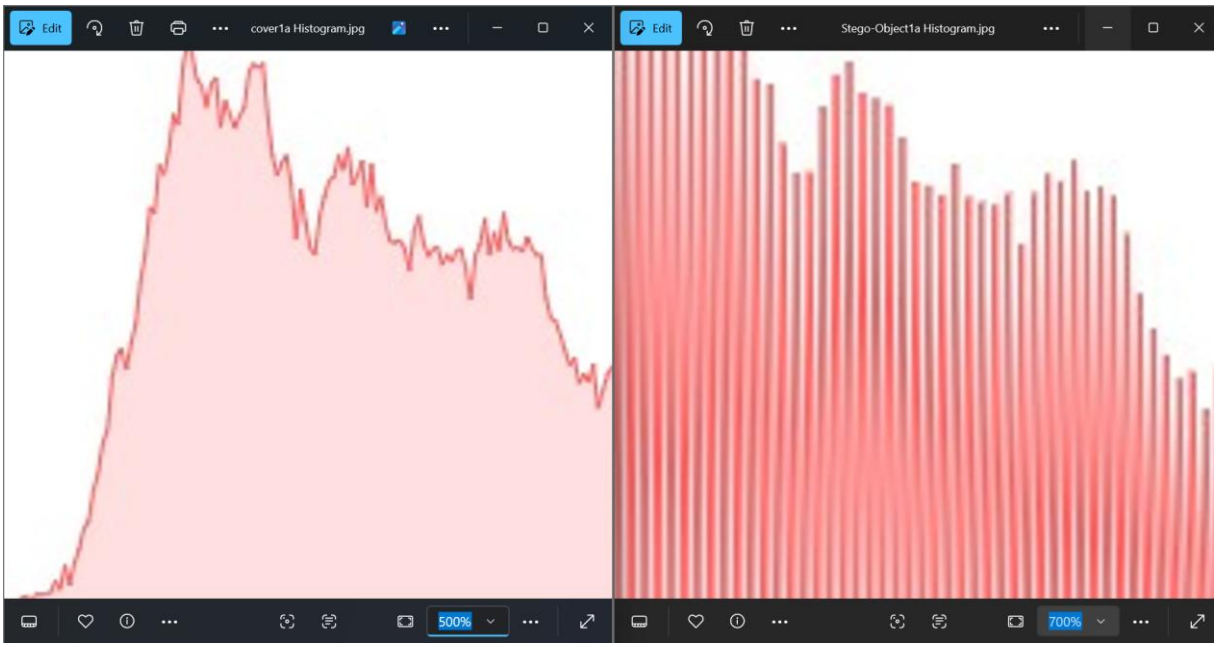
- `image-diff-01.png`
- `image-diff-02.png`
- `image-diff-03.png`

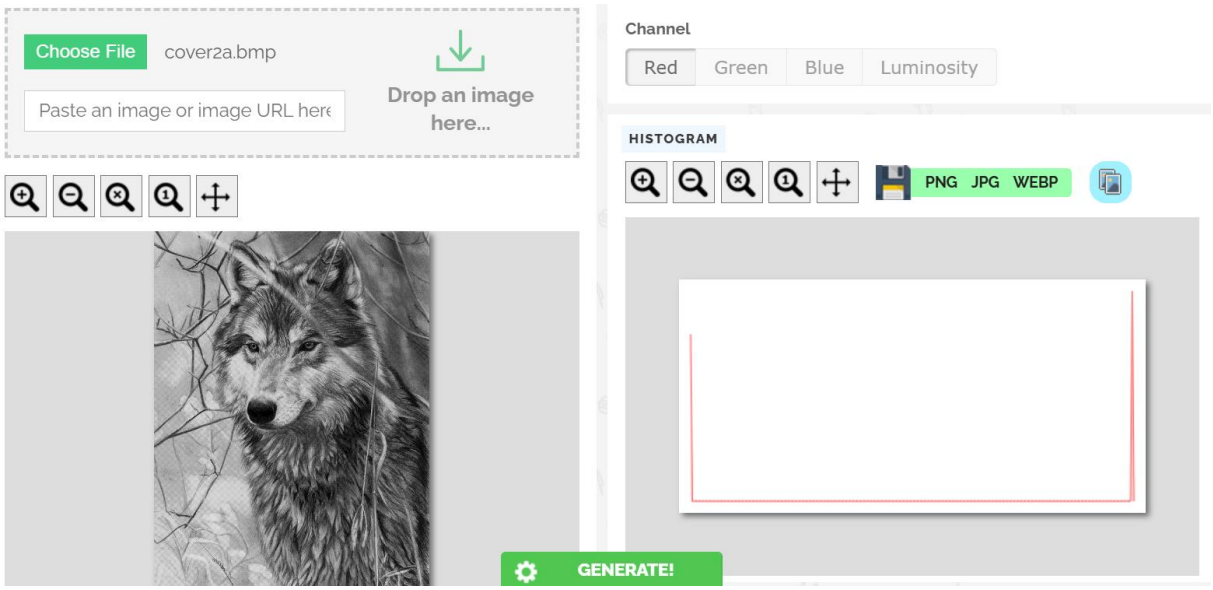
Step#4
Steganalysis (Histogram comparison) [8 points]

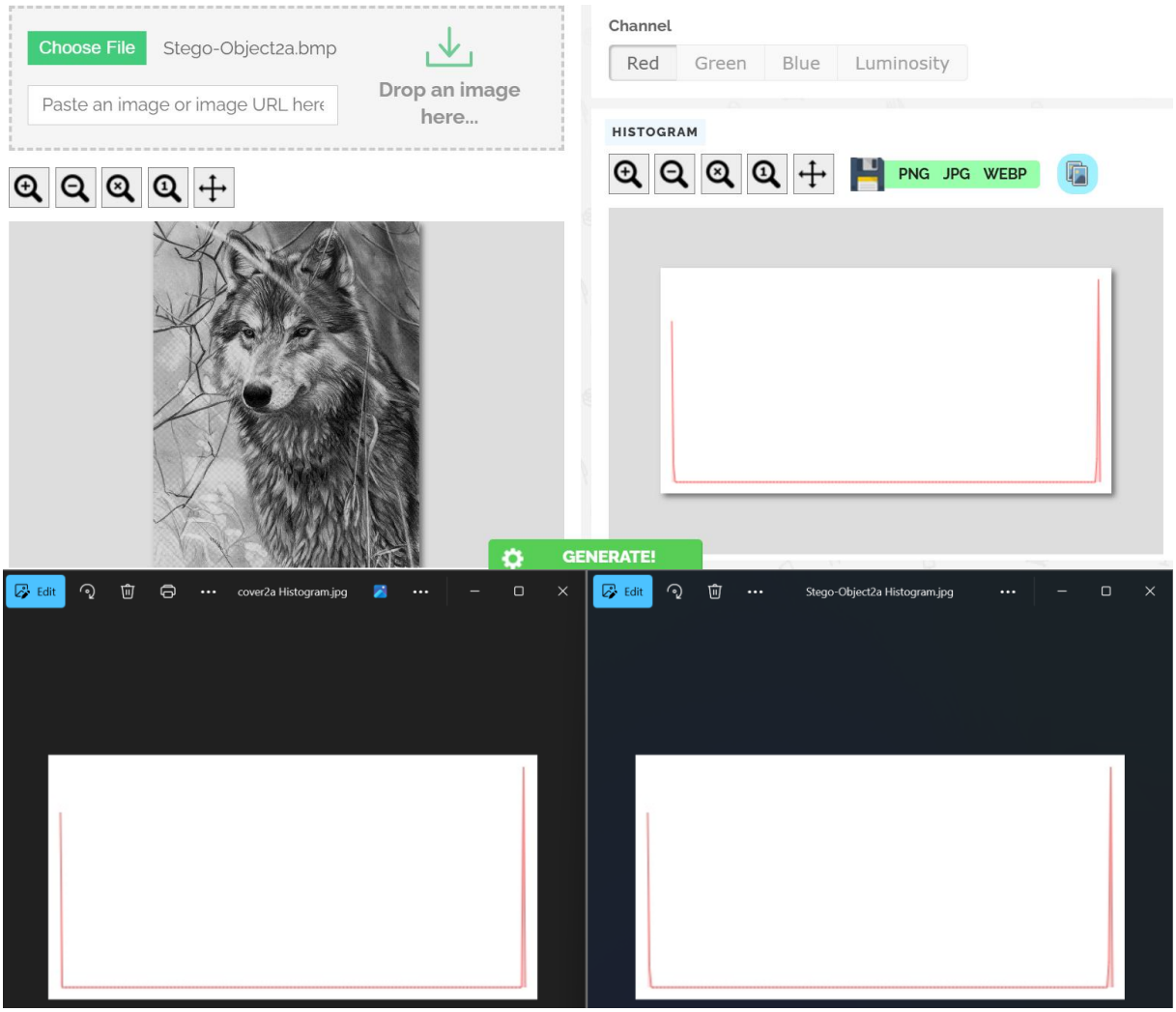
A. Use https://pinetools.com/image-histogram to create histograms of your(Cover and Stego images created in step#1 and #2) ; upload the files, click generate and then save the histogram files of both images and paste them in document or image editor and zoom them

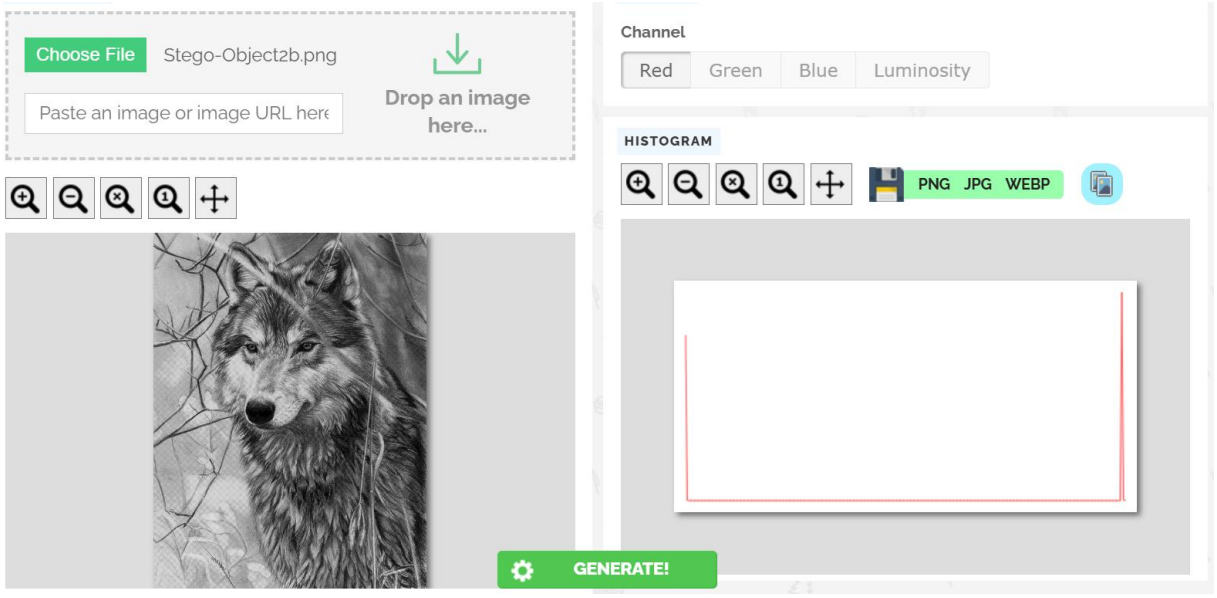

Choose File
Cover1a.bmp
Paste an image or image URL here
Drop an image here...
Channel
Red
Green
Blue
Luminosity
HISTOGRAM
PNG JPG WEBP
GENERATE!

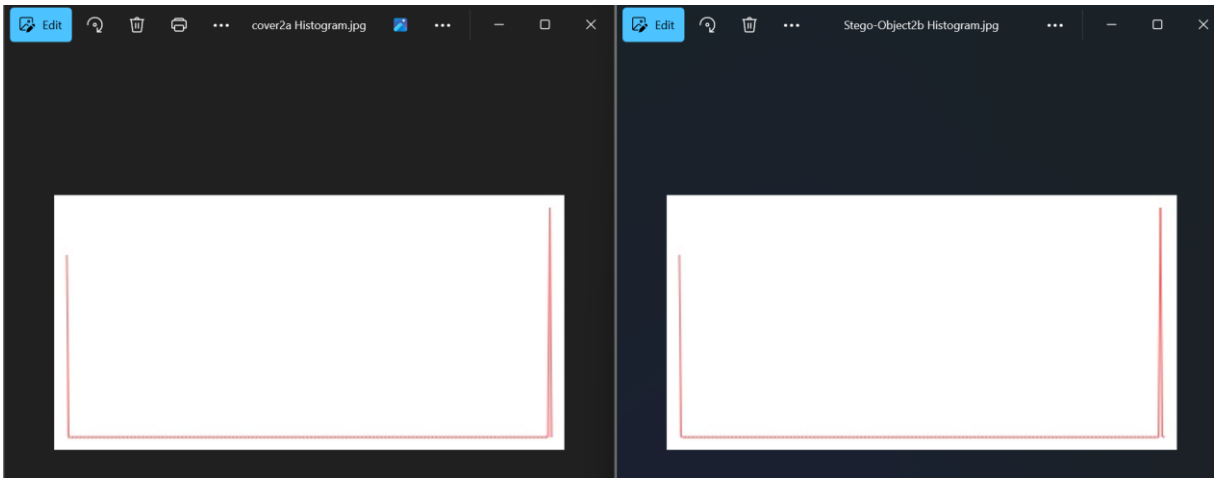

Choose File
Stego-Object1b.bmp
Paste an image or image URL here
Drop an image here...
Channel
Red
Green
Blue
Luminosity
HISTOGRAM
PNG JPG WEBP
GENERATE!


Edit
cover1a Histogram.jpg
500%
Edit
Stego-Object1a Histogram.jpg
700%


Choose File
cover2a.bmp
Paste an image or image URL here
Drop an image here...
Channel
Red
Green
Blue
Luminosity
HISTOGRAM
PNG JPG WEBP
GENERATE!


Choose File
Stego-Object2a.bmp
Paste an image or image URL here
Drop an image here...
Channel
Red
Green
Blue
Luminosity
HISTOGRAM
PNG JPG WEBP
GENERATE!
Edit
cover2a Histogram.jpg
Edit
Stego-Object2a Histogram.jpg


Choose File
Stego-Object2b.png
Paste an image or image URL here
Drop an image here...
Channel
Red
Green
Blue
Luminosity
HISTOGRAM
PNG JPG WEBP
GENERATE!



Report Step 4
Steganalysis (Histogram Comparison)

In this step, I analyzed the histograms of cover and stego images to detect hidden data patterns.

