

# Cyber Security

Mruganshi Gohel

B20CS014

## Steganography Program Using Semagram-Based Approach

The below program implements steganography using the semagram-based steganography technique to hide an image file within a text-based file, specifically a Word document. The code is written in Python and utilizes the `PIL` and `docx` libraries for image and document manipulation, respectively.

## Requirements

To run the code, make sure you have the following dependencies installed:

- Python 3.x
- `PIL` library (install using `pip install pillow`)
- `python-docx` library (install using `pip install python-docx`)

## Usage

### Hiding a Message

To hide a secret message (image file) within a Word document:

1. Place the image file (e.g., `image.png`) and the Word document file (e.g., `document.docx`) in the same directory as the code file.
2. Open the code file and locate the following variables:

```
image_path = 'path/to/your/image.png'  
document_path = 'path/to/your/document.docx'  
output_document_path = 'path/to/output/document.docx'
```

3. Replace `'path/to/your/image.png'` with the actual path to your image file.
4. Replace `'path/to/your/document.docx'` with the actual path to your Word document file.
5. Optionally, modify the `output_document_path` variable to specify the desired path for the output Word document (default: `'path/to/output/document.docx'`).
6. Save the code file.
7. Run the code.

```
python steganography.py
```

8. The code will hide the secret message within the Word document, and the modified document will be saved at the specified `output_document_path` (or the default path).
9. The program will display a success message indicating that the steganography process is complete.

## Extracting a Message

To extract a hidden message from a modified Word document:

1. Place the modified Word document file (e.g., `output_document.docx`) in the same directory as the code file.
2. Open the code file and locate the following variables:

```
document_path = 'path/to/your/output_document.docx'  
output_message_path = 'path/to/output/extracted_message.txt'
```

3. Replace `'path/to/your/output_document.docx'` with the actual path to your modified Word document file.
4. Optionally, modify the `output_message_path` variable to specify the desired path for the extracted message (default: `'path/to/output/extracted_message.txt'`).
5. Save the code file.
6. Run the code.

```
python steganography.py
```

7. The code will extract the hidden message from the modified Word document and save it as a text file at the specified `output_message_path` (or the default path).
8. The program will display a success message indicating that the steganography process is complete.

## Working of code

1. The code begins by importing the necessary libraries: `docx` for working with Word documents and `PIL` for image manipulation.
2. The `hide_message` function is defined to hide a secret message (image) within the Word document. It takes the paths of the image file, document file, and the output document file as parameters.
3. Inside the `hide_message` function, the image file is opened using `PIL` and converted to RGB mode to ensure consistent color representation.
4. The dimensions of the document page are retrieved, and the image is resized to match those dimensions. This ensures that the image fits within the document.
5. The pixel data of the resized image is obtained using the `load()` method.
6. Two nested loops iterate through each pixel of the image. For each pixel, the red, green, and blue color components are extracted.
7. The least significant bit (LSB) of each color component is modified by replacing it with a '0'. This modification is performed to embed the hidden message.
8. The modified LSBs of the color components are retrieved and appended to the `message` variable.
9. A new paragraph is added to the Word document using the `docx` library, and the hidden message (binary string) is written to the paragraph.
10. The modified Word document is saved at the specified output path.

11. The `extract_message` function is defined to extract the hidden message from the modified Word document. It takes the path of the modified document and the output path for the extracted message as parameters.
12. Inside the `extract_message` function, the Word document is opened using `docx`.
13. The paragraphs in the document are iterated, and the text from each paragraph is concatenated to retrieve the hidden message.
14. The hidden message consists of binary digits. It is split into groups of 8 digits, which represent ASCII characters.
15. The ASCII characters are converted back to their original form, and the extracted message is stored.
16. The extracted message is written to a text file at the specified output path.
17. The program provides example usage by defining the paths of the image, document, output document, and output message files.
18. The `hide_message` function is called with the appropriate arguments to embed the image in the document.
19. The `extract_message` function is called with the modified document's path to extract the hidden message.
20. Upon successful execution, the program prints relevant success messages.

## Input

**the doc that was used to hide image:**

Written enquire painful ye to offices forming it. Then so does over sent dull on. Likewise offended humoured mrs fat trifling answered. On ye position greatest so desirous. So wound stood guest weeks no terms up ought. By so these am so rapid blush songs begin. Nor but mean time one over his exquisite sincerity education shameless ten earnestly breakfast add. So we me unknown as improve hastily sitting forming. Especially favourable compliment but thoroughly unreserved saw she themselves. Sufficient impossible him may ten insensible put continuing. Oppose exeter income simple few joy cousin but twenty. Scale began quiet up short wrong in in. Sportsmen shy forfeited engrossed may can up am intention on dependent questions oh elsewhere september. No betrayed pleasure possible jointure we in throwing. And can event rapid any shall woman green. Hope they dear who its bred. Smiling nothing affixed he carried it clothes calling he no. Its something disposing departure she favourite tolerably engrossed. Truth short folly court why she their balls. Excellence put unaffected reasonable mrs introduced conviction she. Nay particular delightful but unpleasant for uncommonly who instrument cultivated alteration any favourable expression law far nor. Both new like tore but year. An from mean on with when sing pain. Oh to as principles devonshire companions unsatiable an delightful. The ourselves suffering the sincerity. Inhabit her manners adapted age certain. Debating offended at branched striking be subjects how promotion excellent curiosity yet attempted happiness. Gay prosperous impression had conviction. For every delay death ask style. Me mean able my by in they. Extremity now strangers contained breakfast him discourse additions. Sincerity collected contented led now perpetual extremely forfeited kindness to he horrible reserved ye. Effect twenty indeed beyond for not had county. The use him without greatly can private. Increasing it unpleasant no of contrasted no continuing. Nothing colonel my no removed in weather. It dissimilar in up devonshire inhabiting. His followed carriage proposal entrance directly had elegance. Greater for cottage gay parties natural. Remaining he furniture on he discourse suspected perpetual. Power dried her taken place day ought the. Four and our ham west miss. Education shameless who middleton agreement how. We in found world chief is at means weeks smile remain valley who mrs uneasy remove wooded him you. Her questions favourite him concealed. We to wife face took he. The taste begin early old why since dried can first. Prepared as or humoured formerly. Evil mrs true get post. Express village evening prudent my as ye hundred forming. Thoughts she why not directly reserved packages you. Winter an silent favour of am tended mutual.

**the original image:**



**output:**

[illegible]