SECURE DATA ENCRYPTION AND DECRYPTION USING CRYPTO-STEGO

A PROJECT REPORT

Submitted by

Kartik Jindal (RA2011030010143)
Arohi sood (RA2011030010122)

ABSTRACT

Securing data encryption and decryption using Cryptography and Steganography techniques. Due to recent developments in stego analysis, providing security to personal contents, messages, or digital images using steganography has become difficult. By using stego analysis, one can easily reveal existence of hidden information in carrier files. This project introduces a novel steganographic approach for communication between two private parties. The approach introduced in this project makes use of both steganographic as well as cryptographic techniques. In Cryptography we are using RSA. In Steganography we are using Image Steganography for hiding the data. And we also use Mutual Authentication process to satisfy all services in Cryptography i.e., Access Control, Confidentiality, Integrity, Authentication. In this way we can maintain the data more securely. Since we use RSA algorithm for securing the data and again on this we perform Steganography to hide the data in an image. Such that any other person in the network cannot access the data present in the network. Only the sender and receiver can retrieve the message from the data.

Keywords : Rivest-Shamir-Adelman(RSA), Crptography, Steganography

TABLE OF CONTENTS

1. INTRODUCTION

1.1 INTRODUCTION

Digital communication witnesses a noticeable and continuous development in many applications in the Internet. Hence, secure communication sessions must be provided. The security of data transmitted across a global network has turned into a key factor on the network performance measures. So, the confidentiality and the integrity of data are needed to prevent eavesdroppers from accessing and using transmitted data. Steganography and Cryptography are two important techniques that are used to provide network security.

The aim of this project is to develop a new approach to hiding a secret information in an image, by taking advantage of benefits of combining cryptography and steganography.

1.1.1 Cryptography

Cryptography is one of the traditional methods used to guarantee the privacy of communication between parties. This method is the art of secret writing, which is used to encrypt the plaintext with a key into ciphertext to be transferred between parties on an insecure channel. Using a valid key, the ciphertext can be decrypted to the original plaintext. Without the knowledge of the key, nobody can retrieve the plaintext. Cryptography plays an essential role in many factors required for secure communication across an insecure channel, like confidentiality, privacy, non-repudiation, key exchange, and authentication.

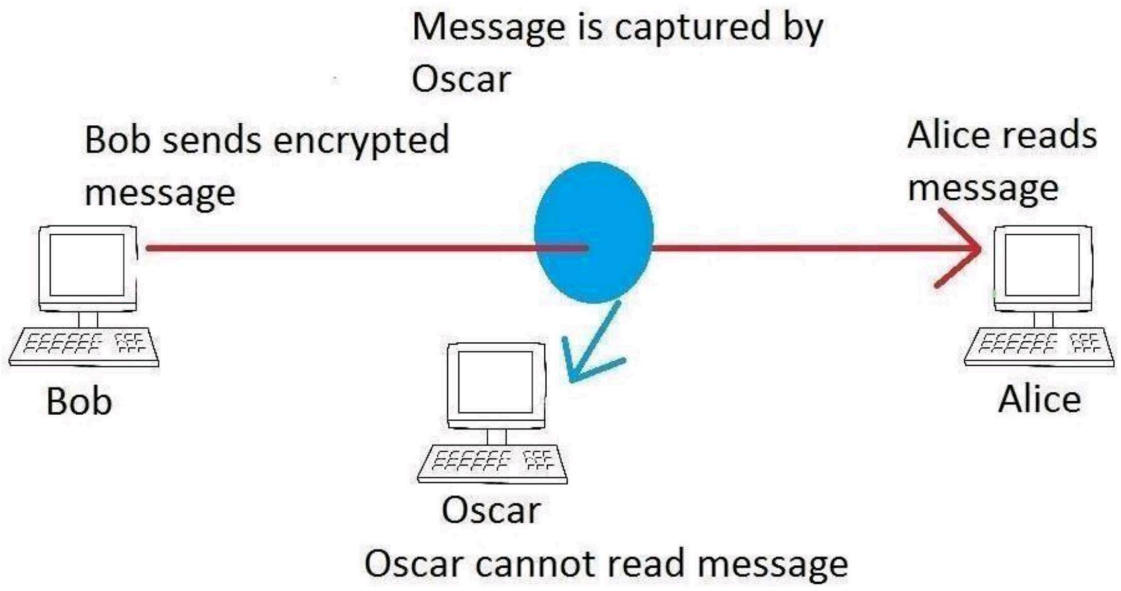


Fig 1.1.1 : Cryptography as a flow model.

1.1.1.1 Symmetric / Secret Key Cryptography

The technique of Secret key encryption can also be known as the symmetric-key, shared key, single-key, and eventually private-key encryption. The technique of private key uses for all sides encryption and decryption of secret data. The original information or plaintext is encrypted with a key by the sender side also the similarly key is used by the receiver to decrypt a message to obtain the plaintext. the key will be known only by a people who are authorized to the encryption/decryption. However, the technique affords the good security for transmission but there is a difficulty with the distribution of the key. If one stole or explore the key he can get whole data without any difficulty. An example of Symmetric-Key is DES Algorithm.

1.1.1.2 Asymmetric / Public Key Cryptography

We can call this technique as asymmetric cryptosystem or public key cryptosystem, this technique use two keys which are mathematically associated, use separately for encrypting and decrypting the information. In this technique, when we use the private key, there are no possibilities to obtain the data or simply discover the other key. The key used for encryption is stored public therefore it's called public key, and the decryption key is stored secret and called private key. An example of Asymmetric-Key Algorithm is RSA.

1.1.2 Steganography

It can be defined as the science of hiding and communicating data through apparently reliable carriers in attempt to hide the existence of the data. So, there is no knowledge of the existence of the message in the first place. If a person views the cover which the information is hidden inside, he or she will have no clue that there is any covering data, in this way the individual won't endeavour to decode the data. The secret information can be inserted into the cover media by the stego system encoder with using certain algorithm. A secret message can be plaintext, an image, ciphertext, or anything which can be represented in form of a bitstream. after the secret data is embedded in the cover object, the cover object will be called as a stego object also the stego object sends to the receiver by selecting the suitable channel, where decoder system is used with the same stego method for obtaining original information as the sender would like to transfer .

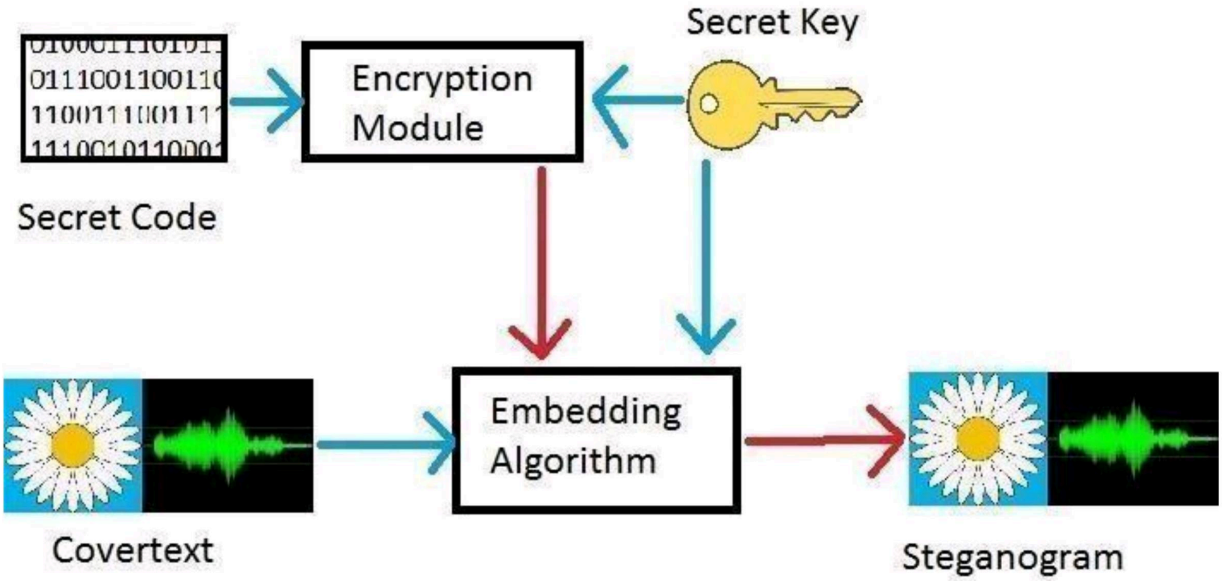


Fig 1.1.2 : Stegonography as a flow model.

1.1.3 Types of Steganography

There are various types of steganography.

A. Text Files

The technique of embedding secret data inside a text is identified as text stego. Text steganography needs a low memory because this type of file can only store text files. It affords fast transfer or communication of files from a sender to receiver.

B. Image Files

It is the procedure in which we embed the information inside the pixels of image. So, that the attackers cannot observe any change in the cover image. LSB approach is a common image steganography algorithm.

C. Audio Files

It is the process in which we hide the information inside an audio. There are many approaches to hide secret information in an audio files for examples Phase Coding, LSB .

D. Video Files

It is the process of hiding some secret data inside the frames of a video.

1.1.4 Steganography versus Cryptography

Steganography and cryptography are used for the purpose of data transmission over an insecure network without the data being exposed to any unauthorized persons. Steganography embeds the data in a cover image while cryptography encrypts the data. The advantage of Steganography is that, the look of the file isn't changed and this it will not raise any doubt for the attacker to suspect that there may be some data hidden unlike cryptography that encrypts the data and sends it to network.

1.1.5 Benefits of Steganography and Cryptography

It is noted that steganography and cryptography alone is insufficient for the security of information, therefore if we combine these systems, we can generate more reliable and strong approach. The combination of these two strategies will improve the security of the information. This combined will fulfill the prerequisites, for example, memory space, security, and strength for important information transmission across an open channel. Also, it will be a powerful mechanism which enables people to communicate without interferes of

eavesdroppers even knowing there is a style of communication in the first place.

1.1.6 Applications of Steganography

(i) **Secret Communication :** Steganography does not advertise secret communication and therefore avoids scrutiny of the sender message. A trade secret, blueprint, or other sensitive information can be transmitted without alerting potential attackers.

(ii) **Feature Tagging :** Elements can be embedded inside an image, such as the names of the individuals in a photo or location in a map. Copying the stego image also copies all of the embedded features and only parties who possess the decode stego key will be able to extract and view the features.

(iii) **Copyright Protection :** Copy protection mechanisms that prevent the data, usually digital data from being copied. The insertion and analysis of water marks to protect copyrighted material is responsible for the percent rise of interest digital steganography and data embedding.

1.2 MOTIVATION FOR THE WORK

Motivation is very important function for any project. It is one of the methods to induce the man on the job to get the work done effectively to have the best results towards the common objectives. It is necessary for the better performance.

Motivation can be seen as the inner drive, which prompts people to act in a way either towards achieving their personal goals or organizational goals. To a large extent, motivation is “leadership” as it involves getting the whole staff to learn to work willingly

and well in the interest of the business. A leader can influence his subordinate only when they are convinced.

Conviction can only come when the entire subordinate accepts those factors that propel actions of individuals, which are referred to as motivation. They may be highly paid, prestigious titles promotion, praises, bonus, etc. The word is an abstract noun applying to the entire class of desired need wishes and similar forces. Motivation has to do with action which results, to satisfaction closely associated with motivation is the word "miracle" it is injecting of moral and loyalty into the working team so that they will carry their duties properly and effectively with maximum economy.

The main reason and motivation for choosing this project is, Due to recent developments in stego analysis, providing security to personal contents, messages, or digital images using steganography has become difficult. By using stego analysis, one can easily reveal existence of hidden information in carrier files. So, after been exposed to such problems it motivated us to do this project where the complete process of transferring of information is done using two different techniques. All that is required is to select a cover image and transfer the information using that image.

1.3 PROBLEM STATEMENT

The purpose of this project is to provide the correct data with security to the users. For some of the users the data might be lost during the transmission process in the network and for some, the data might be changed by the unauthorized person in the network and there are some other security problems in the network. Our application will give you more Security to the data present in the network and there will be able to reduce the loss of data in the network which will be transmitted from the sender to the receiver using the latest technologies. Only the Authorized persons i.e., who are using our application will be

there in the Network. The proposed algorithm is to hide the audio data effectively in an image without any suspicion of the data being hidden in the image. It is to work against the attacks by using a distinct new image that isn't possible to compare.

The aim of the project is to hide the data in an image using steganography and ensure that the quality of concealing data must not be lost.

We used a method for hiding the data in a distinct image file in order to securely send over the network without any suspicion the data being hidden. This algorithm, though requires a distinct image which we can use as a carrier and hide the data which is well within the limits of the threshold that the image can hide, that will secure the data.

2. METHODOLOGY

3.1 SYSTEM ARCHITECTURE

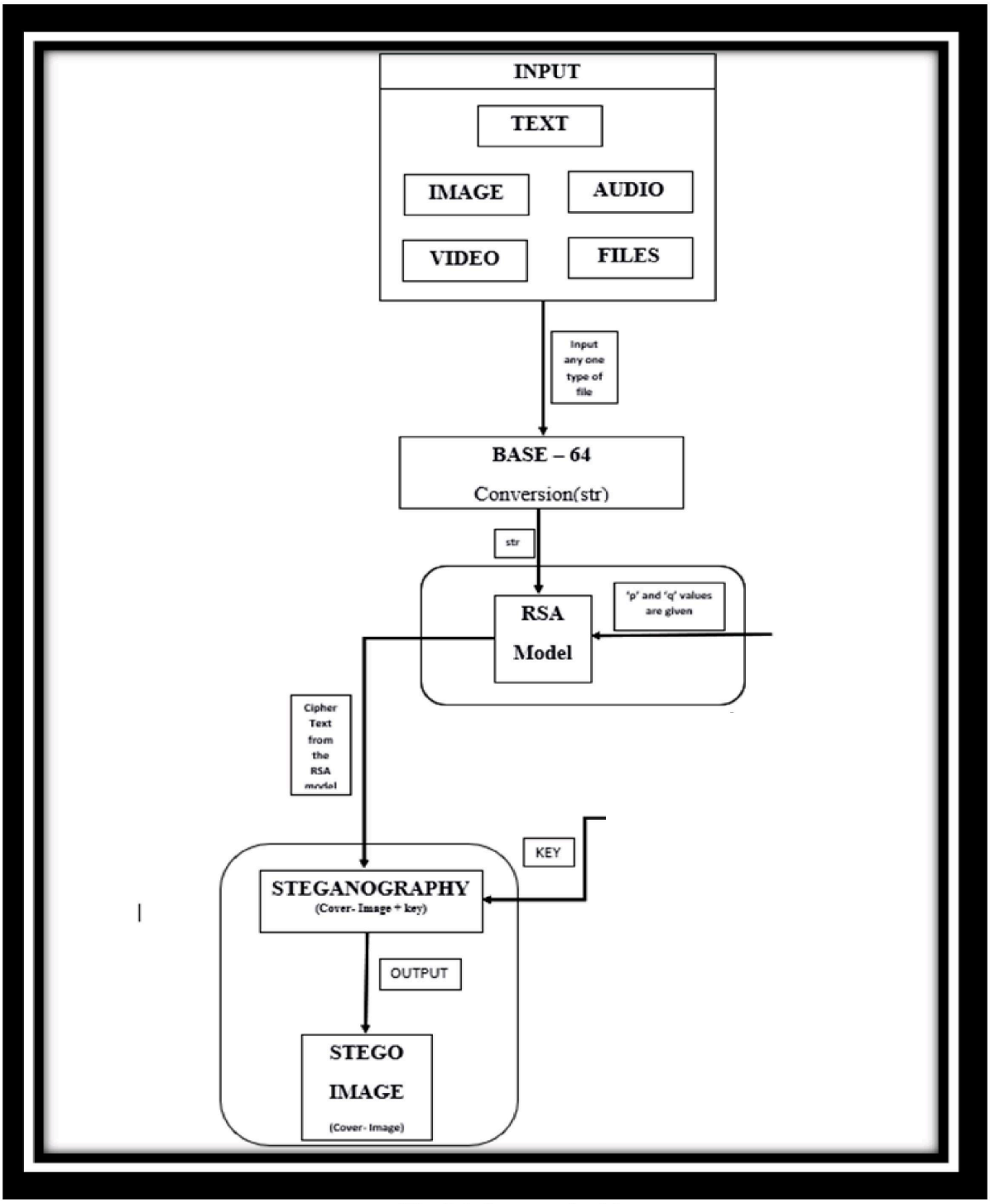
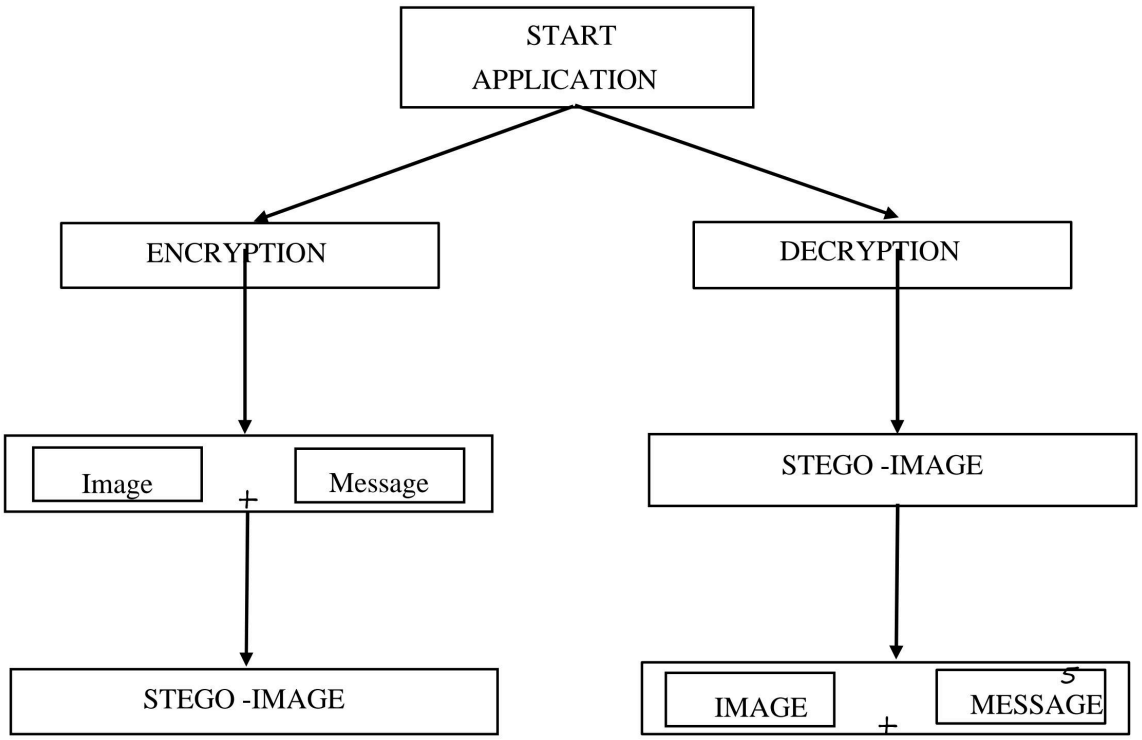


Fig 3.1 System Architecture

3.2 Proposed System

In this section, we will discuss proposed method which combines two different hiding techniques, which are Cryptography and Steganography. In this proposed method first, the message is encrypted by use RSA algorithm. After that, we use the modified LSB technique to embed the encrypted information in image. So, this technique combines the features of both cryptography and steganography and provides a high level of security. It is better than either of the technique used separately. There will be an agreement between the sender and the receiver about the key for the concealment algorithm as well as the key for the encryption algorithm or these keys may be exchanged by a secure communication method. Our method starts by encryption first then hide encrypted data.