A. Histogram Generation

Using PineTools – Image Histogram, I generated histograms for all cover and stego images:

- `cover1a` & `Stego-Object1a`
- `cover2a` & `Stego-Object2a`
- `cover2b` & `Stego-Object2b`

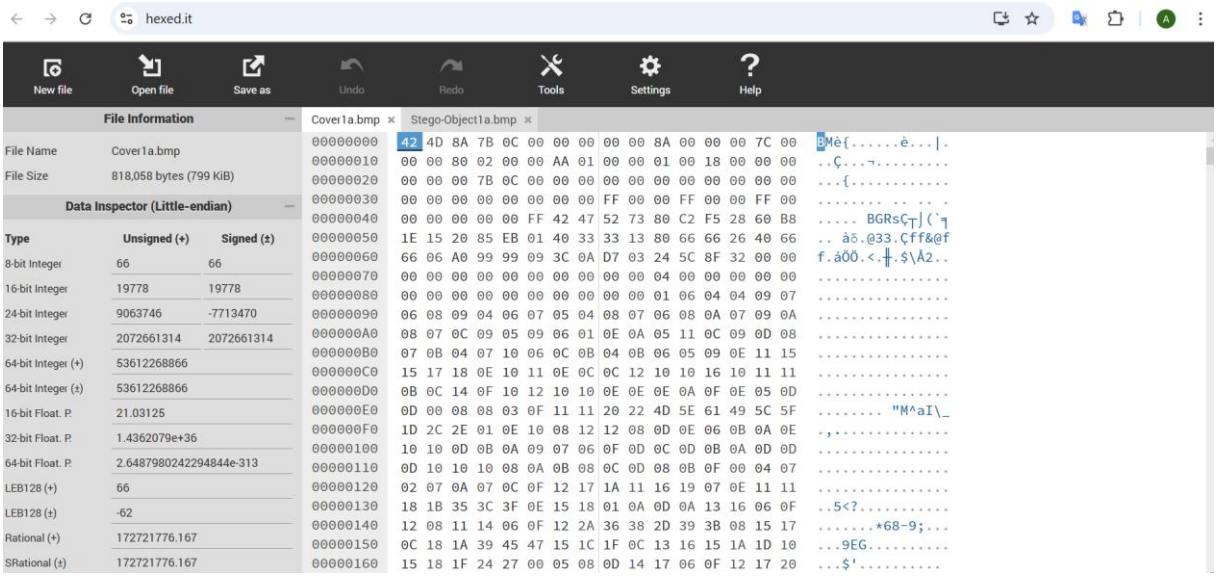
Each histogram was saved and exported for visual comparison.

B. Zoom & Difference Analysis

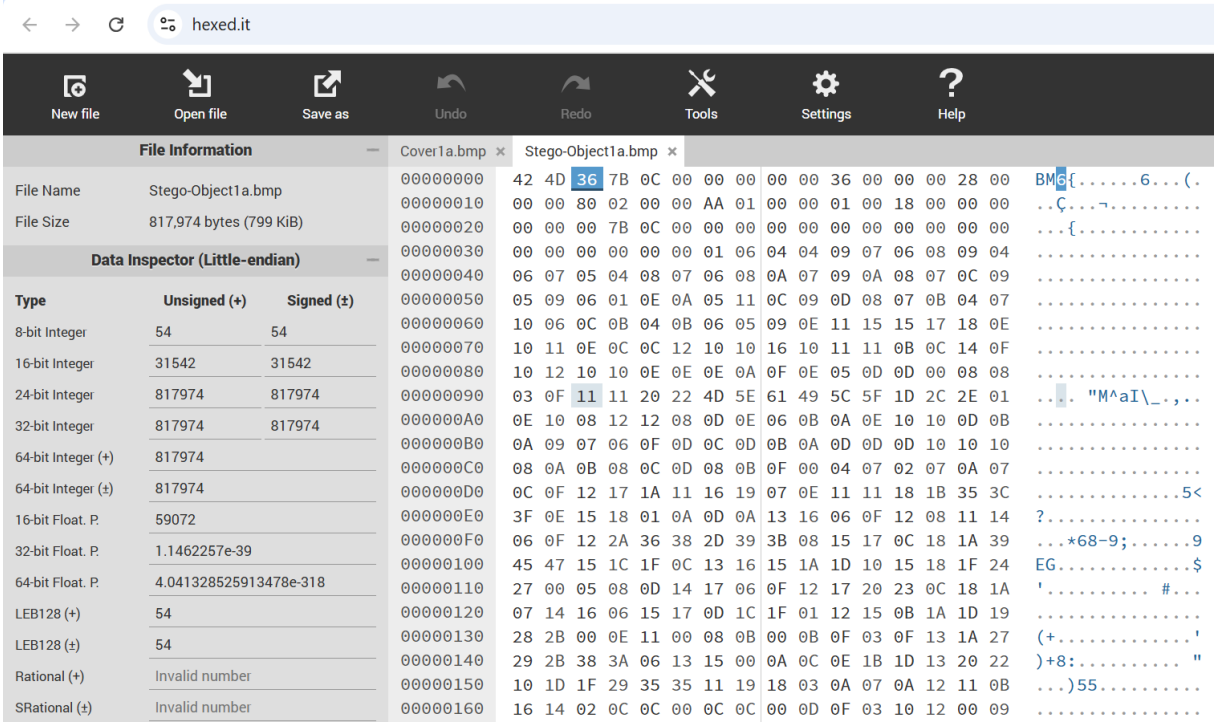
I zoomed each histogram to **500–700%** using an image editor and manually identified areas where the stego histograms differed from the covers. These difference points were marked visually, showing pixel distribution changes caused by data embedding.

Step#5 Steganalysis (HEX map comparison) [6 points]
A. Use https://hexed.it to analyse the HEX values of both types the images (Covers and Stegos created in step#1 and #2), [1 point]

