Secure Banking Transactions via Integrated Image Steganography for Confidential Information Exchange

Importing Libraries

In [1]:
```
import numpy as np
import cv2
import os
from pvd_lib import pvd_lib
from tkinter import filedialog, simpledialog
```

Declaring Global Variables

In [2]:
```
global image_path, text_path

pvd_obj = pvd_lib()
```

Slicing text of based on number of images

In [3]:
```
def slice_text(text_file, images_path):
    with open(text_file, "rb") as myfile:
        data = myfile.read()

    files = []
    for root, _, directory in os.walk(images_path):
        for file in directory:
            if 'Thumbs.db' not in file:
                files.append(os.path.join(root, file))

    length = len(data)
    size = length // len(files)
    tot_blocks = len(files)

    block = []
    start, end = 0, size
    for i in range(tot_blocks):
        chunk = data[start:end]
        block.append(chunk.decode('latin1'))
        start = end
        end += size

    remain = length - start
    if remain > 0:
        chunk = data[start:length]
        block[-1] += chunk.decode('latin1')

    return block, files
```

Encoding of text in the Images

In [4]:
```
def pvd_encoding(image_path, slice_msg, pvd_obj):
    base_name = os.path.basename(os.path.dirname(image_path))
    output_dir = os.path.join('/;;;;;;p;p;ppp', base_name)
    os.makedirs(output_dir, exist_ok=True)

    img_name = os.path.basename(image_path).replace(".jpg", ".png")
    temp_text_path = "data.txt"

    with open(temp_text_path, "wb") as myfile:
        myfile.write(slice_msg.encode())

    pvd_obj.pvd_embed(image_path, temp_text_path, os.path.join(output_dir, img_name))
```

Decoding of text from the Images

In [5]:
```
def pvd_decoding(encoded_path, pvd_obj):
    output = ""
    files = [os.path.join(root, file) for root, _, directory in os.walk(encoded_path) for file in directory if 'Thumbs.db' not in file]

    for file in files:
        temp_text_path = "data.txt"
        pvd_obj.pvd_extract(file, temp_text_path, file)

        with open(temp_text_path, "rb") as myfile:
            data = myfile.read()
        output += data.decode('latin1')

    return output
```

In [6]:
```
pvd_obj = pvd_lib()
```

Declaring input paths

In [7]:
```
text_file = r"sample_text.txt"
image_dir = r"images1"
```

In [8]:
```
blocks, image_files = slice_text(text_file, image_dir)
for i in range(len(image_files)):
    img = cv2.imread(image_files[i])
    img = cv2.resize(img, (600, 600))
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    cv2.imwrite(image_files[i], img)
    pvd_encoding(image_files[i], blocks[i], pvd_obj)
```

Text Extraction from the encoded images

In [9]:
```
encoded_images_dir = r"Encoded_Images\images1"
extracted_text = pvd_decoding(encoded_images_dir, pvd_obj)
print("Extracted Text:", extracted_text)
```

```
Extracted Text: Namani Vamshi Krishna
SBI Bank: Huzurabad Karimnagar
Account Number: 32664599680
CARD Number: 7846 5162 1733
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: