

Cryptographic protocol L1

Project

Aghamir Ahmadov

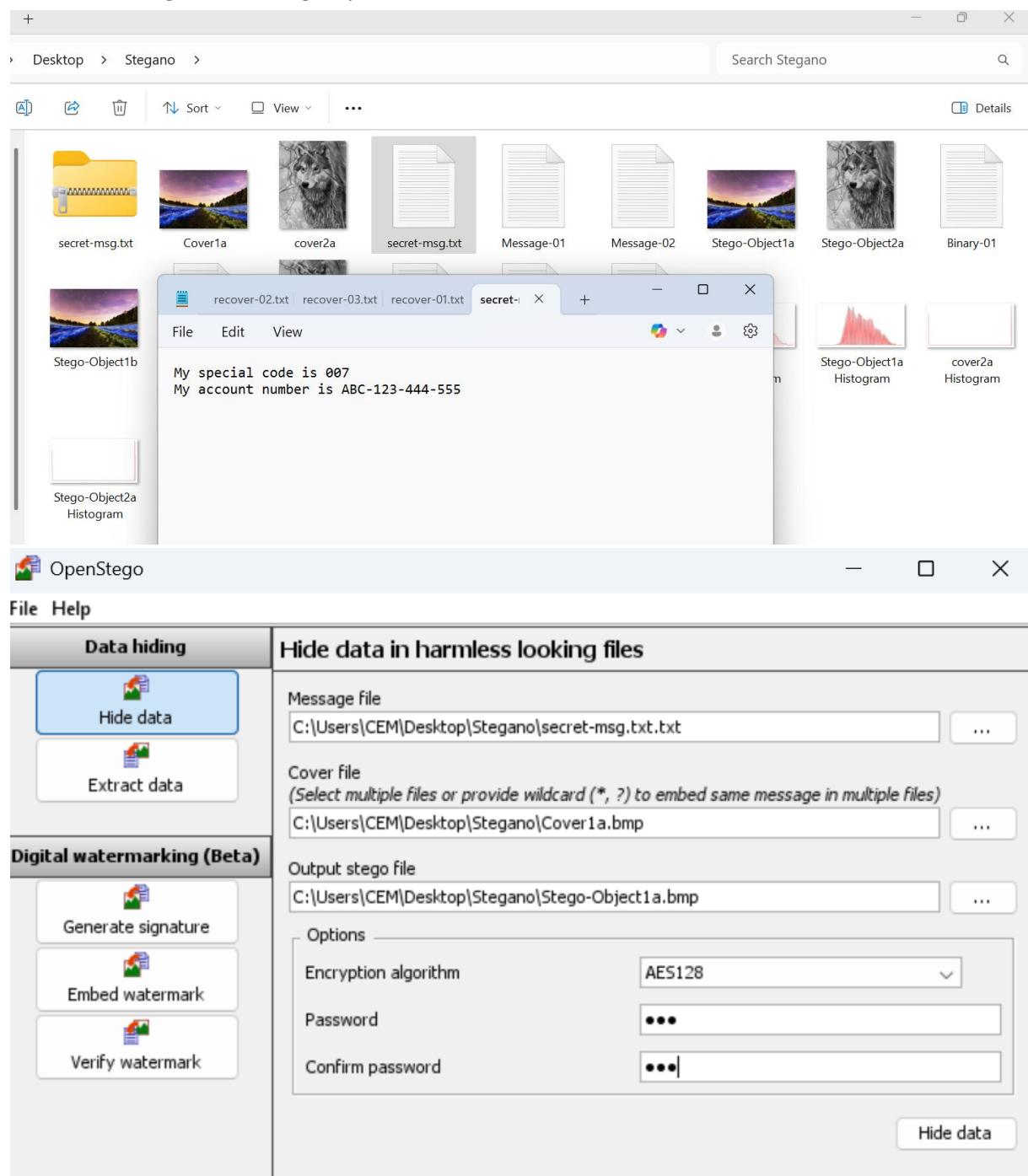
ID 49729

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	

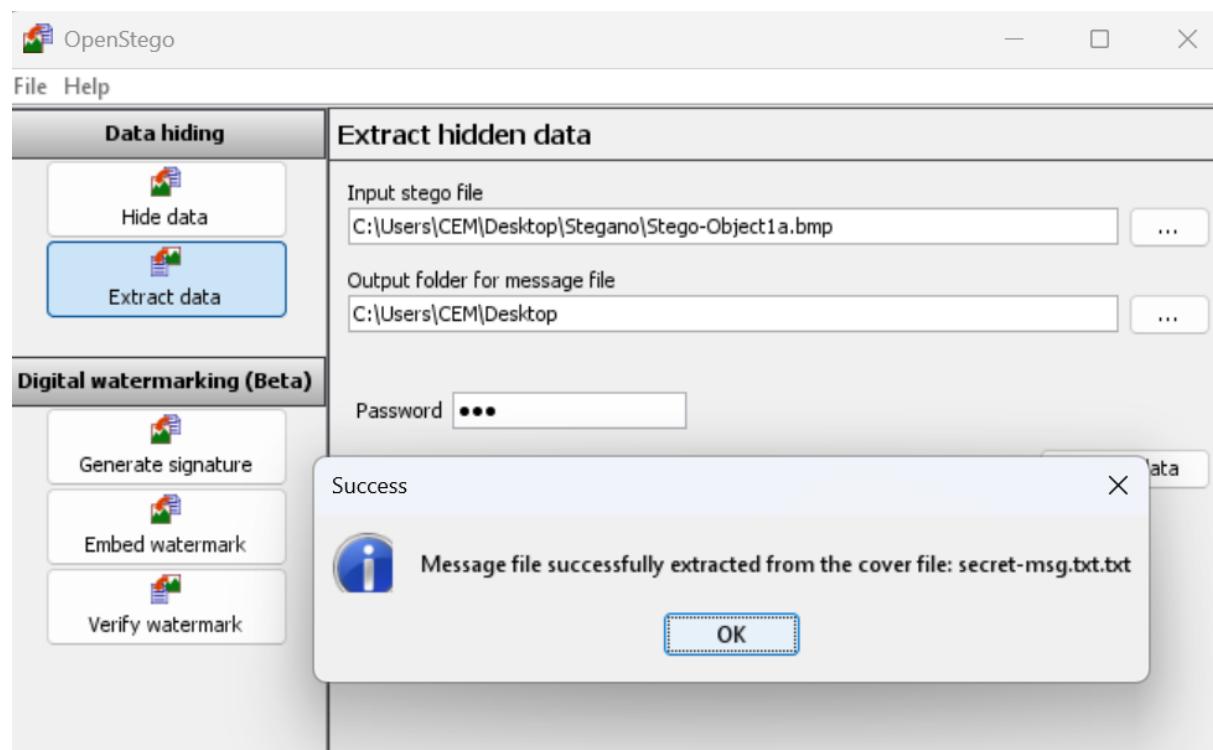
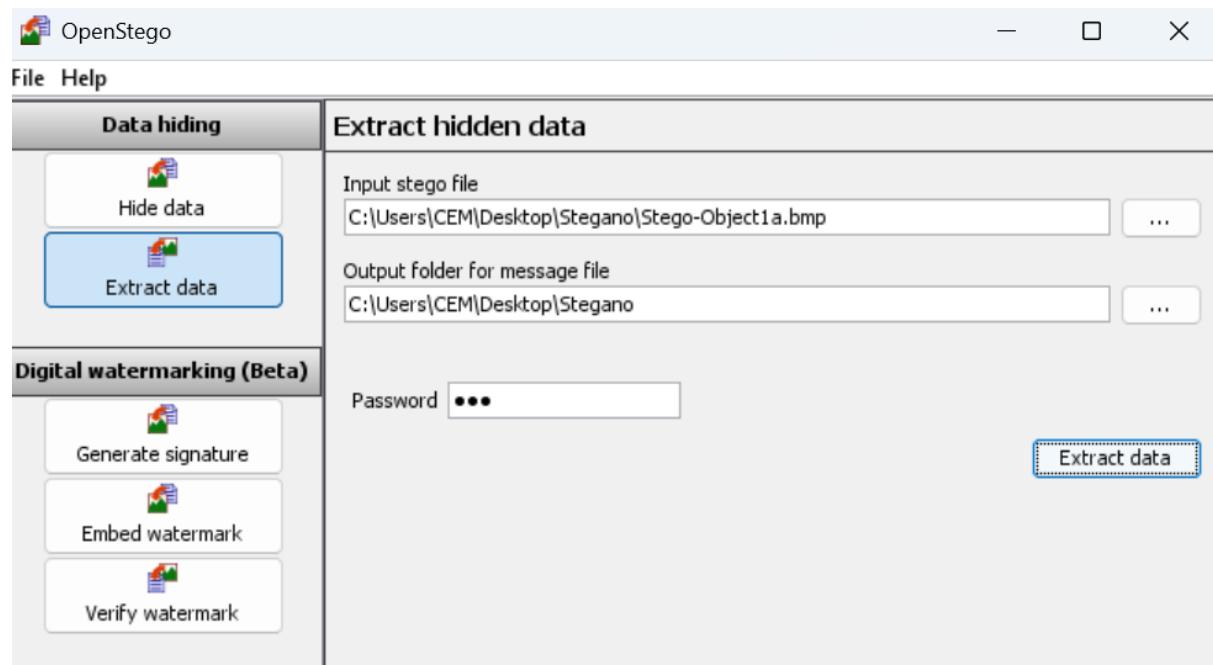
Step #1

Steganography

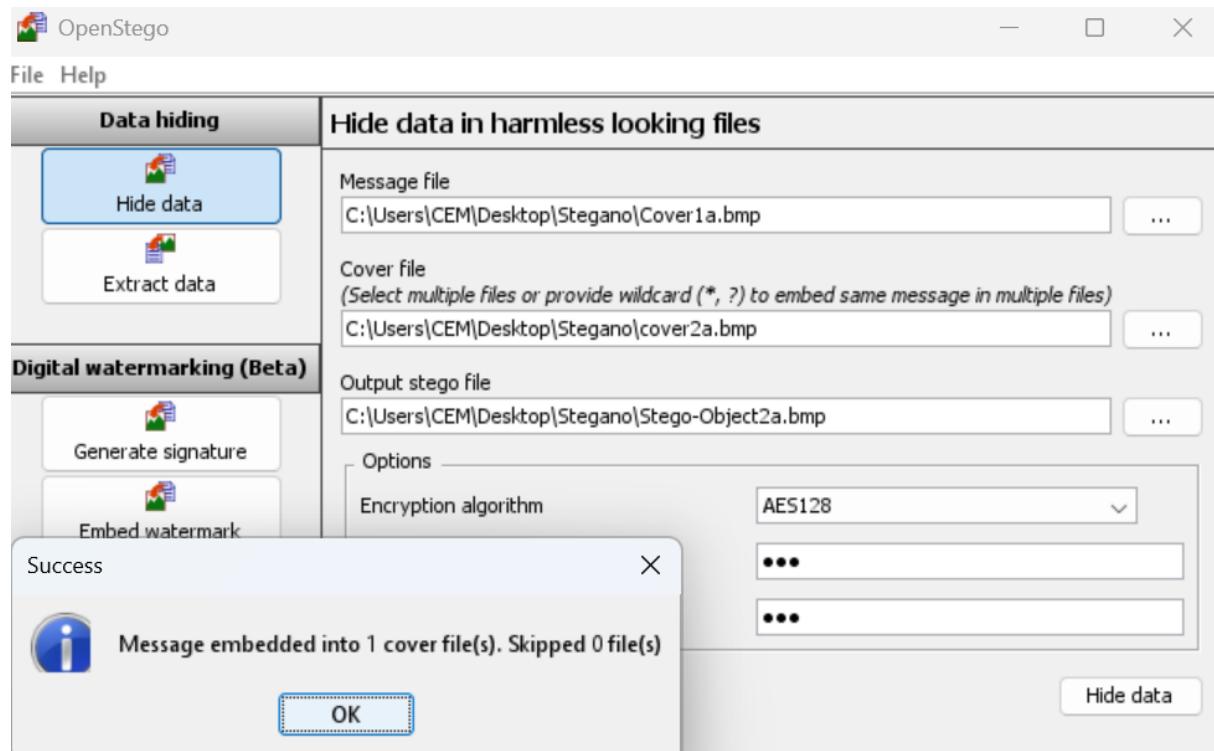
A A text message within image 3 point



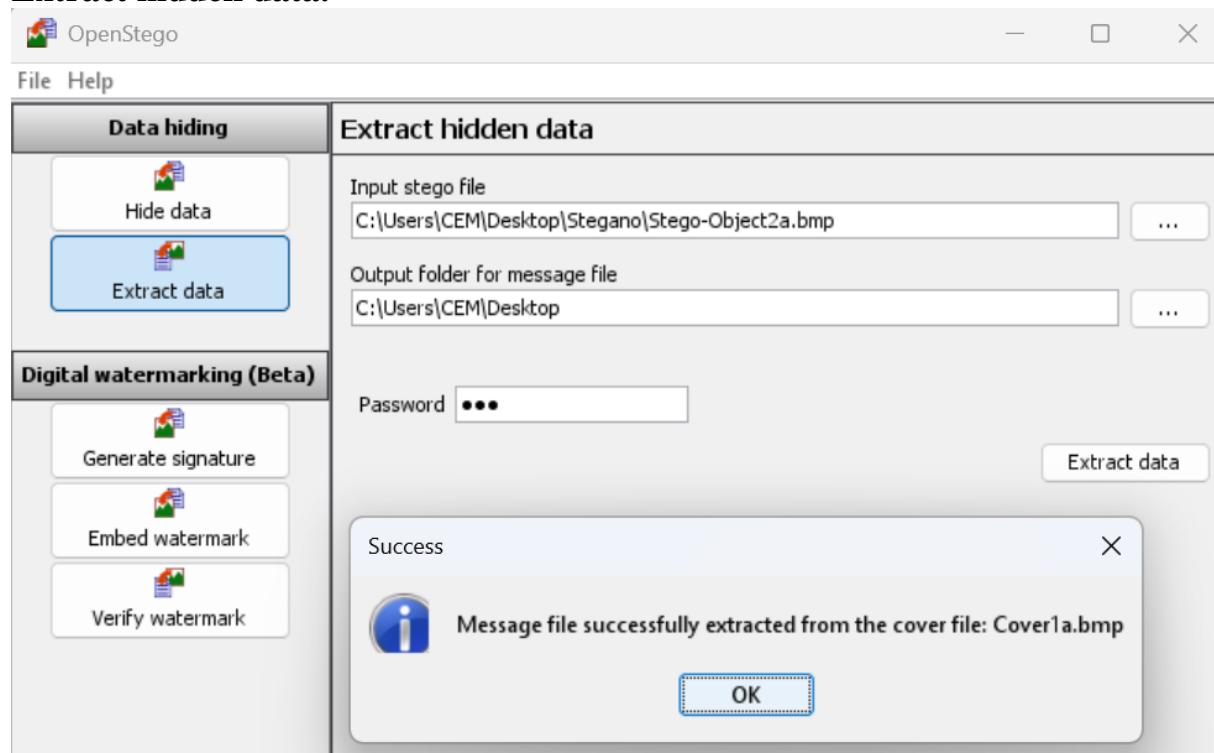
EXTRACTING



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]

Choose File

Cybersecurity is the practice of protecting systems, networks, and data from digital attacks. It involves tools, processes, and training to prevent unauthorized access, ensure data integrity, and defend against evolving cyber threats.

Encode

Binary representation of your message

```
010000110111001011000100110010101110010011100110110010101100011011101010111001001101001011101001111001001000001101001011100110010000011101001101110100110000111001001101100100101001000001101110110011000011100001110010011011101
```

Original



Message hidden in image (right click → save as)



2.a

Choose File cover2a.bmp

cover2a.bmp

Cybersecurity defends computers, networks, and data from cyber threats. It covers areas like ethical hacking, encryption, malware analysis, and system protection to ensure digital safety and privacy in an increasingly connected world.

Encode

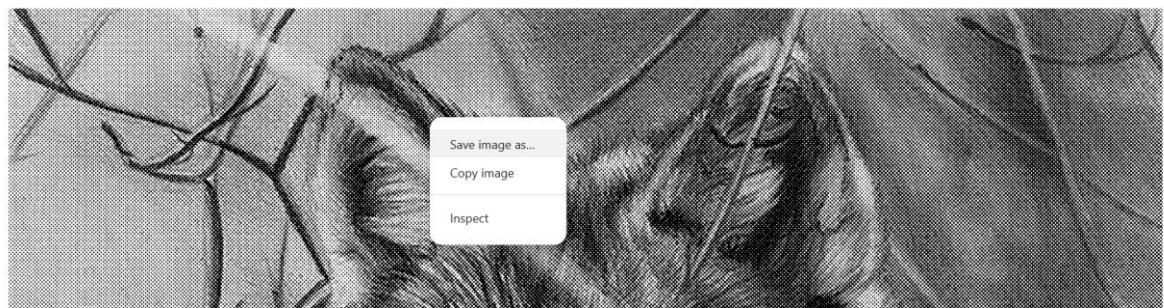
Binary representation of your message

```
0110100101101110001000001100001011011100100000110100101101110011001110010011001010110001011100110110100101110011001110110110001111001  
010000011000110110111101101110011001010110001101110100011001010110010000100000011101110110111100100110100010111011011100101110110001111001
```

Original



Message hidden in image (right click → save as)



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]

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]

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.

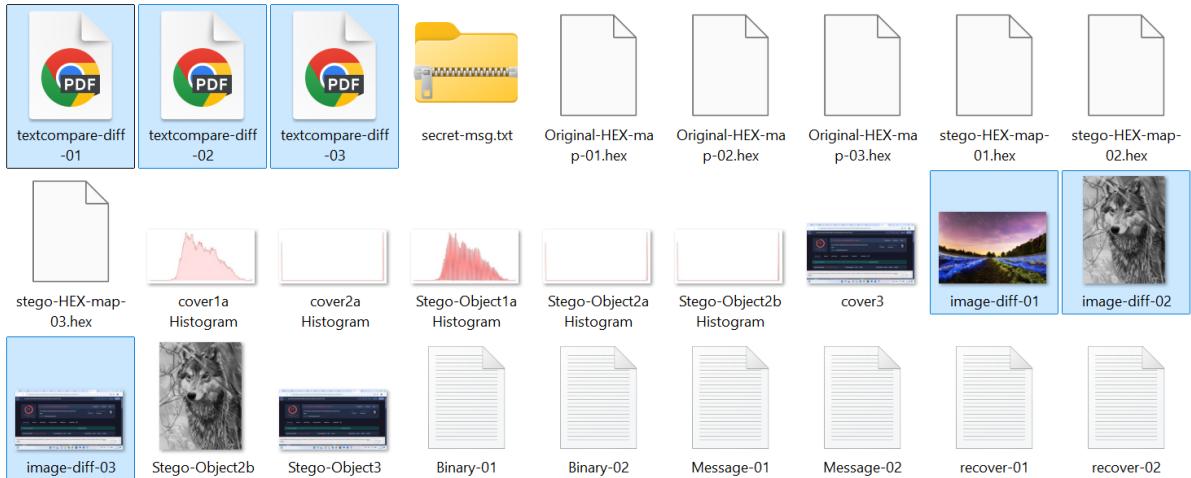
The screenshot shows the main interface of the Image Compare Tool. At the top, there are two large image thumbnails. The left thumbnail is a landscape at night with star trails and a field of blue flowers. Below it is a "CHOOSE ORIGINAL FILE" button with a cloud icon. The right thumbnail is a close-up of a wolf's face. Below it is a "CHOOSE MODIFIED FILE" button with a cloud icon. Below the thumbnails is a toolbar with buttons for "ADD SAMPLE" (with a plus sign), "CLEAR" (with a trash can icon), "SWAP" (with a double arrow icon), and a large blue "COMPARE" button with a magnifying glass icon. The background has a subtle watermark of a person using a laptop.

The screenshot shows the navigation bar of TextCompare Pro. It includes the logo, "TextCompare Pro", and tabs for TEXT, SPREADSHEET, DOCUMENT, IMAGE (which is highlighted in blue), CSV, PRICING, and DOWNLOAD. To the right of the tabs are icons for export (PDF, XLSX, CSV, ZIP), settings, and help.

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.

The screenshot shows the main interface of the Online Image Comparison Tool. It displays two versions of the same image side-by-side. The left image is a screenshot of a malware analysis tool showing a file with a community score of 64/77. The right image is a similar screenshot from a different angle or time. Between the images are several comparison controls: "DIFF" (highlighting differences), "SLIDER" (a horizontal slider for comparing the images), "FADE" (a circular icon for a fading effect), "Differences" (a small preview of the differences), and "DETAILS" (a gear icon for settings). Below these are buttons for "RESET" (with a circular arrow icon), "0.00% Changes" (with a minus sign icon), "Resize to same size" (with a gear icon), and "Expiry Hour" (with a dropdown menu). At the bottom are "SAVE ONLINE" (with a cloud icon) and "VERIFY EMAIL TO EXPORT" (with a download icon).



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](https://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...

Q Q X Q +



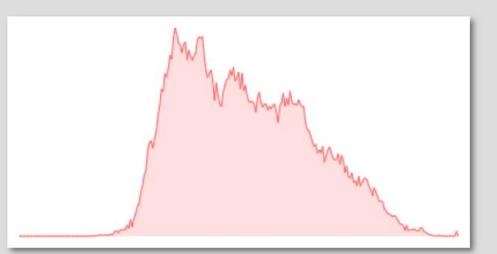
GENERATE!

Channel

Red Green Blue Luminosity

HISTOGRAM

Q Q X Q + H PNG JPG WEBP



Choose File Stego-Object1b.bmp

Paste an image or image URL here

Drop an image here...

Q Q X Q +



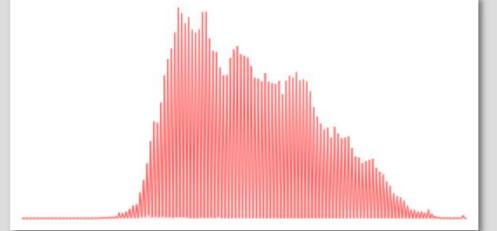
GENERATE!

Channel

Red Green Blue Luminosity

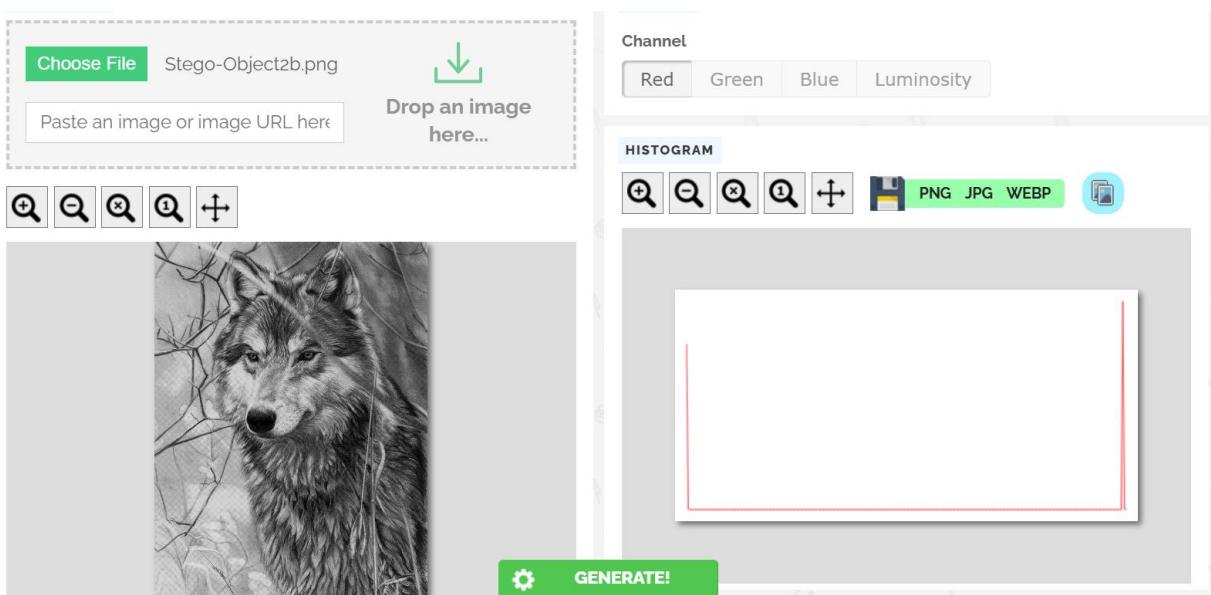
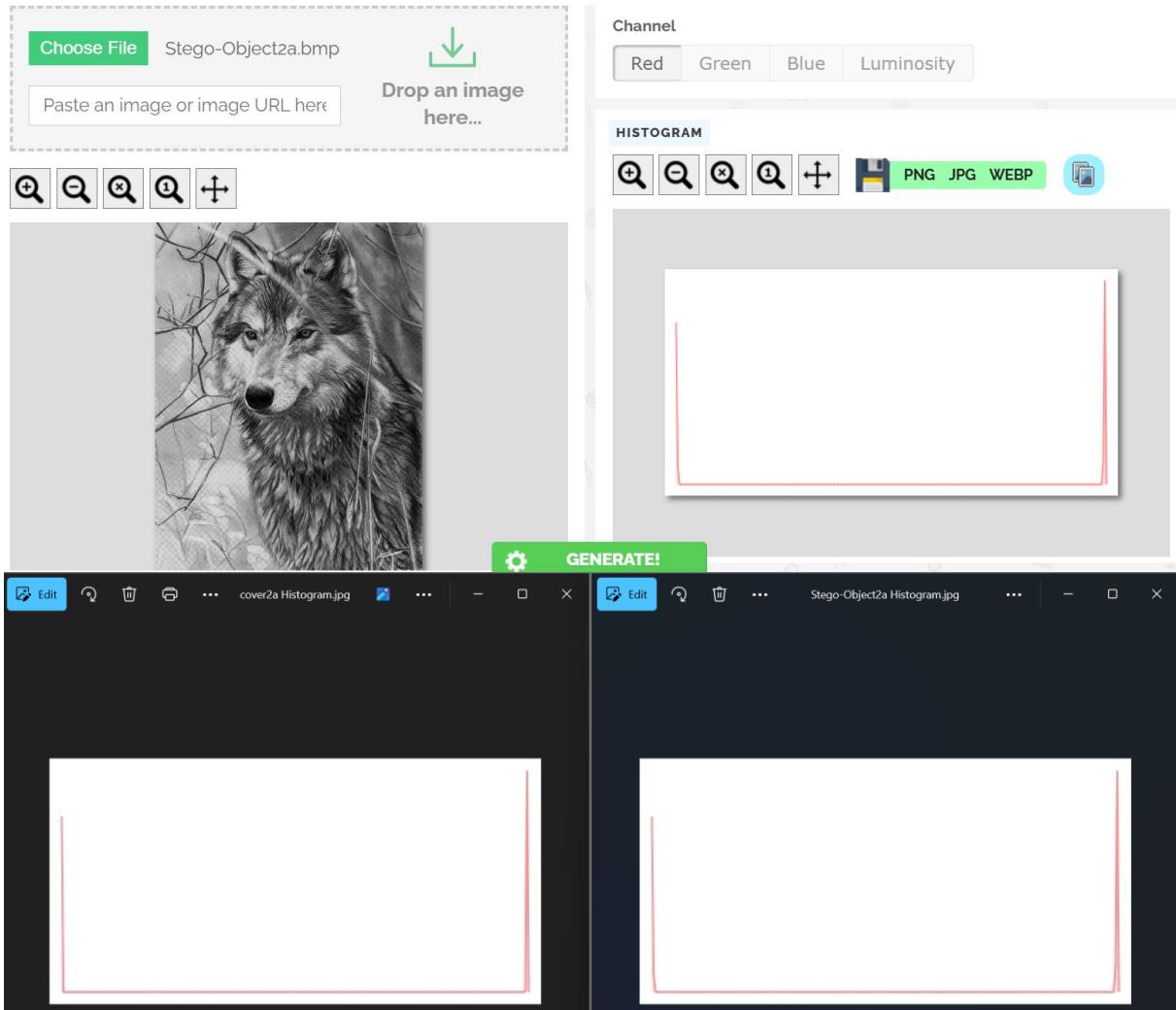
HISTOGRAM

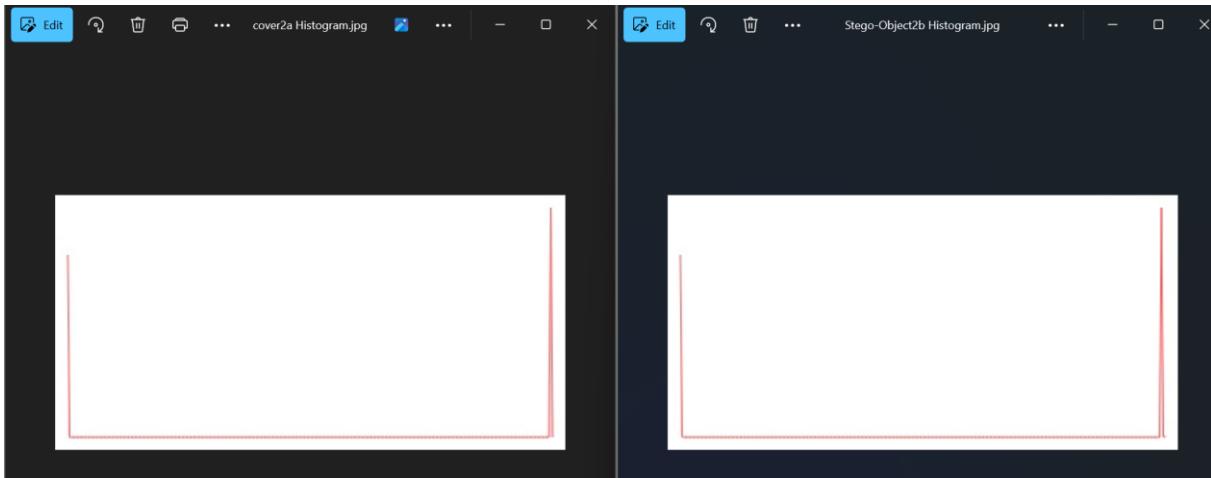
Q Q X Q + H PNG JPG WEBP





This screenshot shows a user interface for generating a histogram. On the left, there is a file input field labeled "Choose File" with "cover2a.bmp" selected, and a "Drop an image or image URL here" input field with a green "Drop here..." icon. Below these are five small icons: a magnifying glass, a double magnifying glass, a single magnifying glass, a question mark, and a plus sign. To the right, there is a "Channel" selection bar with "Red" selected and "Green", "Blue", and "Luminosity" options. A "HISTOGRAM" section contains several icons: a magnifying glass, a double magnifying glass, a single magnifying glass, a question mark, a plus sign, a floppy disk, and file format buttons for "PNG", "JPG", and "WEBP". A blue "GENERATE!" button is located at the bottom of this section. The central area of the interface shows a grayscale image of a wolf's head.





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

File Information		Cover1a.bmp	Stego-Object1a.bmp
File Name	Cover1a.bmp	00000000 42 4D 8A 7B 0C 00 00 00 00 00 00 00 00 00 00	00000000 42 4D 36 7B 0C 00 00 00 00 00 00 00 00 00 00
File Size	818,058 bytes (799 KiB)	00000010 00 00 80 02 00 00 AA 01 00 00 01 00 18 00 00 00	00000010 00 00 80 02 00 00 AA 01 00 00 01 00 18 00 00 00
Data Inspector (Little-endian)			
Type	Unsigned (+)	Signed (±)	
8-bit Integer	66	66	00000020 00 00 00 7B 0C 00 00 00 00 00 00 00 00 00 00 00
16-bit Integer	19778	19778	00000030 00 00 00 00 00 00 00 00 00 00 FF 00 00 FF 00 00 FF 00
24-bit Integer	9063746	-7713470	00000040 00 00 00 00 00 FF 42 47 52 73 80 C2 F5 28 60 B8
32-bit Integer	2072661314	2072661314	00000050 1E 15 20 85 EB 01 48 33 33 13 80 66 66 26 40 66
64-bit Integer (+)	53612268866		00000060 66 06 A0 99 99 09 3C 0A D7 03 24 5C 8F 32 00 00
64-bit Integer (±)	53612268866		00000070 00 00 00 00 00 00 00 00 00 00 00 00 04 00 00 00 00
16-bit Float. P.	21.03125		00000080 00 00 00 00 00 00 00 00 00 00 01 06 04 04 09 07
32-bit Float. P.	1.4362079e+36		00000090 06 08 09 04 06 07 05 04 08 07 06 08 0A 07 09 0A
64-bit Float. P.	2.648798024229484e-313		000000A0 08 07 0C 09 05 09 06 01 0E 0A 05 11 0C 09 0D 08
LEB128 (+)	66		000000B0 07 0B 04 07 10 00 0C 0B 04 0B 05 09 0E 11 15
LEB128 (±)	-62		000000C0 15 17 18 0E 10 11 0E 0C 0C 12 10 10 16 10 11 11
Rational (+)	172721776.167		000000D0 0B 0C 14 0F 10 12 10 10 0E 0E 0E 0A 0F 0E 05 0D
SRational (±)	172721776.167		000000E0 00 00 08 03 0F 11 11 20 22 4D 5E 61 49 5C 5F
			000000F0 1D 2C 2E 01 0E 10 08 12 12 08 0D 0E 06 0B 0A 0E
			00000100 18 10 0D 0B 0A 09 07 06 0F 0D 0C 0D 0B 0A 0D 0D
			00000110 00 10 10 08 0A 0B 08 0C 0D 0B 0F 00 04 07
			00000120 02 07 0A 07 0C 0F 12 17 1A 11 16 19 07 0E 11 11
			00000130 18 1B 35 3C 3F 0E 15 18 01 0A 0D 0A 13 16 06 0F
			00000140 12 08 11 14 06 0F 12 2A 36 38 2D 39 3B 08 15 17
			00000150 0C 18 1A 39 45 47 15 1C 1F 0C 13 16 15 1A 1D 10
			00000160 15 18 1F 24 27 00 05 08 0D 14 17 06 0F 12 17 20

stego

File Information		Cover1a.bmp	Stego-Object1a.bmp
File Name	Stego-Object1a.bmp	00000000 42 4D 8A 7B 0C 00 00 00 00 00 00 00 00 00 00	00000000 42 4D 36 7B 0C 00 00 00 00 00 00 00 00 00 00
File Size	817,974 bytes (799 KiB)	00000010 00 00 80 02 00 00 AA 01 00 00 01 00 18 00 00 00	00000010 00 00 80 02 00 00 AA 01 00 00 01 00 18 00 00 00
Data Inspector (Little-endian)			
Type	Unsigned (+)	Signed (±)	
8-bit Integer	54	54	00000020 00 00 00 7B 0C 00 00 00 00 00 00 00 00 00 00 00
16-bit Integer	31542	31542	00000030 00 00 00 00 00 00 00 01 06 04 04 09 07 06 08 09
24-bit Integer	817974	817974	00000040 06 07 05 04 08 07 06 08 0A 07 09 0A 08 07 0C 09
32-bit Integer	817974	817974	00000050 05 09 06 01 0E 0A 05 11 0C 09 0D 08 07 0B 04 07
64-bit Integer (+)	817974		00000060 10 06 0C 0B 04 0B 06 05 09 0E 11 15 15 17 18 0E
64-bit Integer (±)	817974		00000070 10 11 0E 0C 0C 12 10 10 16 10 11 11 0B 0C 14 0F
16-bit Float. P.	59072		00000080 10 12 10 10 0E 0E 0A 0F 0E 05 0D 0D 00 08 08
32-bit Float. P.	1.1462257e-39		00000090 03 0F 11 11 20 22 4D 5E 61 49 5C 5F 1D 2C 2E 01
64-bit Float. P.	4.041328525913478e-318		000000A0 0E 10 08 12 12 08 0D 0E 06 0B 0A 0E 10 10 0D 0B
LEB128 (+)	54		000000B0 0A 09 07 06 0F 0D 0C 0D 0B 0A 0D 0D 10 10 10
LEB128 (±)	54		000000C0 08 0A 0B 08 0C 0D 08 0B 0F 00 04 07 02 07 0A 07
Rational (+)	Invalid number		000000D0 0C 0F 12 17 1A 11 16 19 07 0E 11 11 18 1B 35 3C
SRational (±)	Invalid number		000000E0 3F 0E 15 18 01 0A 0D 0A 13 16 06 0F 12 08 11 14
			000000F0 06 0F 12 2A 36 38 2D 39 3B 08 15 17 0C 18 1A 39
			00000100 45 47 15 1C 1F 0C 13 16 15 1A 1D 10 15 18 1F 24
			00000110 27 00 05 08 0D 14 17 06 0F 12 17 20 23 0C 18 1A
			00000120 07 14 16 06 15 17 0D 1C 1F 01 12 15 0B 1A 1D 19
			00000130 00 00 00 00 00 00 00 00 00 00 00 00 03 0F 13 1A 27
			00000140 29 2B 38 3A 06 13 15 00 0A 0C 0E 1B 1D 13 20 22
			00000150 10 1D 1F 29 35 35 11 19 18 03 0A 07 0A 12 11 0B
			00000160 16 14 02 0C 0C 00 0C 0C 00 0D 0F 03 10 12 00 09

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:

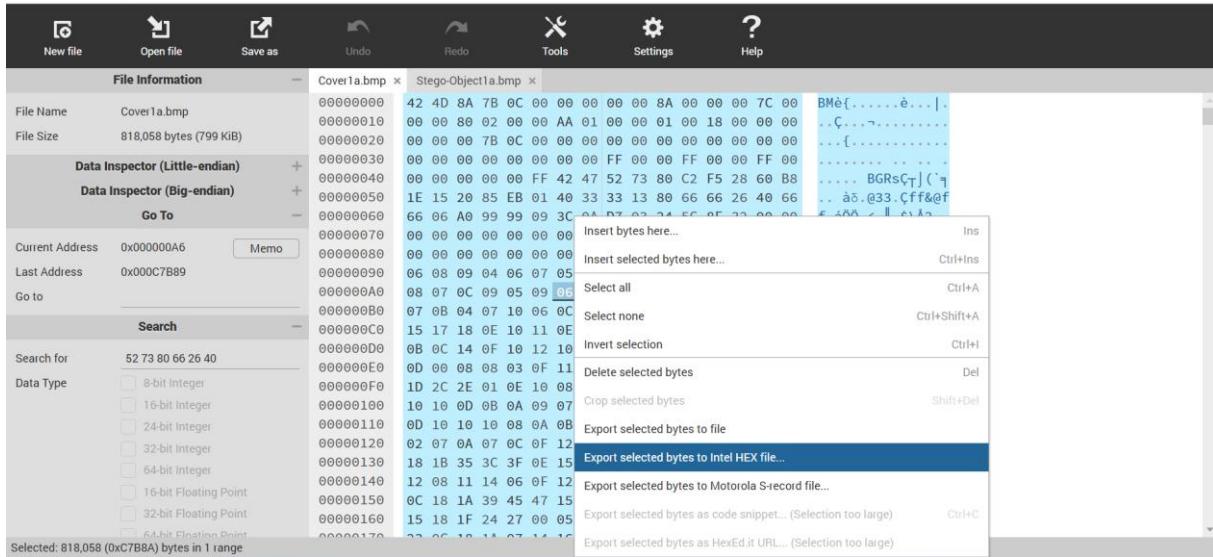
Several bytes throughout the hex map differ between the two files—most notably in the data section, not the header—confirming the presence of a hidden message.

These changes suggest the payload (hidden message) has been embedded by altering pixel bytes slightly.

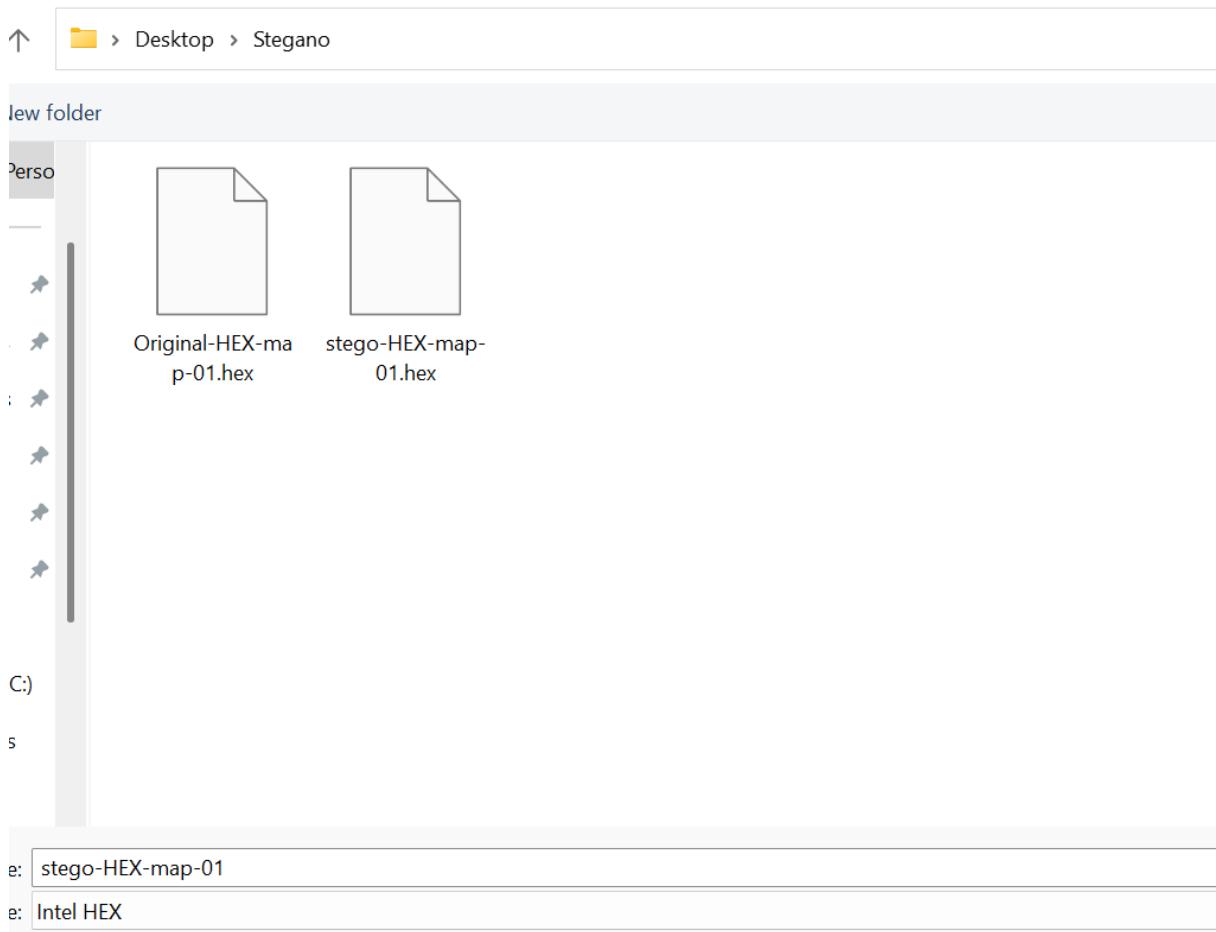
Orginal: 42 4D 8A 7B 0C 00

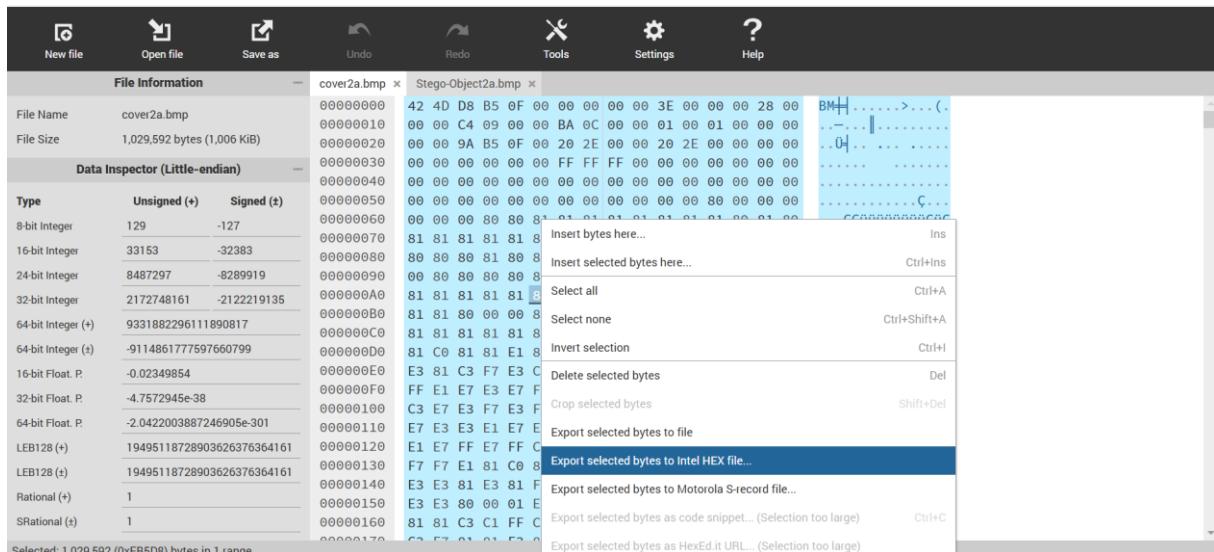
Stego: 42 4D 36 7B 0C 00

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]

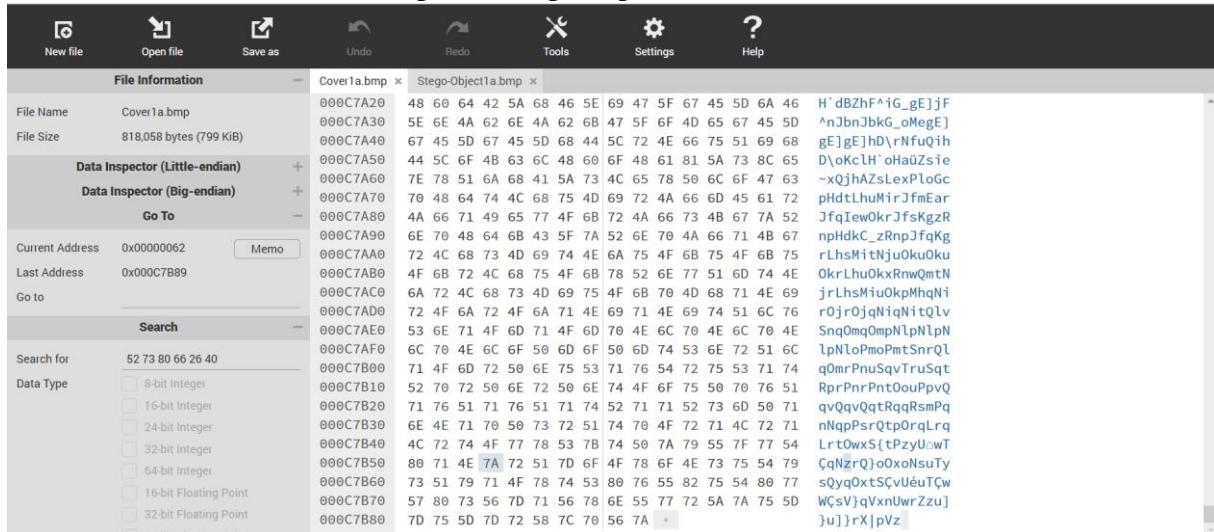


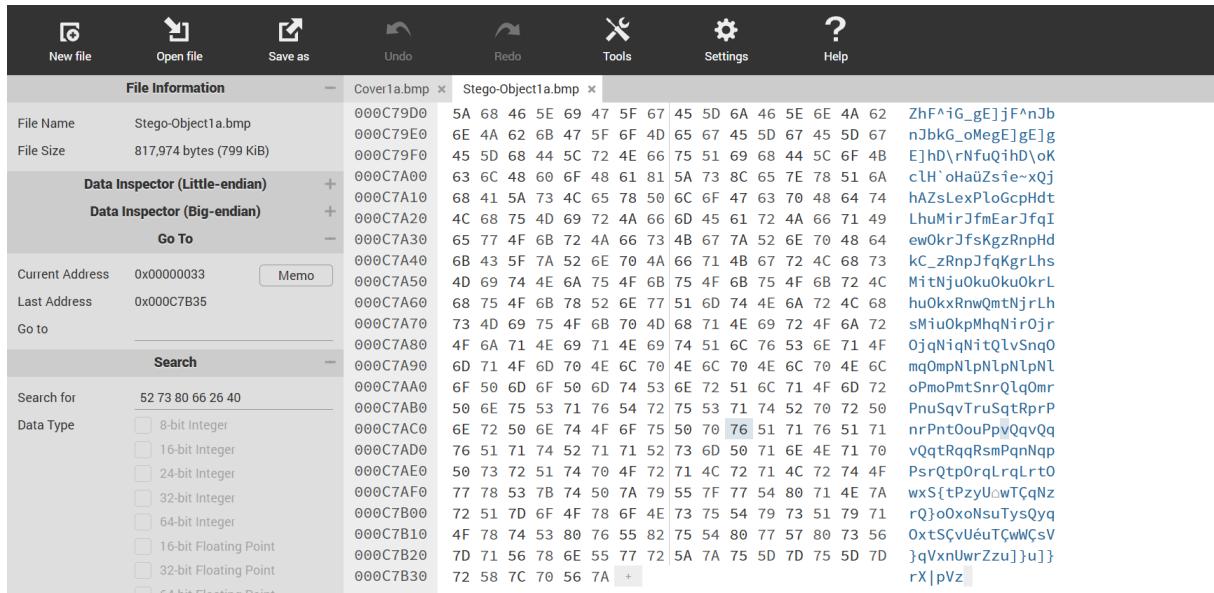
D. Save the HEX-file with Original-HEX-map-01 and stego-HEX-map-01, ...02....and03...respectively.[1 point]





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.

Original: 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

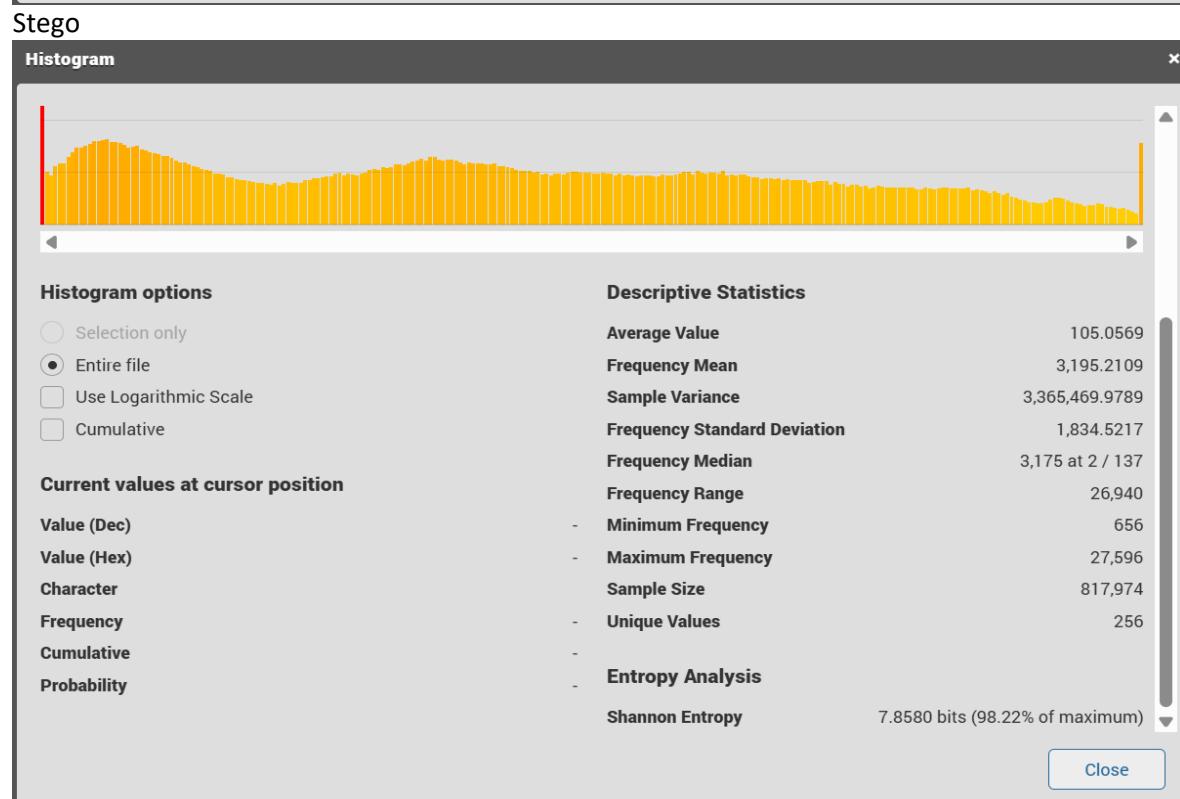
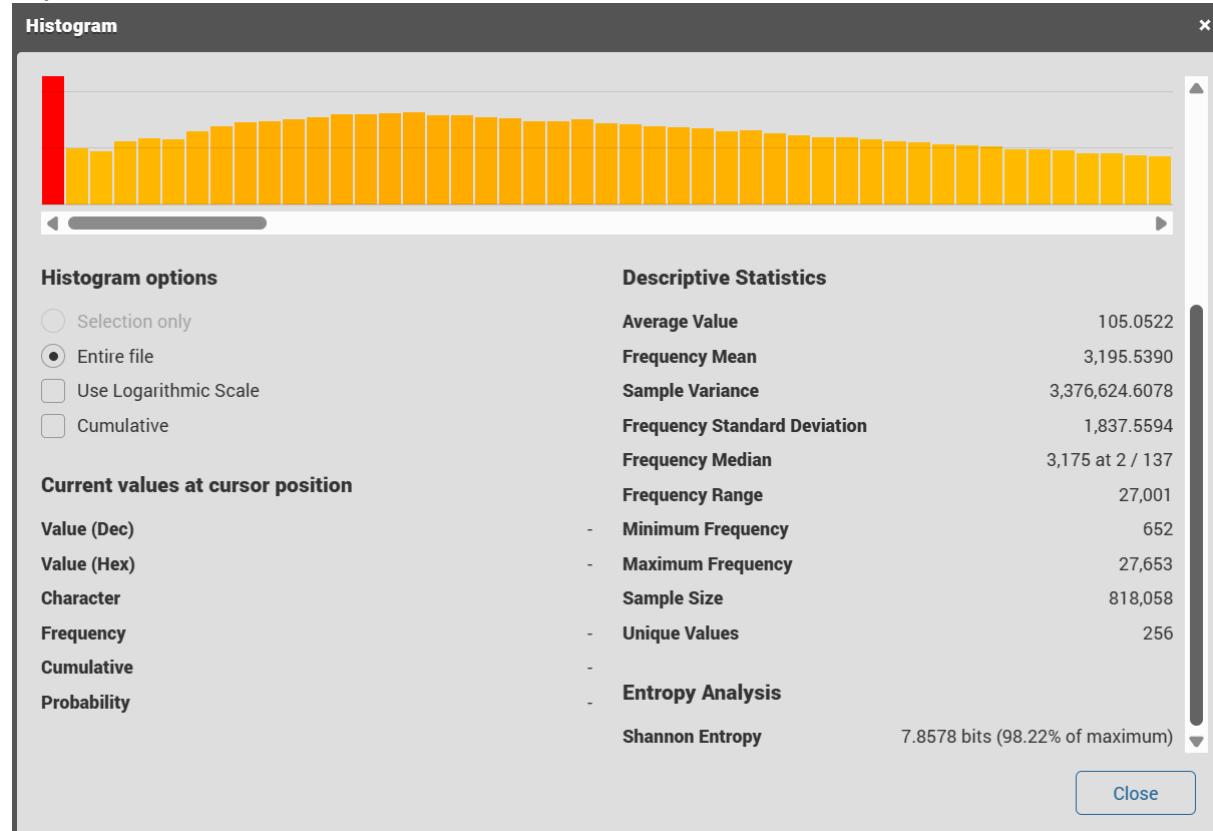
Saved each export with the following names:

Original-HEX-map-01.hex & stego-HEX-map-01.hex

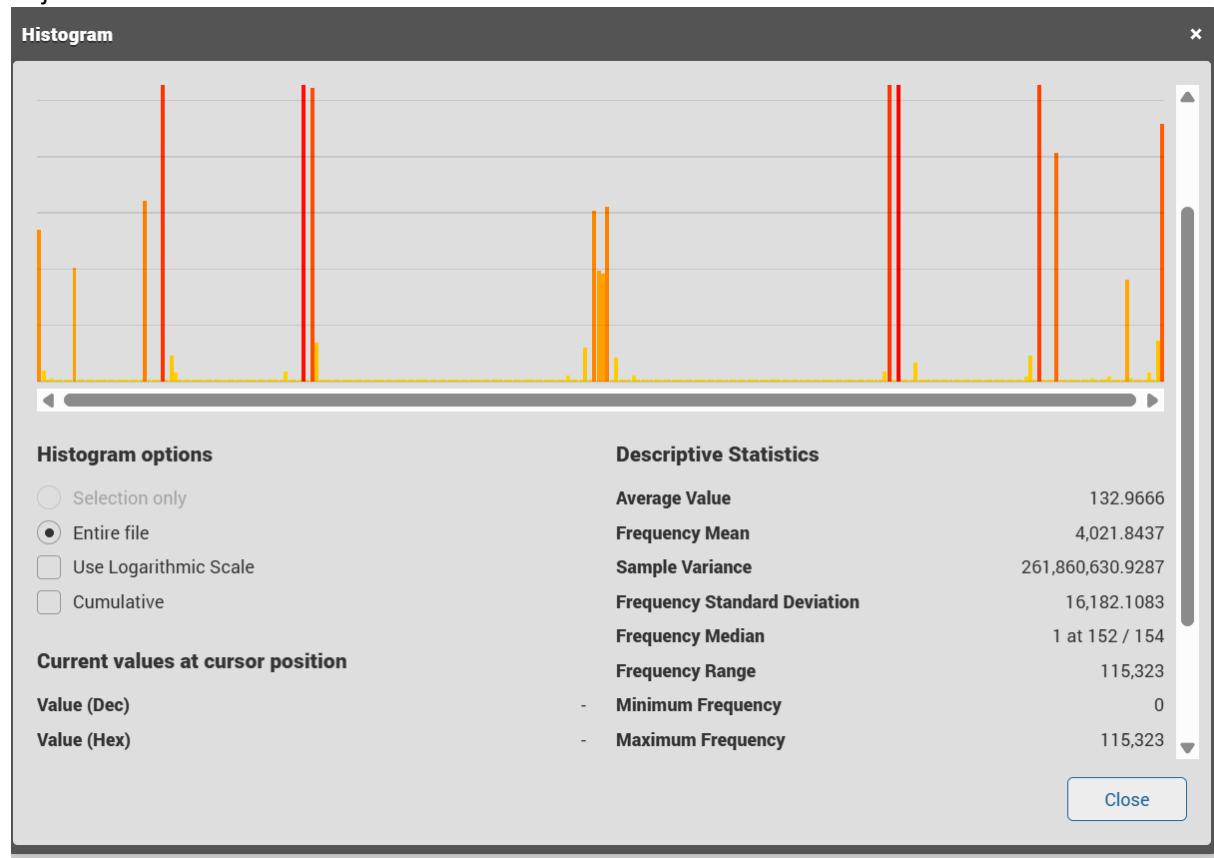
Step#6 Steganalysis (Descriptive Statistics comparison) [6 points]

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]

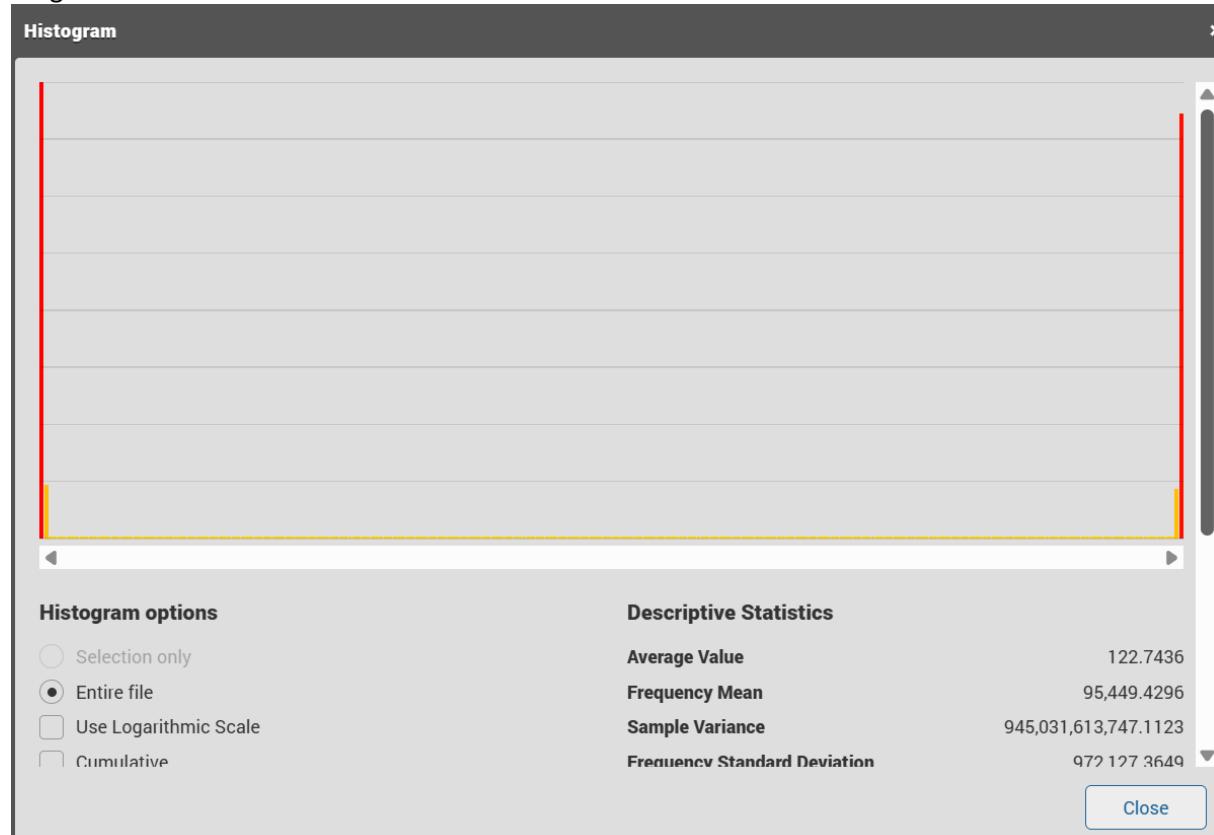
B. Scroll down the histograms and check Descriptive Statistics original:

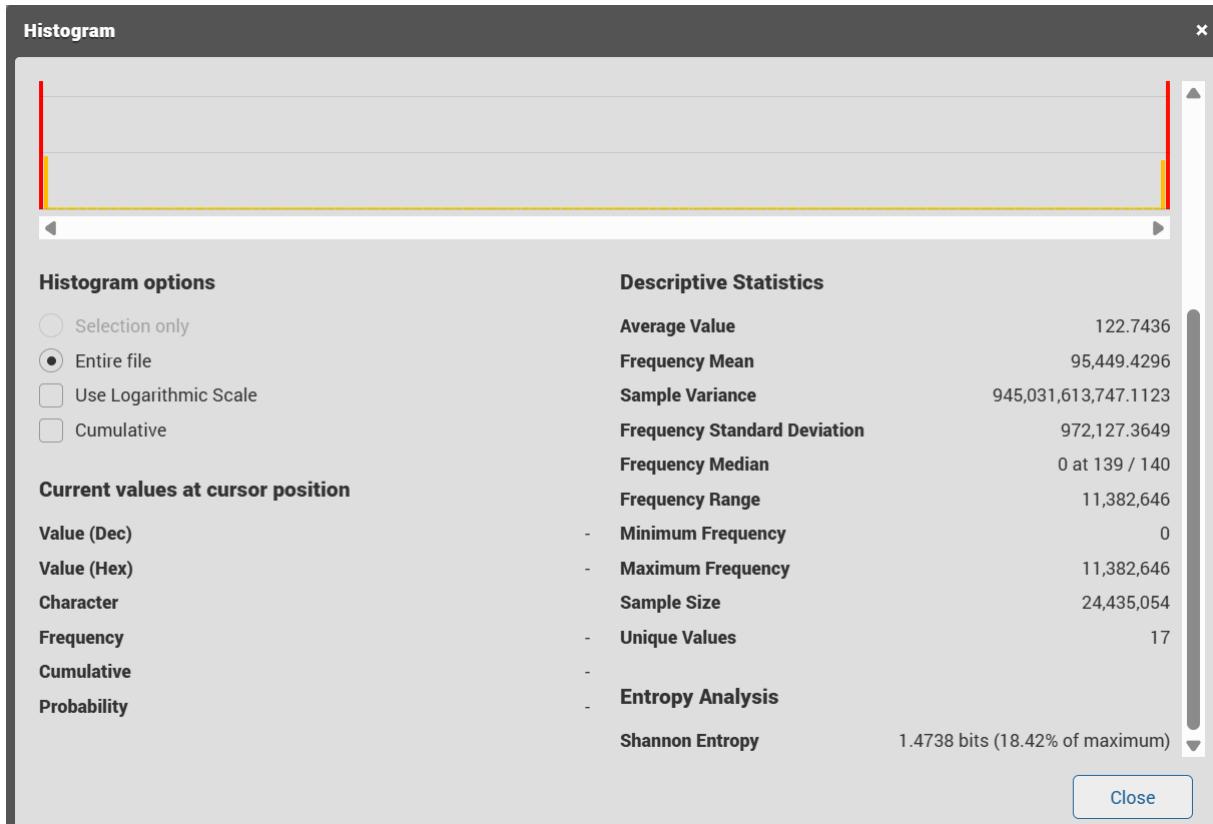


Orjinal

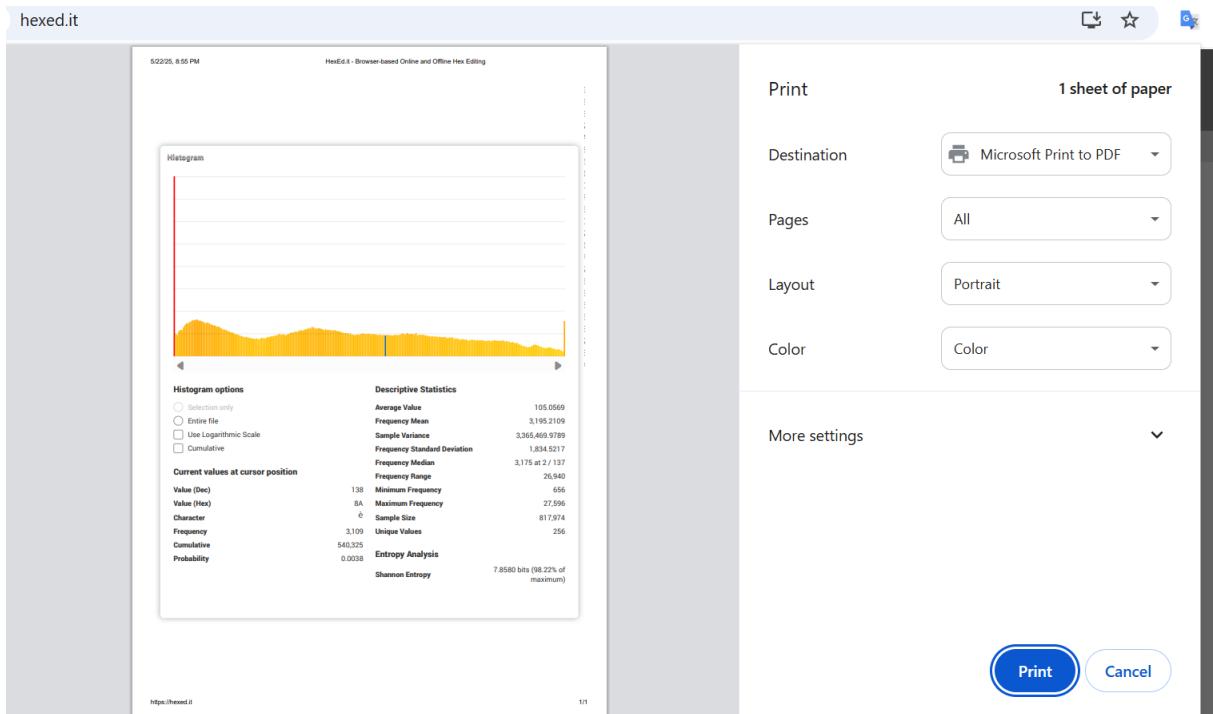


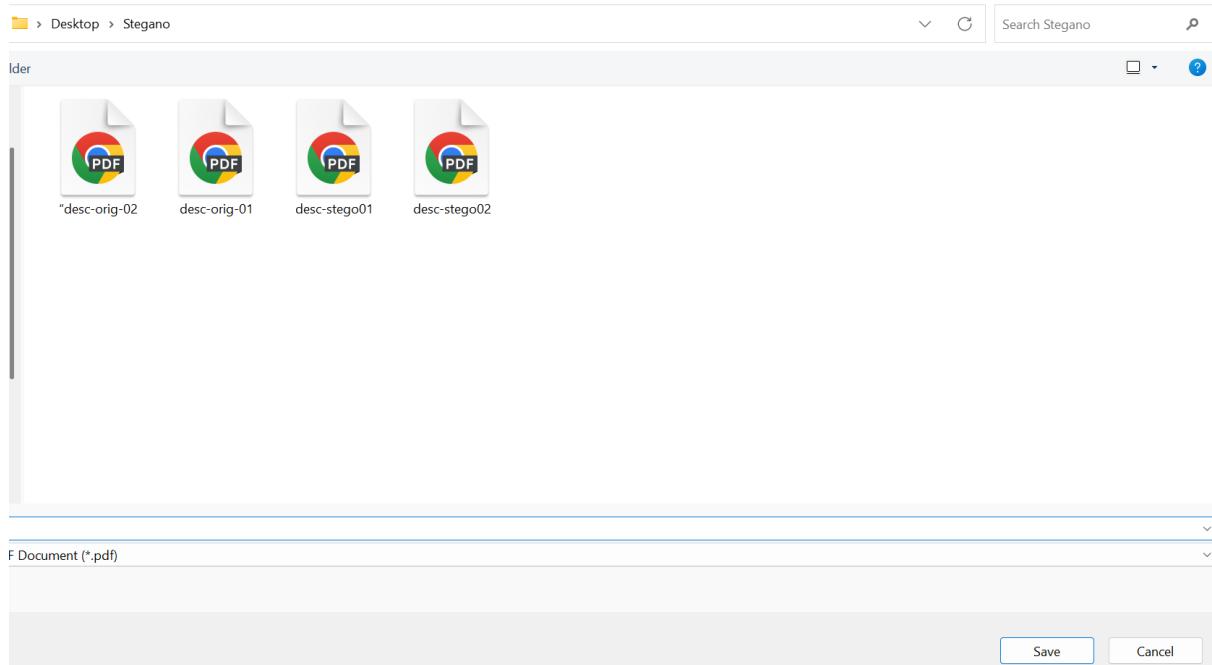
Stego





C. Right click the histogram and click “Print” and save the file as “desc-orig-01.pdf” and “desc-stego01.pdf” and02.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.