cover



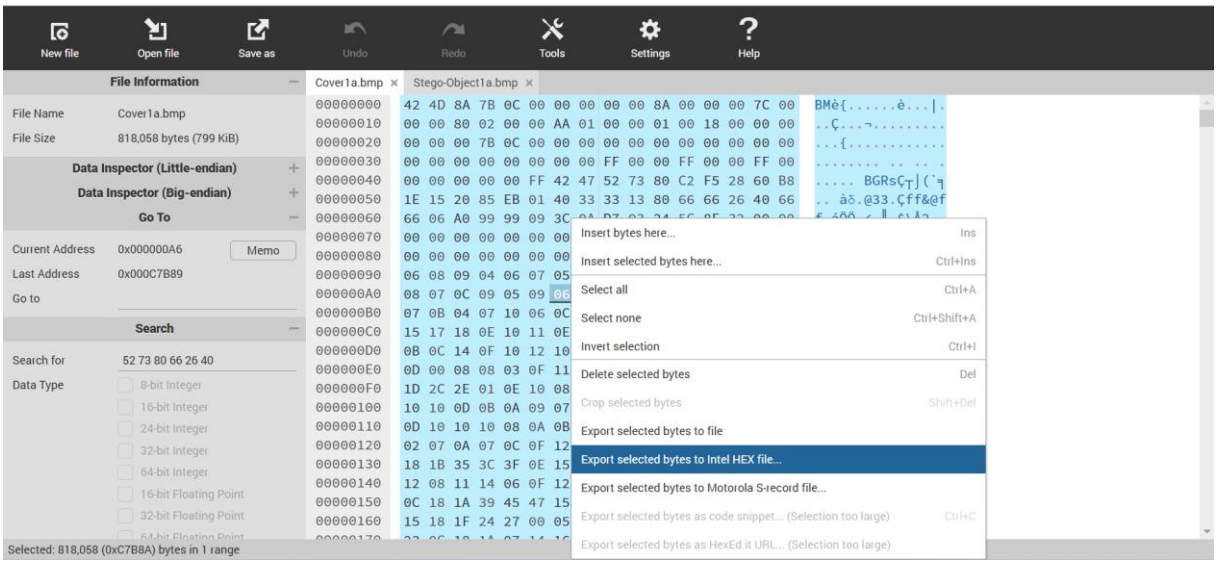
stegro



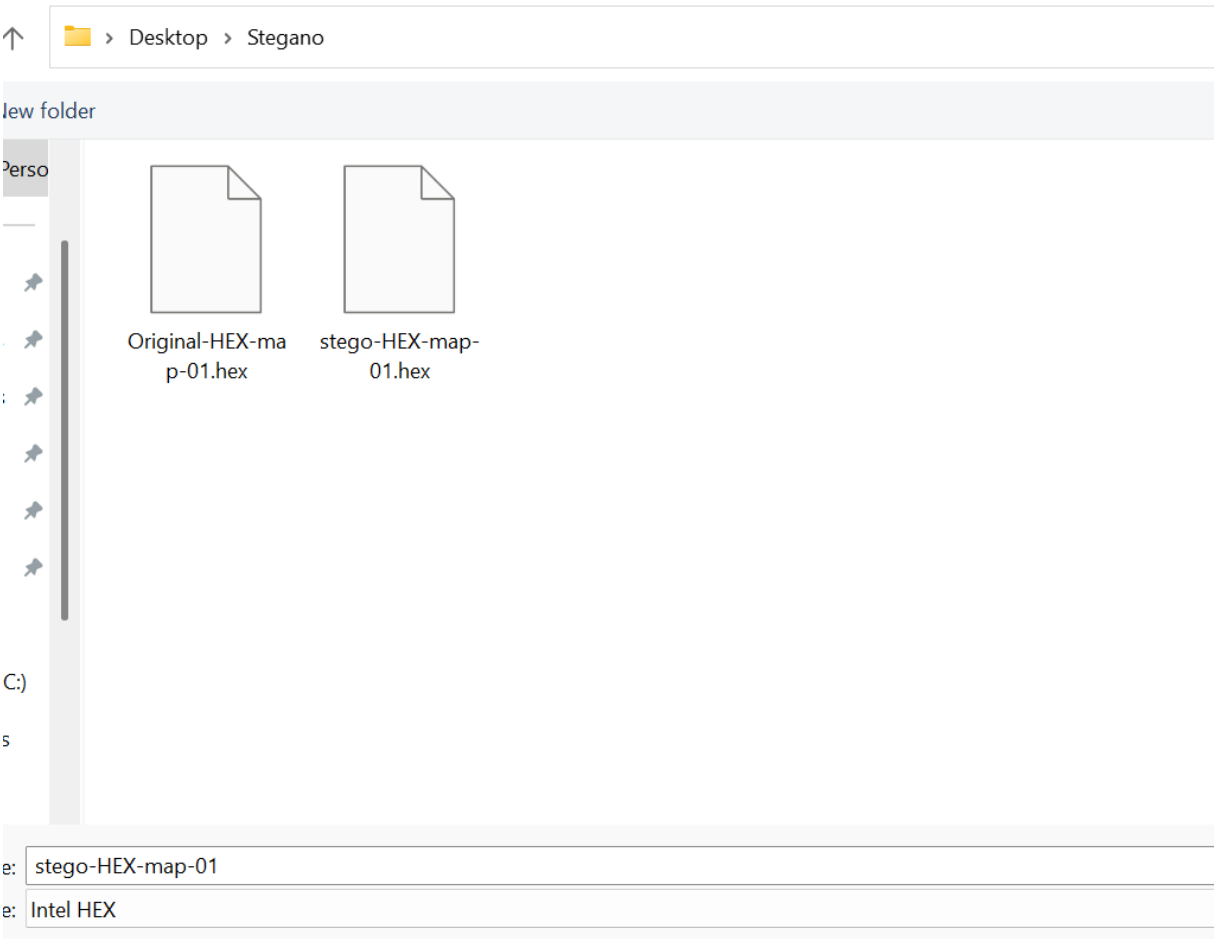
B. Load both files, the HEX map and its text will be shown, scroll up-down in the HEX map to check the difference in both files HEX map and mark for the differences. [1 point]
Answer:
Several bytes throughout the hex map differ between the two files—most notably in the data section, not the header—confirming the presence of a hidden message.

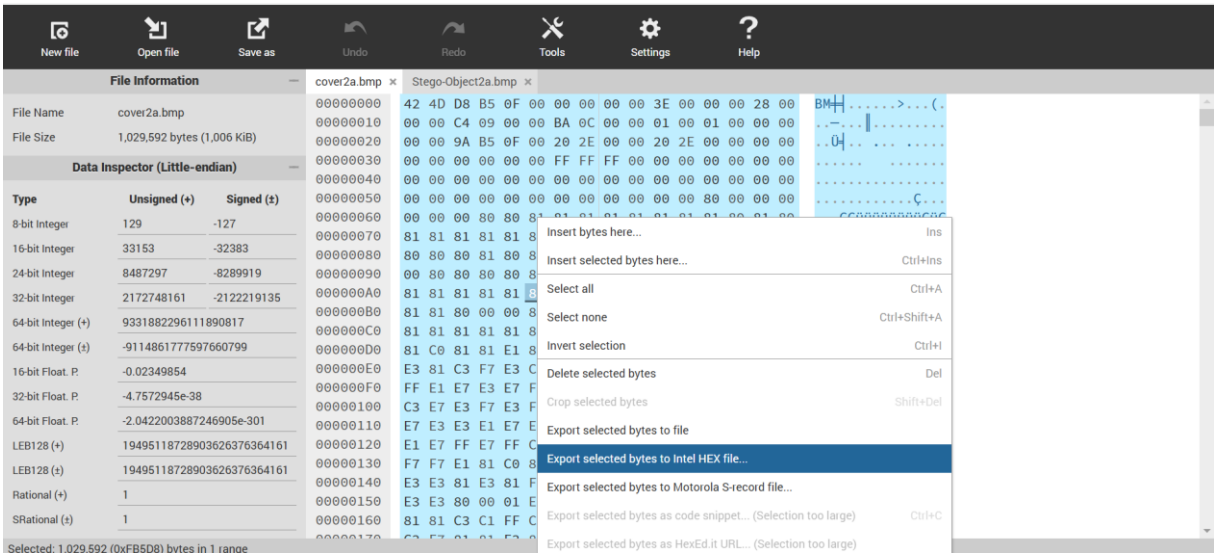
These changes suggest the payload (hidden message) has been embedded by altering pixel bytes slightly.
Orginal:42 4D 8A 7B 0C 00
Stego: 42 4D 36 7B 0C 00

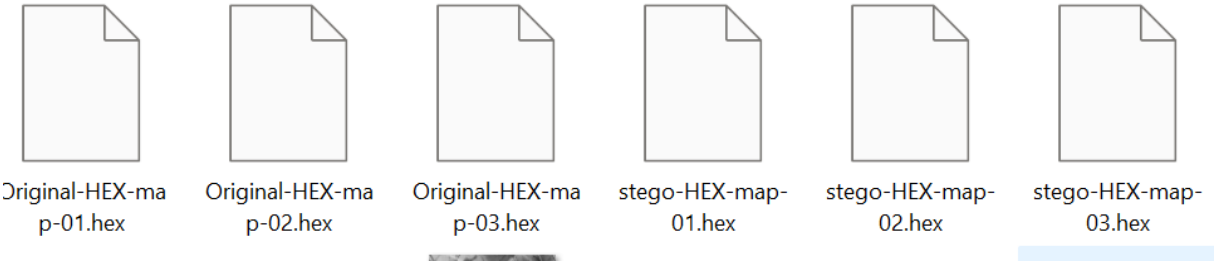
C. Right click the HEX-Map, choose select all and then again right click and choose “Export selected bytes to intel HEX file”, Click OK and then [1 point]



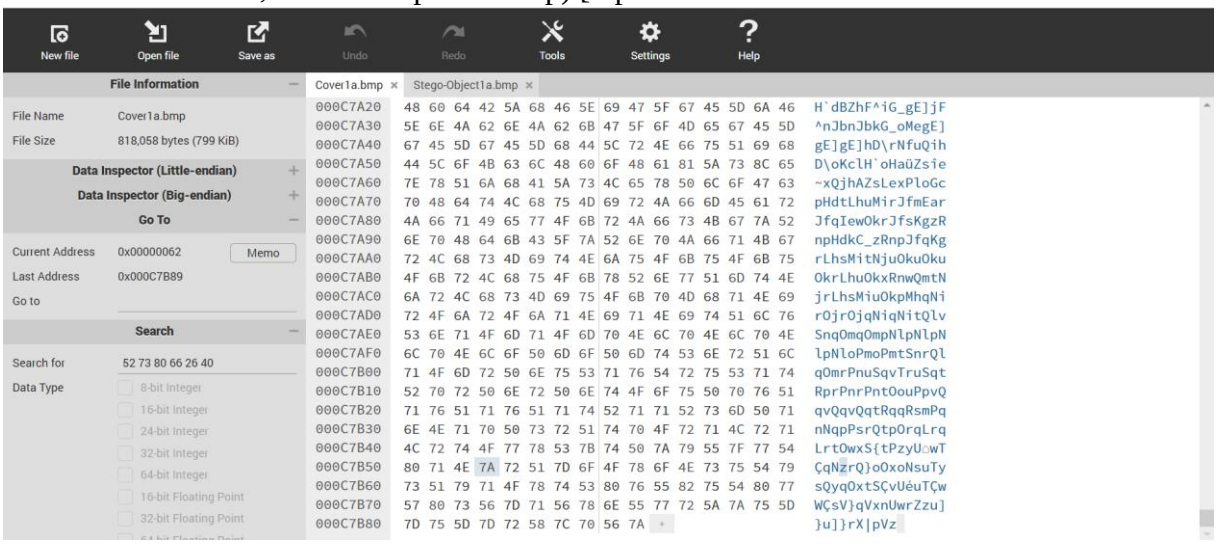
D. Save the HEX-file with Original-HEX-map-01 and stego-HEX-map-01, …02….and …..03…respectively.[1 point]

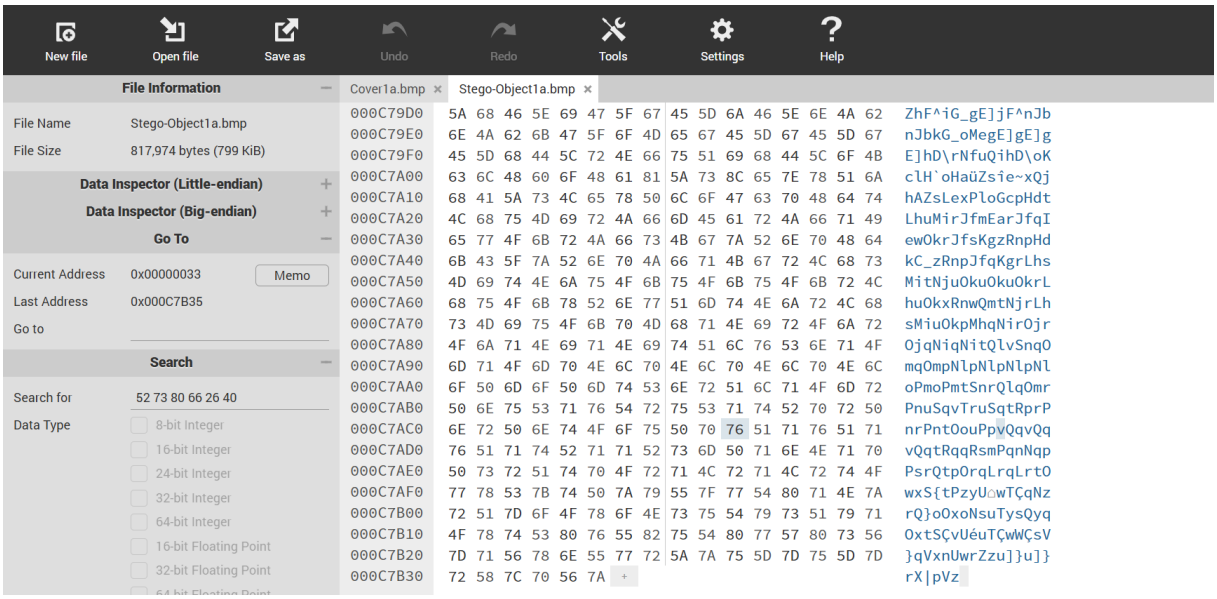






E. Scroll down to the end of the maps and compare first and last few lines and indicate the difference. (Option: You can also use MS-DOS command “comp” to compare both files or # of lines in both files, check comp/? for help) [2points





subtle byte differences start appearing after the header section. These changes are typically found in the least significant bits (LSBs) of pixel data, a common technique used in steganography.
Orginal:7D 75 5D 7D 72 58 7C 70 56 7A
Stego: 72 58 7C 70 56 7A

Report Step 5
Steganalysis (HEX Map Comparison)

In this step, I performed a hex-level analysis of the cover and stego images to identify hidden data modifications.

HEX Analysis

I used HexEd.it to open and view the HEX and ASCII content of the images:
Original Cover vs. Stego files for each image pair

Visual Comparison

By scrolling through the HEX map, I manually observed and marked differences in byte values between the cover and stego files.

Export HEX Files

Each HEX view was:

- Fully selected (Right-click → *Select all*)
- Exported to **Intel HEX format** using:
  Right-click → Export selected bytes to Intel HEX file → OK

File Naming

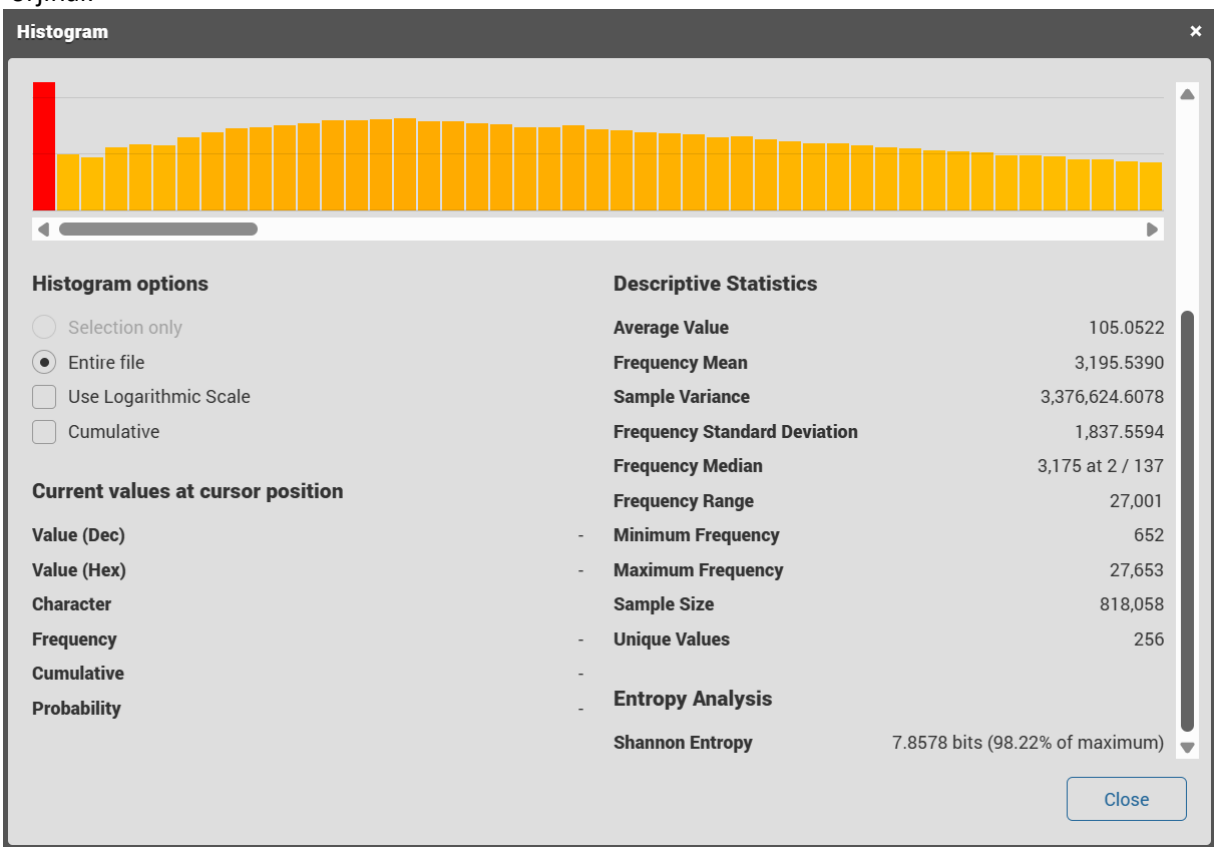
Saved each export with the following names:
`Original-HEX-map-01.hex` & `stego-HEX-map-01.hex`