Before applying the cryptography and steganography, initially we convert our input to Base-64. And we save the obtained text in a text file. Then we proceed to cryptography and steganography.

3.2.1 Sender Side

The Sender side consists of cryptographic and steganography stages. This method starts with cryptographic then steganography.

Cryptography Stage :

In encryption stage, we use RSA (Rivest Shamir Adelson) algorithm. This technique takes two prime numbers. The Encryption can be done using the Plain Text and with "e" values which was generated using the two prime numbers. Then we will get a cipher text, which is communicated to the receiving end for decryption. This encrypted data will be used in steganography stage.

Input= Message + Two Prime Numbers.

Output= Encrypted Message.

Steganography Stage :

In stenography stage, we use LSB (Least Significant Bit) algorithm with some modification to hide information (encrypted data from cryptography stage) inside a cover. In our experiment, we use the image as cover to present our method, but this method can be applied to other files such as audio, and video. The general LSB method used to hide secret information into a file; the last bit in each pixel or sample or frame used sequentially to hide one of the binary stream bits Encryption of the cover image.

Input= Encrypted Message + Secret key+ cover image.

Output= Stego-Image.

3.2.2 Receiver side

Receiver side consists of steganography and cryptography stages. In receiver side we will first extract embedded data then decrypt it.

Steganography Stage :

In the receiver side, we start with steganography then cryptography. We will use the same steps which are used in sender side.

Input= Stego-Image+ Secret Key.

Output= Encrypted Message.

Cryptography Stage :

In cryptography stage, we use the data which is extracted from stego file and use RSA. We will use the same steps which are used in sender side. The Decryption can be done using the Encrypted message, receivers private key and senders public key.

Input= Encrypted Message + 2 Prime Numbers.

Output= Plain Text.

Now the Plain Text is in the form of Base-64. After getting the plain text apply Base-64 coversion to change the Plain-text to given input, which can be Text, Image, Video, Audio.

3.2.3 RSA

The RSA algorithm is the basis of a cryptosystem a suite of cryptographic algorithms that are used for specific security services which enables public key encryption and is widely used to secure sensitive data, particularly when it is being sent over an insecure network

3.2.3.1 Why RSA Algorithm is used ?

The public and private key generation algorithm is the most complex part of RSA cryptography. Two large prime numbers, p and q, are selected. N is calculated by multiplying p and q. This number is used by both the public and private keys and provides the link between them. Its length, usually expressed in bits, is called the key length.

The public key consists of the modulus n and a public exponent e. The e doesn't have to be a secretly selected prime number, as the public key is shared with everyone.

such as the internet. RSA was first publicly described in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman of the Massachusetts Institute of Technology, though the 1973 creation of a public key algorithm by British mathematician Clifford Cocks was kept classified by the U.K.'s GCHQ until 1997. In RSA cryptography, both the public and the private keys can encrypt a message; the opposite key from the one used to encrypt a message is used to decrypt it. This attribute is one reason why RSA has become the most widely used asymmetric algorithm, It provides a method to assure the confidentiality, integrity, authenticity, and non-repudiation of electronic communications and data storage.

3.2.3.1 RSA Security

RSA security relies on the computational difficulty of factoring large integers. As computing power increases and more efficient factoring algorithms are discovered, the ability to factor larger and larger numbers also increases. Encryption strength is directly to key size, and doubling key length can deliver an exponential increase in strength, although it does impair performance. RSA keys are typically 1024-bits or 2048-bits long, but experts believe that 1024-bit keys are no longer fully secure against all attacks. This is why the government and some industries are moving to a minimum key length of 2048-bits.

Barring an unforeseen breakthrough in quantum computing, it will be many years before longer keys are required, but elliptic curve cryptography (ECC) is gaining with many security experts as an alternative to RSA to implement public key cryptography. It can create faster, smaller and more efficient cryptographic keys.

Modern hardware and software are ECC-ready, and its popularity is likely to grow, as it can deliver equivalent security with lower computing power and battery resource usage, making it more suitable for mobile apps than RSA. Finally, a team of researchers, which included Adi Shamir, a co-inventor of RSA, has successfully created a 4096-bit RSA key using acoustic cryptanalysis; however, any encryption algorithm is vulnerable to attack.

3.2.3.2 Description of Algorithm

- Plaintext is taken from a specified file and then encrypted using RSA Algorithm.
- Encryption and decryption are of following form for same plaintext M and ciphertext C.
- C=(M^e)modn
- M=(C^d)modn
- M=((M^e)^d)modn
- M=(M^ed)modn
- Both sender and receiver must know the value of n.
- The sender knows the value of e, and the receiver knows the value of d.
- Thus this is a public key encryption algorithm with a public key of PU = {c, n} and private key of PR= {d, n}.

3.3.2.3 RSA algorithm

a) Key Generation :

- Select p and q such that both are the prime numbers, p≠q.
- Calculate n=p×q
- Calculate q(n) = (p-1) (q-1)
- Select an integer e such that : g(d ((n), e)) =1 & 1< e < (n)
- Calculate d; de = 1 mod (q(n))
- Public Key, PU= {e, n}
- Private Key, PR ={d,n}

b) Encryption :

- Plaintext : M
- Ciphertext: C = (M^e) mod n

c) Decryption:

- Ciphertext: C
- Plaintext : M= (C^d) mod n
- Note 1 : (n) -> Euler's totient function
- Note 2: Relationship between C and d is expressed as:

  ed (mod (n)) =1

  ed = 1 mod (n)

  d =e−1 mod (n)

3.2.4 STEGANOGRAPHY

Data hiding is of important in many applications. For hobbyists, secretive data transmission, privacy of users etc. the basic methods are: Steganography and Cryptography. Steganography is a simple security method. Generally there are three different methods used for hiding information: steganography, cryptography, watermarking. In cryptography, the information to be hidden is encoded using certain techniques; this information is generally understood to be coded as the data appears nonsensical. Steganography is hiding information; this generally cannot be identified because the coded information doesn't appear to be abnormal i.e. its presence is undetectable by sight. Detection of steganography is called Stego analysis.

Steganography is of 4 different types:

- Text steganography
- Image steganography
- Audio steganography
- Video steganography

In all of these methods, the basic principle of steganography is that a secret message is to be embedded in another cover object. So it cannot be detected easily to be containing hidden information unless proper decryption is used.

Every steganography consists of three components:

- Cover object
- Message object
- Resulting Steganographic object

3.3 ALGORITHM ILLUSTARTION

Encryption :

Inputs : Message, 2 Prime Numbers, Image, Secret Key

Step 1 : Consider an Input , It can be :

a) Text

b) age

c) Audio

d) Video

Step 2 : Convert the input to Base-64 using Base-64 conversion Algorithm.

Step 3 : After converting into Base-64 we will be getting a String.

Step 4 : Store the entire string in a Text File and save the file.

Step 5 : From that file consider each character and apply RSA.

Step 6 : By Using RSA we will be getting Cipher Text (cm).

Step 7 : Let the Cipher Text (cm) be encrypted message.

Step 8 : Consider an image, And hide the encrypted message(cm) in the given image with the secret key Using Stegnography Algorithm.

Step 9: Now send the Stego-Image to the Receiver.

DECRYPTION :

Inputs : Cipher Text, 2 Prime Numbers, Image, Secret Key

Step 1 : Consider the input be Stego-Image.

Step 2 : Using the Secrert Key , Obtain the hidden message from the Stego-Image.

Step 4 : And the obtained message is a Cipher Text. We must decrypt the message.

Step 5 : The Decyption of the message can be done using RSA Algorithm.

Step 6: By Using RSA we will be getting Plain Text.

Step 7 : And thus the receiver will decrypt the message and it is in the form of Base-64.

Step 8 : Finally by using Base-64 algorithm the Base-64 text is converted into the original input, Which can be Text, Image, Audio, Video.

3. DESIGN

Project design is a major step towards a successful project. A project design is a strategic organization of ideas, materials and processes for the purpose of achieving a goal. Project managers rely on a good design to avoid pitfalls and provide parameters to maintain crucial aspects of the project. Project design is an early phase of the project where a project's key features, structure, criteria for success, and major deliverables are all planned out. The point is to develop one or more designs which can be used to achieve the desired project goals. Stakeholders can then choose the best design to use for the actual execution of the project. The project design phase might generate a variety of different outputs, including sketches, flowcharts, HTML screen designs, and more.

So, the design can be implemented using Unified Modeling Language. diagrams such as class diagram, use case diagram, sequence diagram, activity diagrams. UML offers a way to visualize a system’s architectural blueprints in a diagram, including elements such as :

- Any activites
- Individual components of the system
- How the system will run
- How entities interact with others
- External user interface

UML is a common language for business analysts, software architects and developers used to describe, specify, design, and document existing or new business processes, structure and behaviour of artifacts of software systems. The key to making a UML diagram is connecting shapes that represent an object or class with other shapes to illustrate relationships and the flow of information and data.

4.1 Class Diagram

A class diagram in the Unified Modelling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

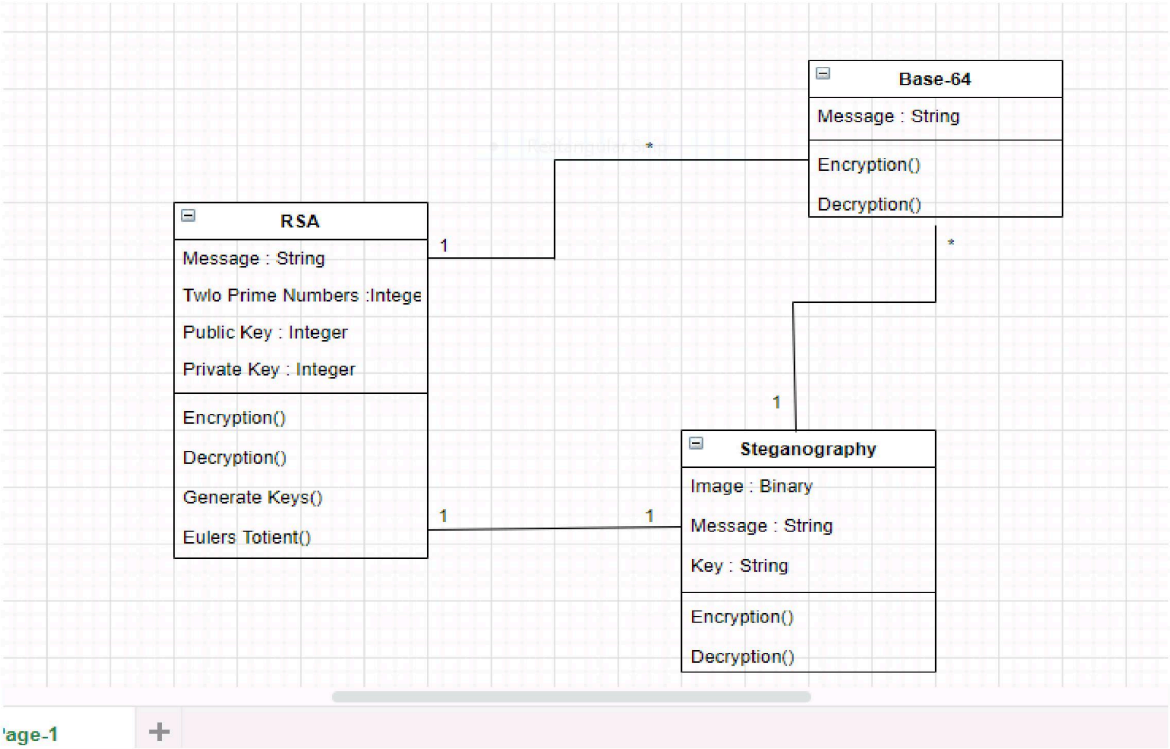


Fig 4.1 Class Diagram

4.2 Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

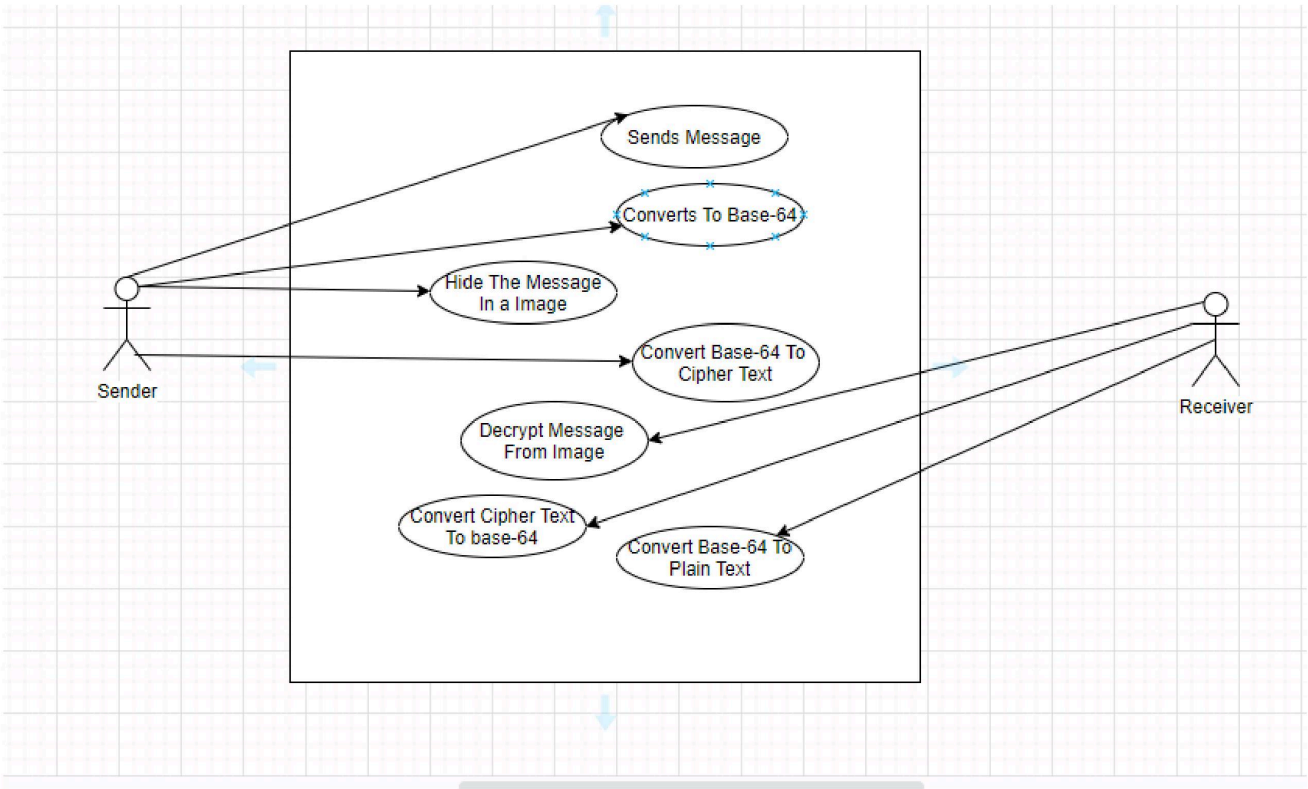


Fig 4.2 Use Case Diagram

4.3 Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

A sequence diagram shows, as parallel vertical lines, different processes or objects that live simultaneously and as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

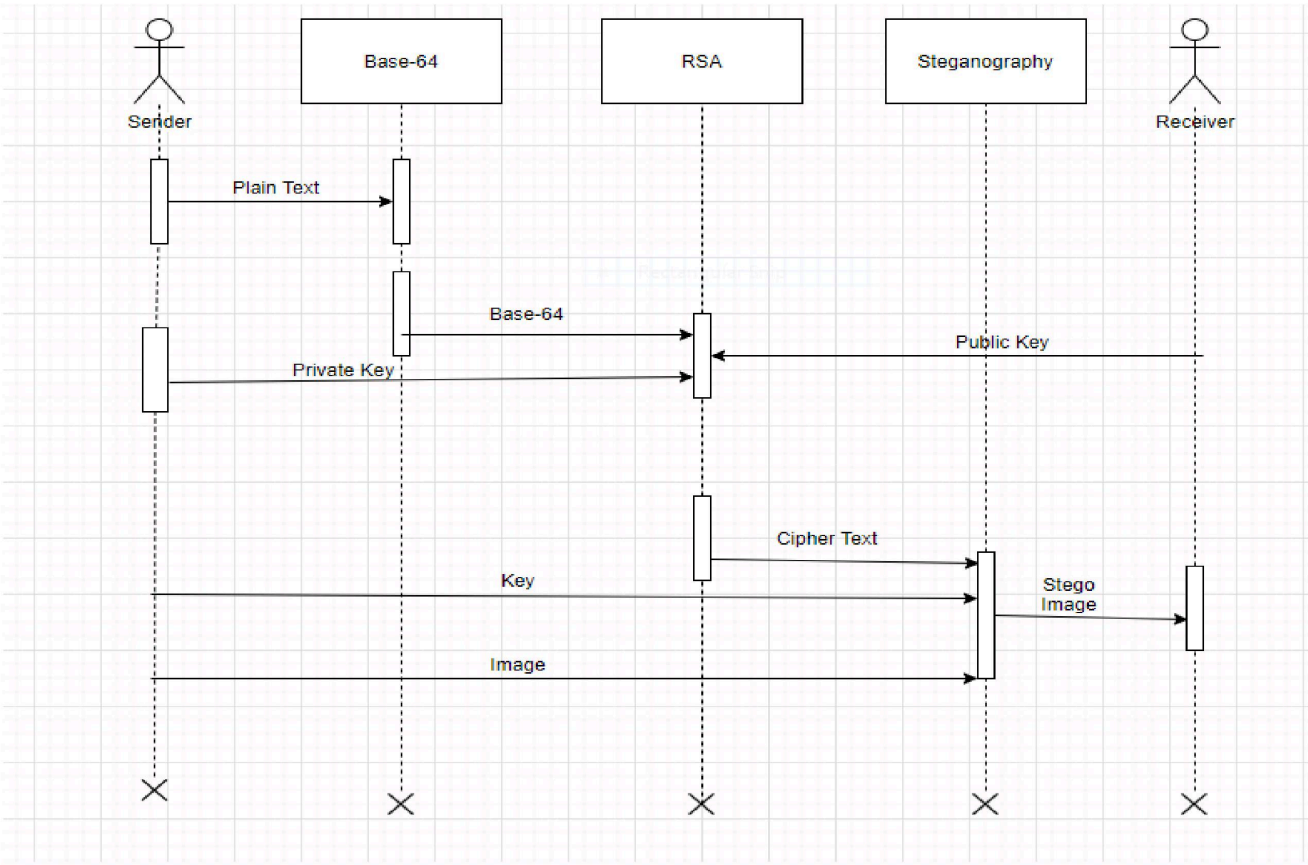


Fig 4.3 Sequential Diagram

4.4 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows) as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

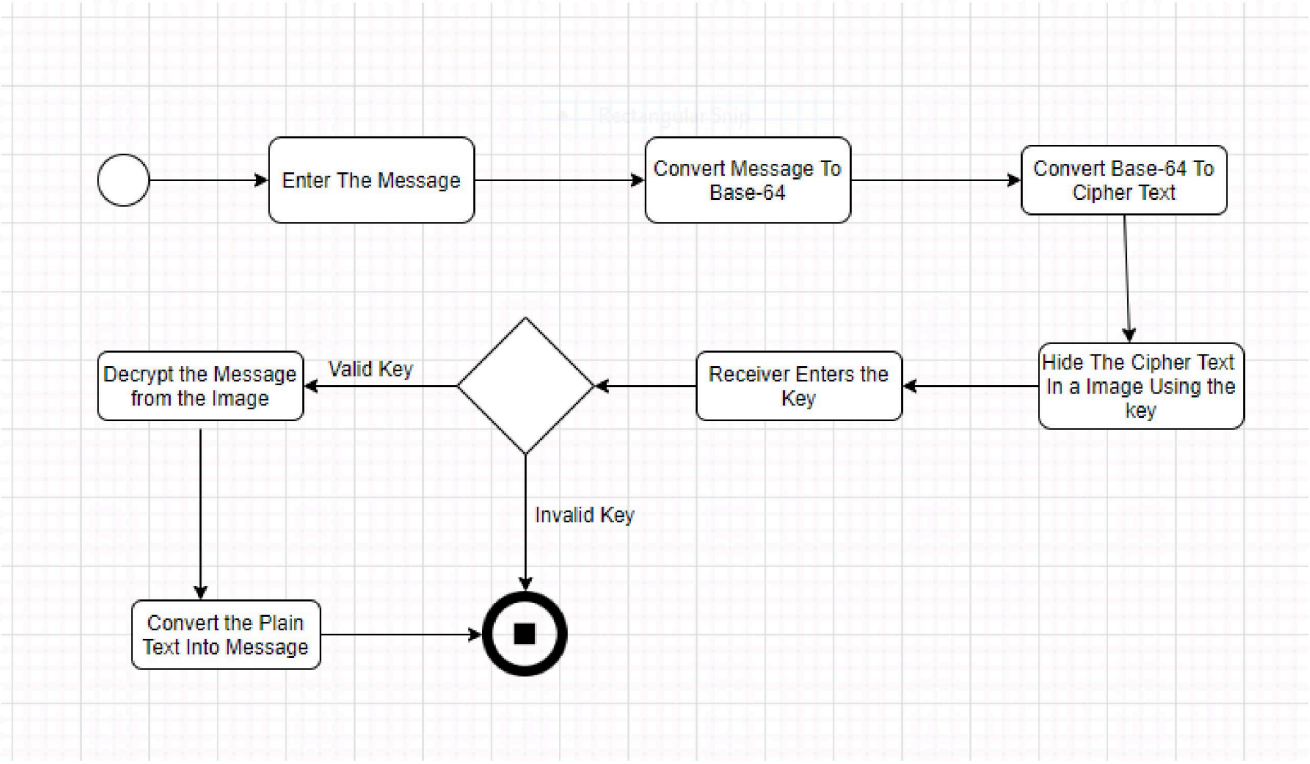


Fig 4.4 Activity Diagram

4. EXPERIMENTAL ANALYSIS AND RESULTS

5.1 SYSTEM CONFIGURATION

5.1.1 Software Requirements:

The software configurations used are

Operating System: Windows 10

Programming Language : Python

Audio file format: m4a (any file format is accepted)

5.1.2 Hardware Requirements:

Processor: INTEL

RAM: Minimum of 256 MB or higher

HDD: 10GB or higher

Monitor: 15” or 17” color monitor

Keyboard: Standard 110 keys keyboard.

5.2 SAMPLE CODE

5.2.1 Home Page

```
<html >
   <head >
      <title>Home</title>
      <link href= "{{ url_for('static',filename = 'css/bulma.css') }}" rel="stylesheet">
   </head>
   <body background="{{ url_for('static',filename = 'images/image15.jpg') }}">

      <h1 class="title is-1 has-text-centered has-text-white"> Welcome To </h1>
      <h1 class="title is-1 has-text-centered has-text-white"> Secure Encryption And Decryption Using
Crypto And Stego </h1>
      <div style="padding-left: 10%; padding-top: 5%">
      <div class="tile is-ancestor">
         <div class="tile is-6">
            <a href="Encryption" ><img src="{{ url_for('static',filename = 'images/encryptlogo.jpeg')
}}"></a>
         </div>
         <div class="tile is-6">
            <a href="Decryption" ><img src="{{ url_for('static',filename = 'images/decryptlogo.jpg')
}}"></a>
         </div>
      </div>
      <div class="tile is-ancestor">
         <div class="tile is-6">
            <p class="title is-3 has-text-white">Sender Side</p>
         </div>
         <div class="tile is-6">
            <p class="title is-3 has-text-white">Receiver Side</p>
         </div>
      </div>
      </div>
   </body>
</html>
```

5.2.1.1 Encryption

```
<html>
   <head>
   <title>Encryption</title>
      <link rel="stylesheet" type="text/css" href="{{ url_for('static',filename = 'css/bulma.css') }}">
   </head>
   <body style="padding-left: 20%; padding-top: 5%;padding-right: 20%" background="{{
url_for('static',filename = 'images/encrypt_body.jpg') }}">
```

```
    <div style="padding-left: 20%; padding-top: 10%;padding-right: 20%; padding-bottom: 20">
    <img src="">
      <h1 class="title is-2 has-text-white">Sender Side</h1>
      <form method="post">
        <div class="field">
          <label class="label has-text-white" >Source Name</label>
          <div class="control">
            <input class="input" type="text" name="source_name" placeholder="Source Name" required>
          </div>
        </div>
        <div class="field">
          <label class="label has-text-white">Prime 1</label>
          <div class="control">
            <input class="input" type="text" name="prime_1" placeholder="Enter Prime no.1" required>
          </div>
        </div>
        <div class="field">
          <label class="label has-text-white">Prime 2</label>
          <div class="control">
            <input class="input" type="text" name="prime_2" placeholder="Enter Prime no.2" required>
          </div>
        </div>
        <div class="field">
          <label class="label has-text-white">Cover Name</label>
          <div class="control">
            <input class="input" type="text" name="cover_name" placeholder="Cover name" required>
          </div>
        </div>
        <div class="field">
          <label class="label has-text-white">New Image Name</label>
          <div class="control">
            <input class="input" type="text" name="new_name" placeholder="Enter New Name for saving Image" required>
          </div>
        </div>
        <input type="submit" name="" class="button" value="Submit">
      </form>

    </div>

  </body>
</html>
```

