



Information Security

Audio, Video and Image Steganography

Dawood Sarfraz

20P-0153

BSCS-7B



STEGANOGRAPHY:

Steganography is basically referred as hidden writing. It is entirely different from cryptography. In cryptography ,we basically perform the encryption and decryption by changing the text into another form.



Carrier Medium:

- a. This could be an image, audio file, video, or any other form of digital data.

Hidden Information:

- a. It could be text, an image, a file, or any other information.

Embedding:

- a. In an image, changes might be made to the least significant bits of the pixel values.

Stego Medium:

- a. The resulting file or message, which now contains both the original content and the hidden information, is called the stego medium.

Extraction:

- a. The recipient can extract the hidden information from the stego medium.



Types of Steganography:

1. Image Steganography
2. Text Steganography
3. Video Steganography
4. Audio Steganography

RSA Algorithm:

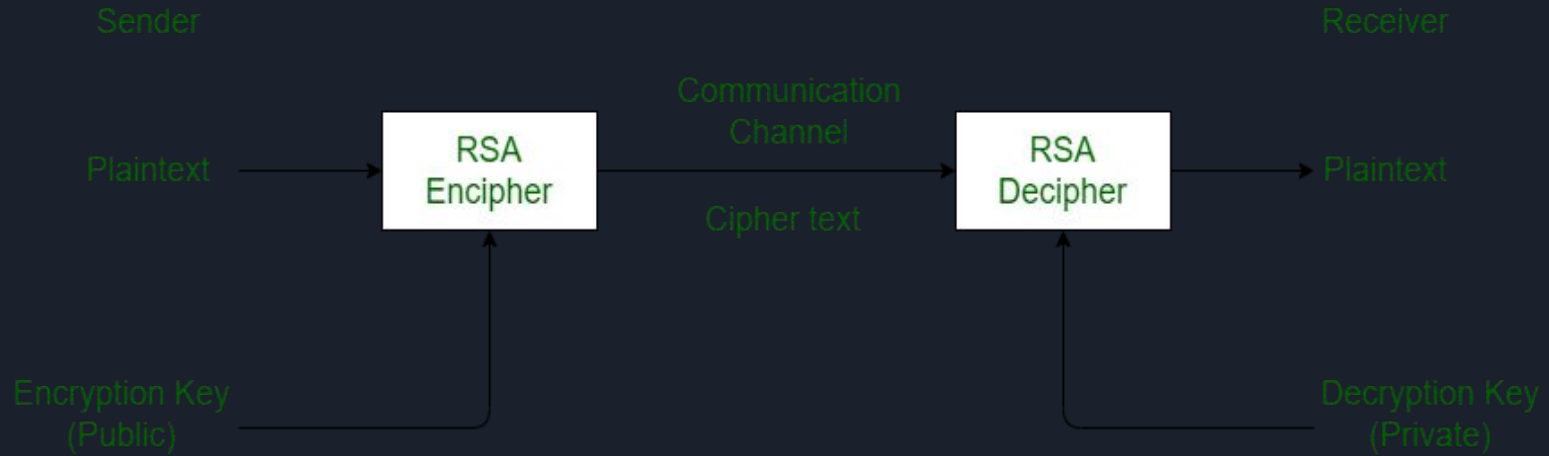




IMAGE STEGANOGRAPHY WITH RSA ALGORITHM

How the image steganography works:

Before encrypting file into image file, we use the well defined RSA Algorithm to increase the security of the data transmission.

Then, we use this encrypted message to be encrypted once again into the image file.

Every image is formed of the pixels. There are a number of pixels in the image. Every pixel is formed of 24 bits.

Each pixel has 8 bits of red color, 8 bits of green color and 8 bits of blue color(RGB).

24 bit image - 16.7 million colors

Red

0	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

Green


0	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

Blue

0	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---



1 Pixel

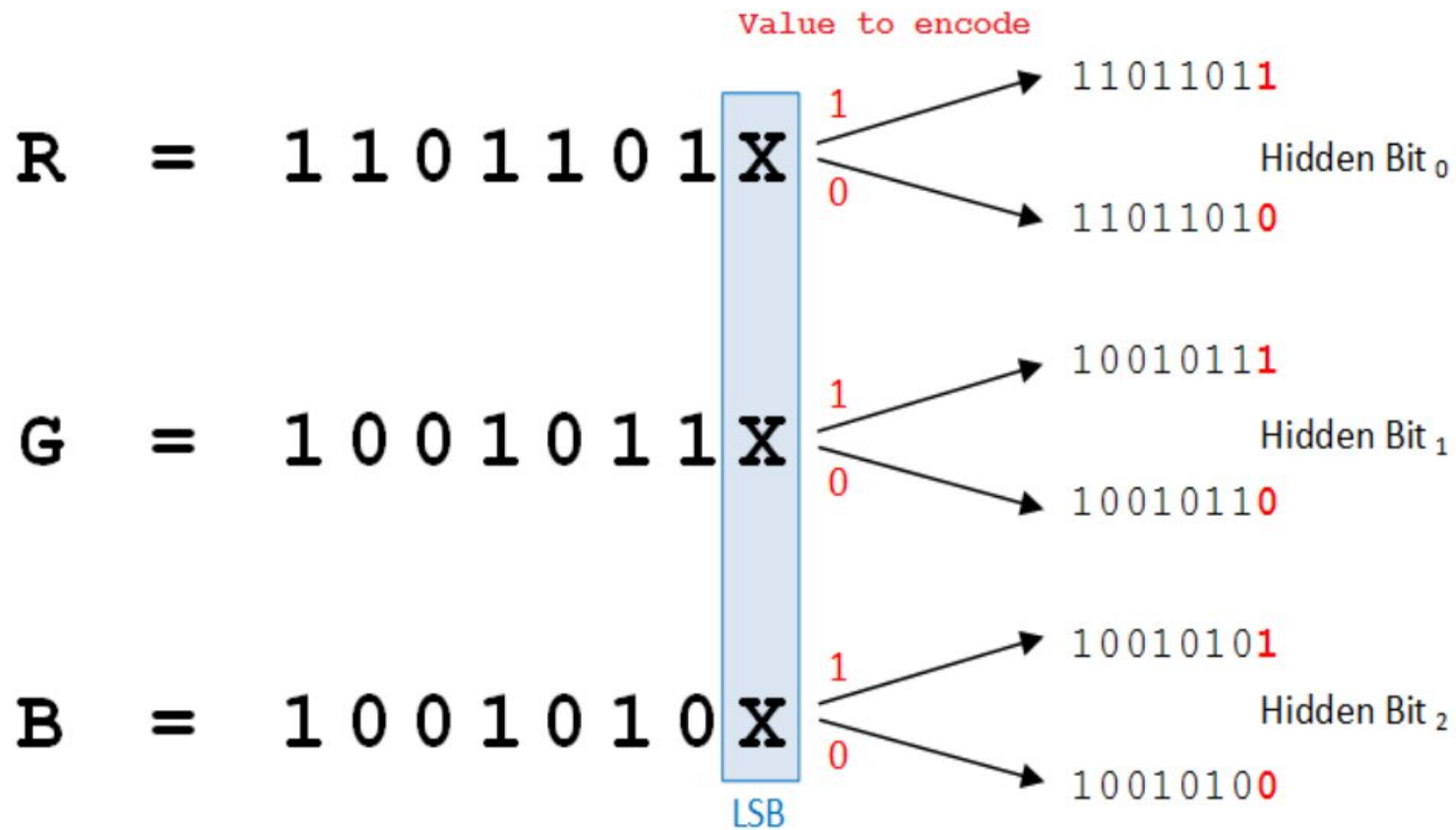



Also the file that we want to send to the receiver is also formed from a number of bits.

Now the next step is to store every 3 bits of the file in every pixel's Least Significant bit (LSB) i.e. 1-1 bit in 8th bit of red, blue, green color bits.

In this way our encryption will be done.

In this way, decryption can also be performed by extracting the 8th bit of every color from each pixel.





In **Encryption** we start a loop and in loop we access every byte of the file 3 times .

We also use the functions 'byte-to-bool' and 'bool-to-byte' which converts byte into bits.

Starting from LSB, in the first iteration we first put 3 bits of the file in the red pixel, then in the green one and in third i.e. blue we put 2 bits.

The chromatic influence of blue color is maximum for human eye so we change only 2 bits of it.

When all the bytes of the file are not able to store in the last bits of every color then we start to replace the 7th bit of every color.

In this way the storing capacity of the image file increases but little noise is also increased in the image

In the same way we can perform the **Decryption**



Audio Steganography:

Audio Steganography is the scheme of hiding the existence of secret information by concealing it into another medium such as audio file.

Audio Steganography is a technique used to transmit hidden information by modifying an audio signal in an imperceptible manner. It is the science of hiding some secret text or audio information in a host message. The host message before Steganography and Stego message after Steganography have the same characteristics.

Embedding secret messages in digital sound is a more difficult process.

Varieties of techniques for embedding information in digital audio have been established.



METHODS OF AUDIO DATA HIDING:

1. Least Significant Bit
2. Spread Spectrum
3. Discrete Wavelets Transform
4. Echo Data Hiding



LEAST SIGNIFICANT BIT

Least significant bit (LSB) coding is the simplest way to embed information in a digital audio file.

By substituting the least significant bit of each sampling point with a binary message.

In implementations of LSB coding, the least significant bits of a sample are replaced with message bits.

To extract a secret message from an LSB encoded sound file, the receiver needs access to the sequence of sample indices used in the embedding process.



SPREAD SPECTRUM TECHNIQUE:

Spread Spectrum steganography on audio data will be implemented with the following scheme:

Transform the audio cover object in time-domain into frequency-domain using Fast Fourier Transform.

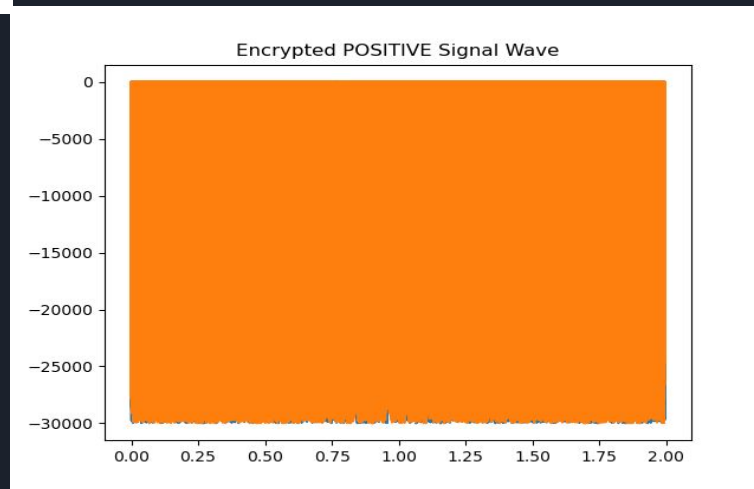
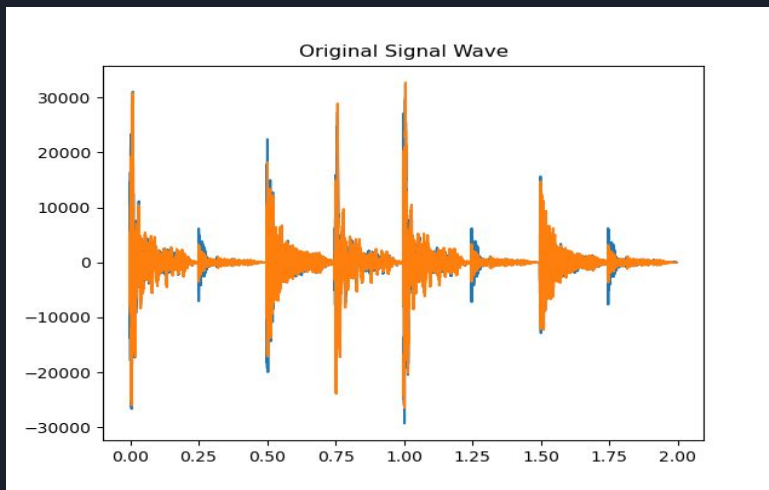
Adding the information signal by using spread-spectrum to the cover object in frequency-domain.

Transform back the audio cover object from frequency –domain into time –domain using inverse fast Fourier Transform



DISCRETE WAVELETS TRANSFORM:

Wavelet transform is used to convert a spatial domain into frequency domain. The use of wavelet in audio stenographic model lies in the fact that the wavelet transform clearly separates the high frequency and low frequency information.





Echo Data Hiding:

One common application of echo data hiding is in digital audio watermarking, where additional information is embedded in audio signals without significantly affecting the perceived quality of the audio. This technique is also used in other types of media, such as images and videos.

It's important to note that the success of echo data hiding depends on various factors, including the sensitivity of the human senses to changes in the carrier signal and the robustness of the method against various types of analysis and attacks.



Video Steganography:

Video steganography involves hiding information within a video file. This can be done using various techniques to embed data in a way that is not easily perceptible. Here's a basic overview of how video steganography might be implemented:



By using LSB(Least Significant Bit algorithm):

The most common and popular method of modern day steganography is to make use of LSB of picture's pixel information.

This technique works best when the file is longer than the message file and if image is grayscale.

When applying LSB techniques to each byte of a 24 bit image, three bits can be encoded into each pixel.



Data Embedding Algorithm:

Step 1: Extract the pixels of the cover image.

Step 2: Extract the characters of the text le.

Step 3: Extract the characters from the Stego key.

Step 4: Choose first pixel and pick characters of the Stego key and place it in first component of pixel.

Step 5: Place some terminating symbol to indicate end of the key. 0 has been used as a

terminating symbol in this algorithm.

Step 6: Insert characters of text le in each rst Component of next pixels by replacing it.

Step 7: Repeat step 6 till all the characters has been embedded.



Data Extraction Algorithm:

Step 1: Extract the pixels of the stego image.

Step 2: Now, start from first pixel and extract stego key characters from first component of the pixels.

Step3: up to terminating symbol, otherwise follow step 4.

Step 4: If this extracted key matches with the key entered by the receiver, then follow Step 5, otherwise terminate the program

Step 5: If the key is correct, then go to next pixels and extract secret message characters from first component of next pixels. Follow Step 5 till up to terminating symbol, otherwise follow step 6.

Step 6: Extract secret message



Thank You