Step#6 Steganalysis (Descriptive Statistics comparison) [6 points]

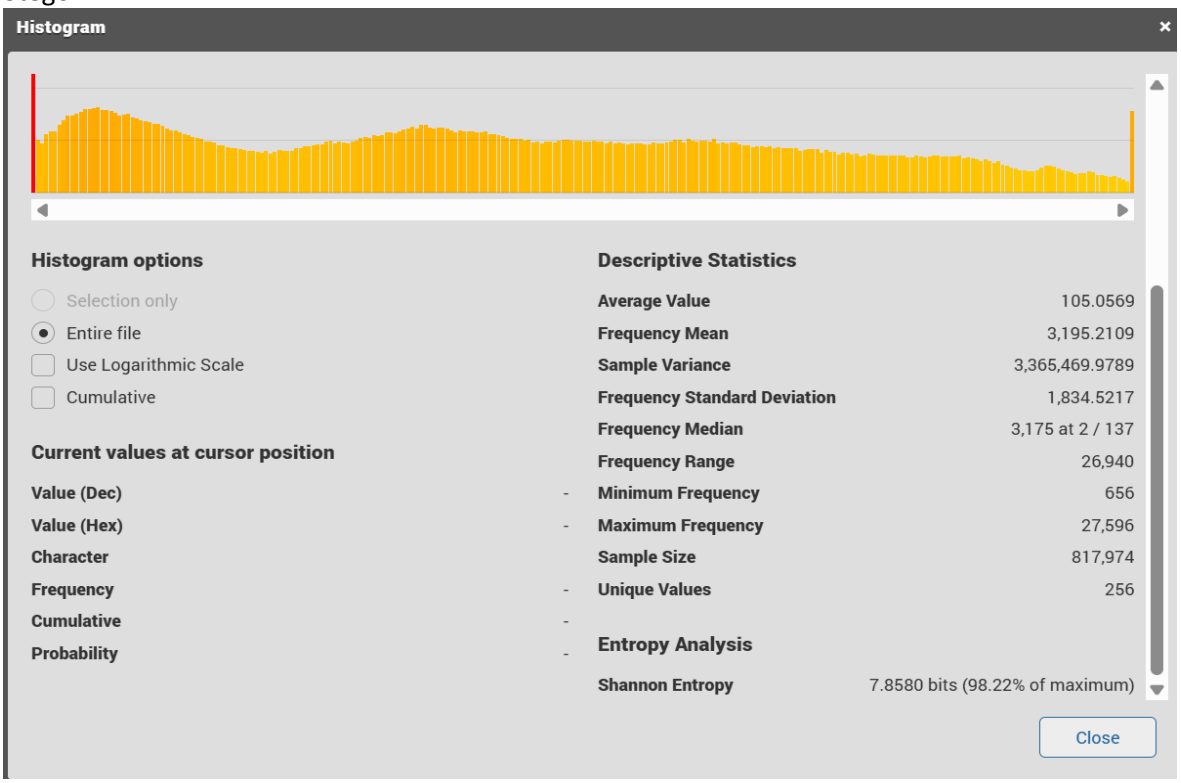
A. Click Tools after loading images (at https://hexed.it) and then click Histogram to generate the histograms of both (original and Stego) files; [2 points]

B. Scroll down the histograms and check Descriptive Statistics

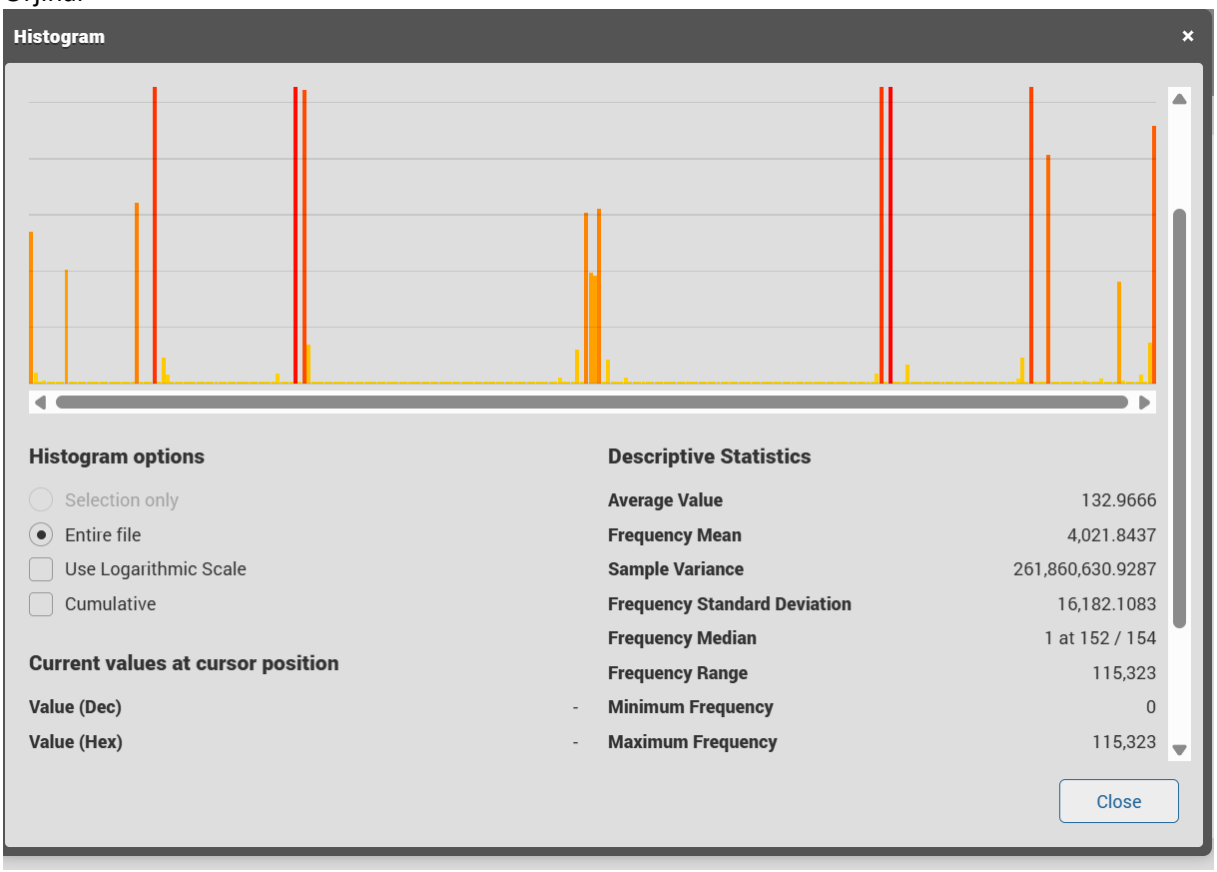
orjinal:



