INFORMATION SECURITY PROJECT

IMAGE ENCRYPTION USING SHUFFLING And AES

Submitted by- Vaibhav Sharma	(15103311)B8
Jitesh Pabla	(15103297)B8
Sajal Subodh	(15103332)B8

Introduction

Visual cryptography is a cryptographic technique which allows visual information (pictures, text, etc.) to be encrypted in such a way that decryption becomes the job of the person to decrypt via sight reading.

One of the best-known techniques has been credited to Moni Naor and Adi Shamir, who developed it in 1994.They demonstrated a visual secret sharing scheme, where an image was broken up into n shares so that only someone with all n shares could decrypt the image, while any n − 1 shares revealed no information about the original image. Each share was printed on a separate transparency, and decryption was performed by overlaying the shares. When all n shares were overlaid, the original image would appear. There are several generalizations of the basic scheme including k-out-of-n visual cryptography.

APPLICATION

The image encryption is to transmit the image securely over the network so that no unauthorized user can able to decrypt the image. Image encryption, video encryption, chaos based encryption have applications in many fields including the internet communication, transmission, medical imaging. Tele-medicine and military Communication, etc. The evolution of encryption is moving towards a future of endless possibilities. The image data have special properties such as bulk capability, high redundancy and high correlation among the pixels. Encryption techniques are very useful tools to protect secret information. Encryption will be defined as the conversion of plain message into a form called a cipher text that cannot be read by any people without decrypting the encrypted text. Decryption is the reverse process of encryption which is the process of converting the encrypted text into its original plain text, so that it can be read.

Technique Used

In this project, we are using 2-level encryption for an image for more secure sharing of images. First, we are encrypting this image by converting this image to string format and applying AES and after that we are using modified version (2,n)Share Encryption for better results. For decyption , we are using Secret image generated during encryption.

AES Encryption :

AES is based on a design principle known as a substitution-permutation network, a combination of both substitution and permutation, and is fast in both software and hardware. AES is a

variant of Rijndael which has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. By contrast, the Rijndael specification per se is specified with block and key sizes that may be any multiple of 32 bits, with a minimum of 128 and a maximum of 256 bits.

AES operates on a 4 × 4 column-major order matrix of bytes, termed the state, although some versions of Rijndael have a larger block size and have additional columns in the state. The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the ciphertext.

Each round consists of several processing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

For implenting AES we are using python library Crypto.Cipher.

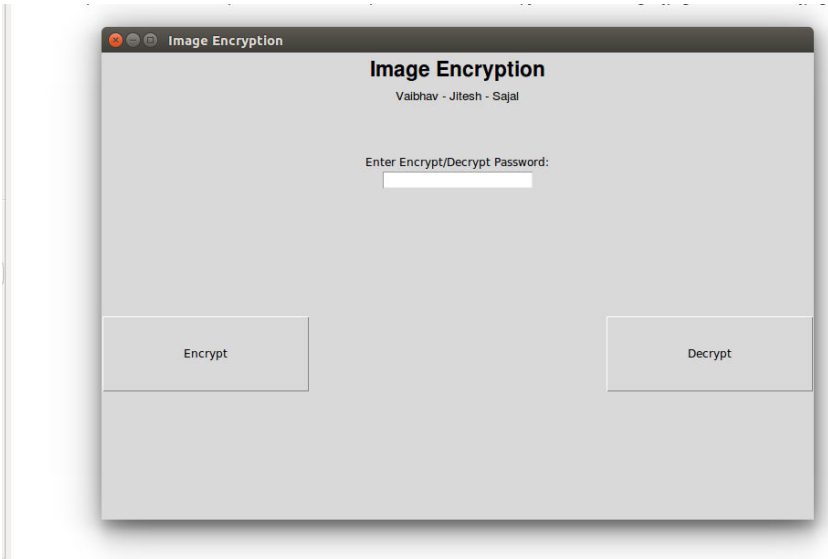
(2,n)Share Encryption:

In this scheme we have a secret image which is encoded into N shares printed on transparencies. The shares appear random and contain no decipherable information about the underlying secret image, however if any 2 of the shares are stacked on top of one another the secret image becomes decipherable by the human eye.

Every pixel from the secret image is encoded into multiple subpixels in each share image using a matrix to determine the color of the pixels. In the (2,N) case a white pixel in the secret image is encoded using a matrix from the following set, where each row gives the subpixel pattern for one of the components.

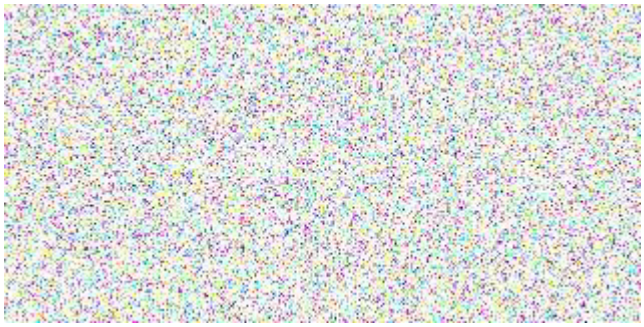
Modification -: As this algo uses 1bit pixel image and we are working on rgb(3X8 byte pixel) images, So we are shuffling it with 0 to 255 bits for each layer and generating secret image and after that we are encrypting our message image by using (mssg[i]+seret[i])%256 and decrypting using (cipher[i]-secret[i])%256.

GUI:



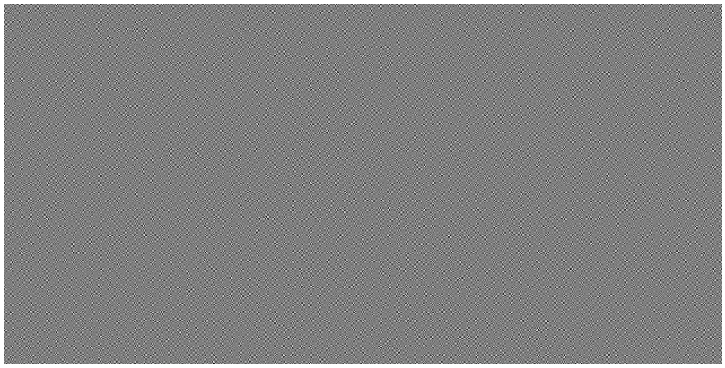
TESTCASE 1 : -

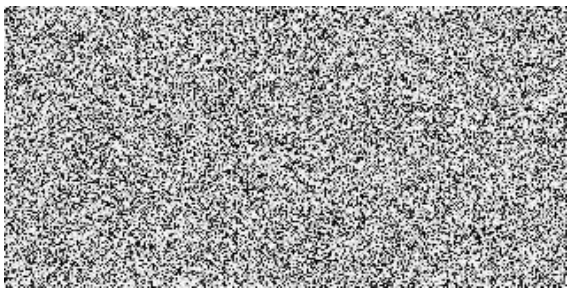
pass - vaibhav



1) original

2)AES Encrypt





3) (2,n) share encrypt

4) (2,n) share decrypt

5) AES Decrypt

Platform Used

1. Python 2.7
2. Linux

References

1. https://en.wikipedia.org/wiki/Visual_cryptography
2. A Survey On Different Image Encryption and Decryption Techniques. Rinki Pakshwar, Vijay Kumar Trivedi, Vineet Richhariya
    a. http://ijcsit.com/docs/Volume%204/Vol4Issue1/ijcsit2013040126.pdf