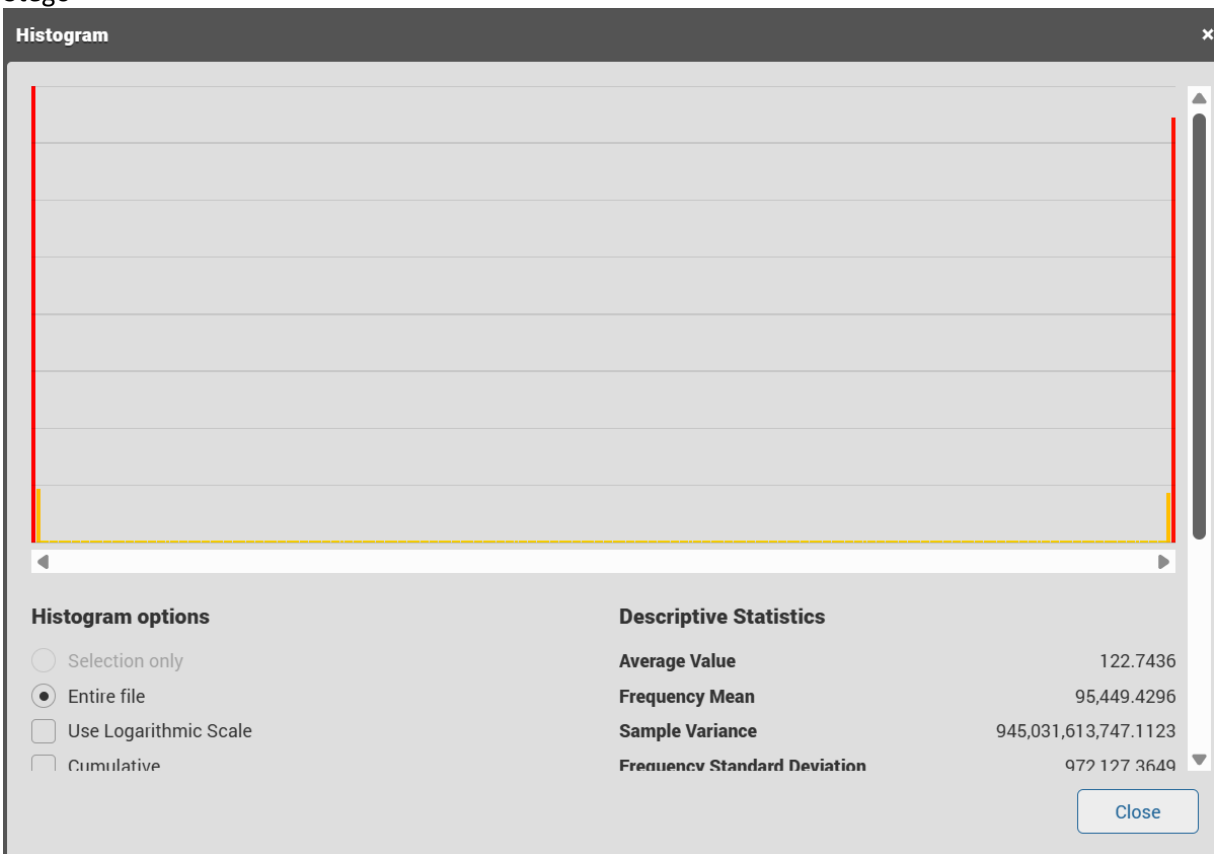
Stego

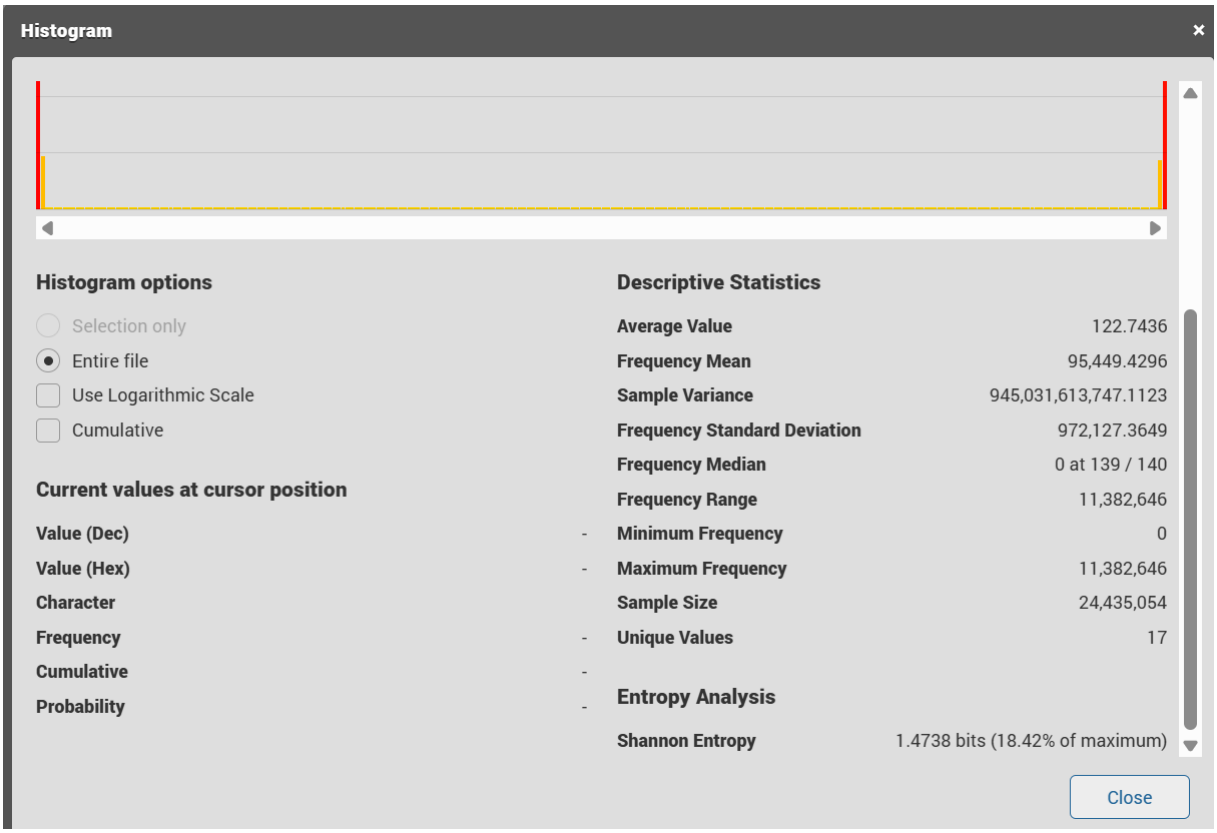


Orjinal

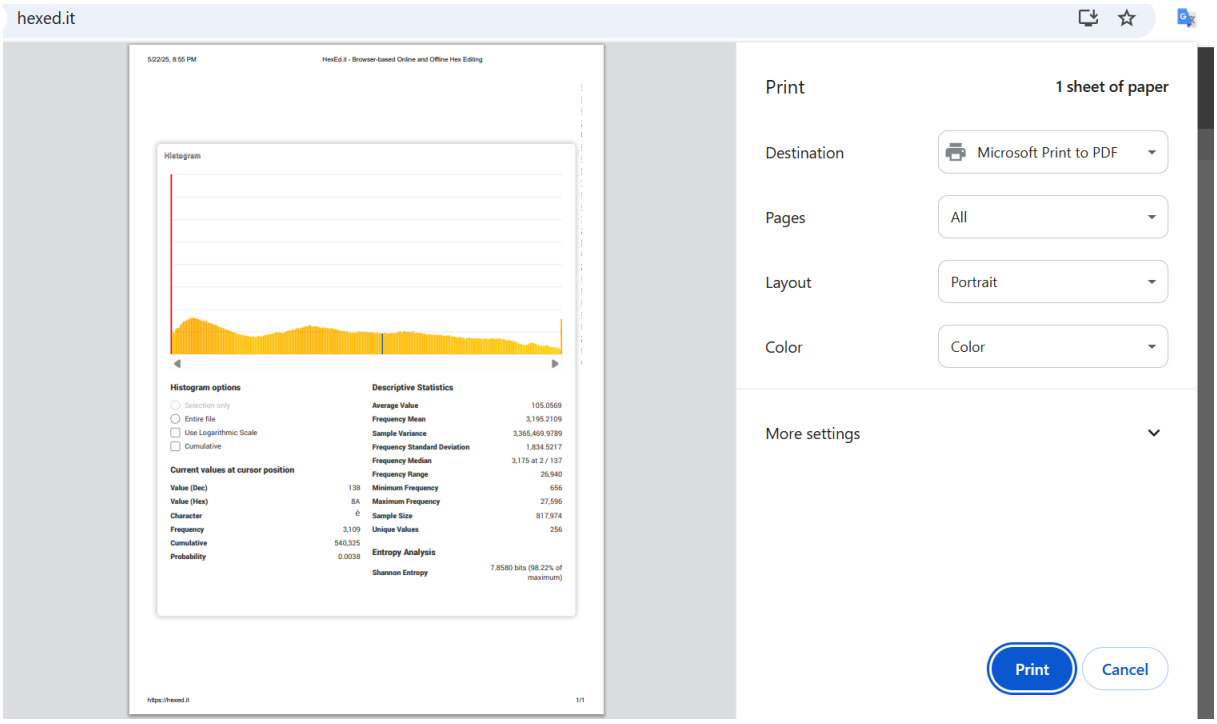


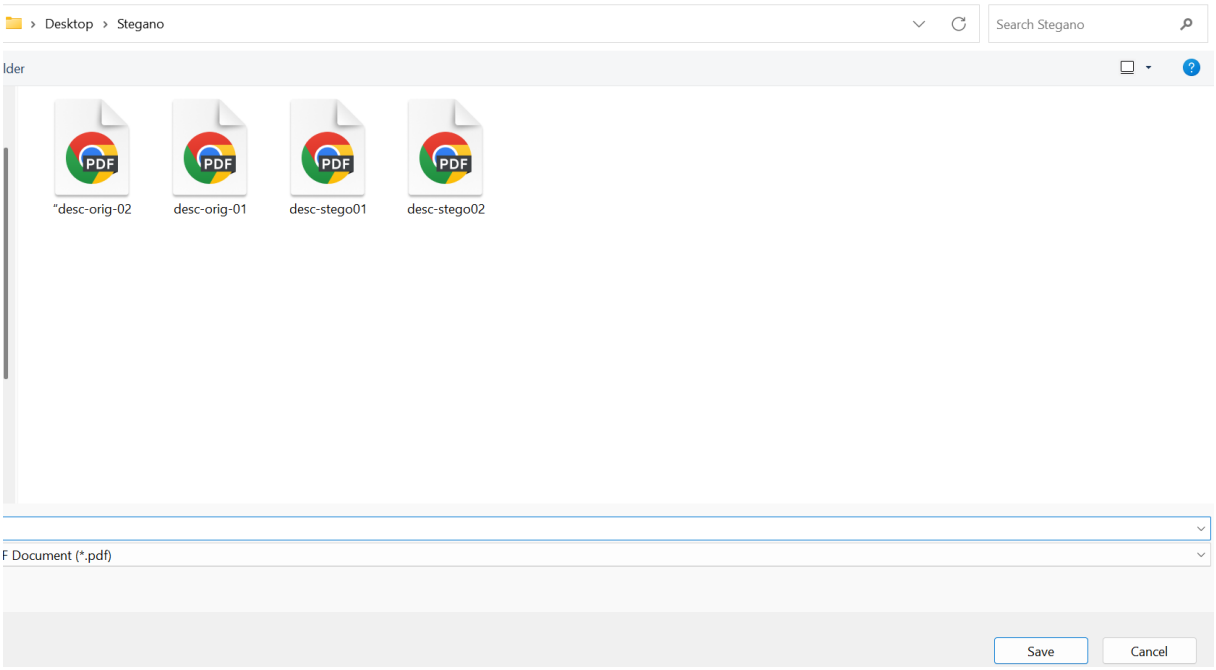
Stego





C. Right click the histogram and click “Print” and save the file as “desc-orig-01.pdf” and “desc-stego01.pdf” and …..02.pdf, …03.pdf files for the originals and stego files respectively. [2 points]





D. Open both .pdf file and compare descriptive statistics difference like Average values, Frequency Means, Simple variance, Sample sizes and other various features of both files which are different. Highlight them. [2 points]

Statistic	File 1	File 2	Difference
Average Value	105.0522	105.0569	+0.0047
Frequency Mean	3,195.5390	3,195.2109	−0.3281
Sample Variance	3,376,624.6078	3,365,469.9789	−11,154.6289
Frequency Std. Deviation	1,837.5594	1,834.5217	−3.0377
Frequency Median	3,175 at 2 / 137	3,175 at 2 / 137	Identical
Frequency Range	27,001	26,940	−61
Minimum Frequency	652	656	+4
Maximum Frequency	27,653	27,596	−57
Sample Size	818,058	817,974	−84
Unique Values	256	256	Identical

Cover(file1) has a slightly larger sample size and slightly higher maximum frequency, variance, and standard deviation.
Stego (file2) however, contains a hidden file embedded using steganography. Despite this, the statistical differences between the two files are very small.
Both files have similar sample sizes (818,058 for Cover and 817,974 for Stego) and almost identical Shannon entropy values (7.8578 and 7.8580 bits), showing that both files are highly randomized. The high entropy in File 2 helps conceal the presence of hidden data, making it difficult to detect visually or statistically.
In summary, Stego successfully hides extra information without causing major statistical changes, making the steganographic technique effective and stealthy**.**

Report Step 6
Steganalysis (Descriptive Statistics Comparison)

In this step, I analyzed the descriptive statistics of the original and stego image files using the histogram tool in HexEd.it.

A. Histogram & Statistics Generation
After loading each image file on HexEd.it, I used:
Tools → Histogram **to generate the statistical view for both the cover and stego files.**

B. Saving PDF Reports
I right-clicked the histogram and selected Print**, saving each as:**

- `desc-orig-01.pdf` & `desc-stego01.pdf`
- `desc-orig-02.pdf` & `desc-stego02.pdf`
- `desc-orig-03.pdf` & `desc-stego03.pdf`

C. Statistical Comparison

I opened and compared the PDF files side-by-side, focusing on:

- **Average values**
- **Frequency means**
- **Simple variance**
- **Sample sizes**

Highlighted all noticeable differences, which reflect the statistical changes caused by hidden data within the stego files.

Step#7 Reporting the Results [6 points]
The reports for each step have been added accordingly

Conclusion

Through detailed steganalysis, I compared cover and stego images using multiple methods—text and image diff tools, histograms, HEX maps, and descriptive statistics. Each step revealed subtle but measurable differences introduced by hidden data. These findings confirm that steganographic embedding alters file structure, pixel data, and statistical patterns, even when visual changes are minimal.