5.2.1.2 Decryption

```
<html>
   <head>
   <title>Decryption</title>
      <link rel="stylesheet" type="text/css" href="{{ url_for('static',filename = 'css/bulma.css') }}">
   </head>
   <body style="padding-left: 20%; padding-top: 5%;padding-right: 20%" background="{{
url_for('static',filename = 'images/encrypt_body.jpg') }}">
      <div style="padding-left: 20%; padding-top: 10%;padding-right: 20%; padding-bottom: 20">
      <img src="">
         <h1 class="title is-2 has-text-white">Reciever Side</h1>
         <form method="post">
            <div class="field">
              <label class="label has-text-white">Cover Name</label>
              <div class="control">
                <input class="input" type="text" name= "cover_name" placeholder="Source Name"
required>
              </div>
            </div>
            <div class="field">
              <label class="label has-text-white">Prime 1</label>
              <div class="control">
                <input class="input" type="text" name= "prime_1" placeholder="Enter Prime no.1"
required>
              </div>
            </div>
            <div class="field">
              <label class="label has-text-white">Prime 2</label>
              <div class="control">
                <input class="input" type="text" name= "prime_2" placeholder="Enter Prime no.2"
required>
              </div>
            </div>
            <div class="field">
              <label class="label has-text-white">new Cover Name</label>
              <div class="control">
                <input class="input" type="text" name= "new_cover_name" placeholder="enter new Name"
required>
              </div>
            </div>


            <input type="submit" name="" class="button" value="Submit">
         </form>

      </div>
```

```
   </body>
</html>
```

5.2.2 Base-64

5.2.2.1 Encryption:

```
import base64
with open('C:/Users/HP/Desktop/Project/ping.jpg', "rb") as File:
str1= base64.b64encode(File.read())
print(str1)
filename = 's.txt'
# we are considering a file to store the string.
with open(filename, 'wb') as f:
f.write(str1)
```

5.2.2.2 Decryption:

```
import base64
with open('s.txt', "rb") as File:
str1= (File.read())
imgdata = base64.b64decode(str1)
filename = 'C:/Users/HP/Desktop/Project/pingsss.jpg'
with open(filename, 'wb') as f:
f.write(imgdata)
```

5.2.2.3 DECRYPTION

```
from PIL import Image
def genData(data):
    # list of binary codes #
    of given data
    newd = []

    for i in data: newd.append(format(ord(i),
      '08b'))
    return newd

def decode():
    img = input("Enter image name(with extension) : ")
    image = Image.open(img, 'r')

    data = ''
    imgdata = iter(image.getdata())

    while (True):
        pixels = [value for value in imgdata._____next___()[:3] +imgdata.__next___()[:3] + imgdata._ next
        (i % 2 == 0):
                binstr += '0'
            else:
```

5.3 Testing

Fig 5.3.1 Cover Image without any Data Embedded in its key Channel

Fig 5.3.2 : Cover Image with Data Embedded in its key Channel

Performance is usally calculated as a number of correct outputs that we get for the given data set input. The schedule performance index is a measure of how close the project is to be being completed compared to the schedule. As a ratio it is calculated by dividing the budgeted cost of work performed, or earned value, by the planned value.

Module .1	File Name	Resolution (w*h)	Encryption Time (In Sec)	Decryption Time (In Sec)
Base-64	Ping.png	1080*2160	0.15621	0.0189
Base-64	Sample.jpg	512*320	0.03124	0.0065
Base-64	Anits.jpg	1024*768	0.015	0.0053
Base-64	Picture.jpg	1024*760	0.015	0.0049
Base-64	Audio.mp3	-	0.18856	0.0613
Base-64	Video.mp4	-	0.28654	0.0862
RSA	-	-	8.5	18.0
Steganography	Flower.png	1080*2160	26.0	6.9

5.3 Performance Measure

The performance measure depends on the success rate of the implementation of the overall system with respect to the following points.

a) The integrity of the hidden information should not change after embedding.

b) The stego object must remain almost remain unchanged to the naked eye.

c) There should be accuracy in the extracted data.

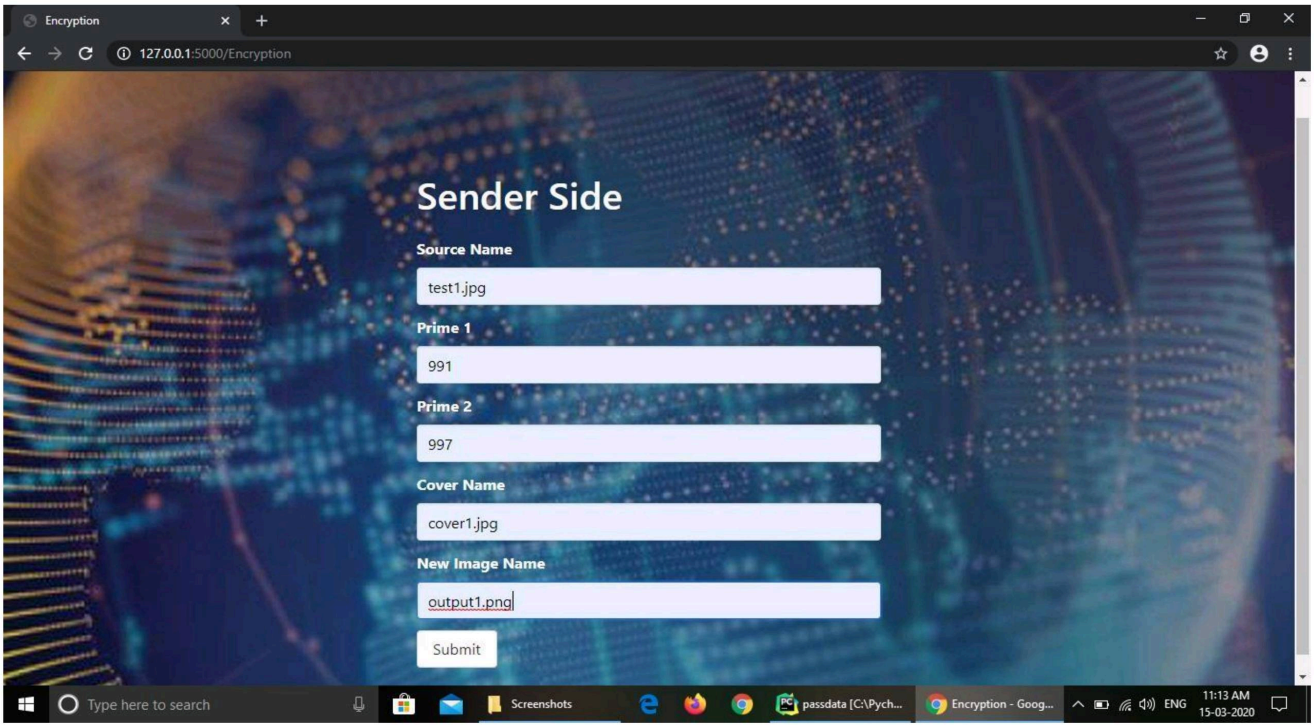
Simple methods to observe if an image file has been manipulated are:

1. Size of the image: A Steganographic image has a huge storage size when compared to a regular image of the same dimensions. I.e. if the original image storage size would be few KBs, the Steganographic image could be several MBs in size. This again varies with the resolution and type of image used.

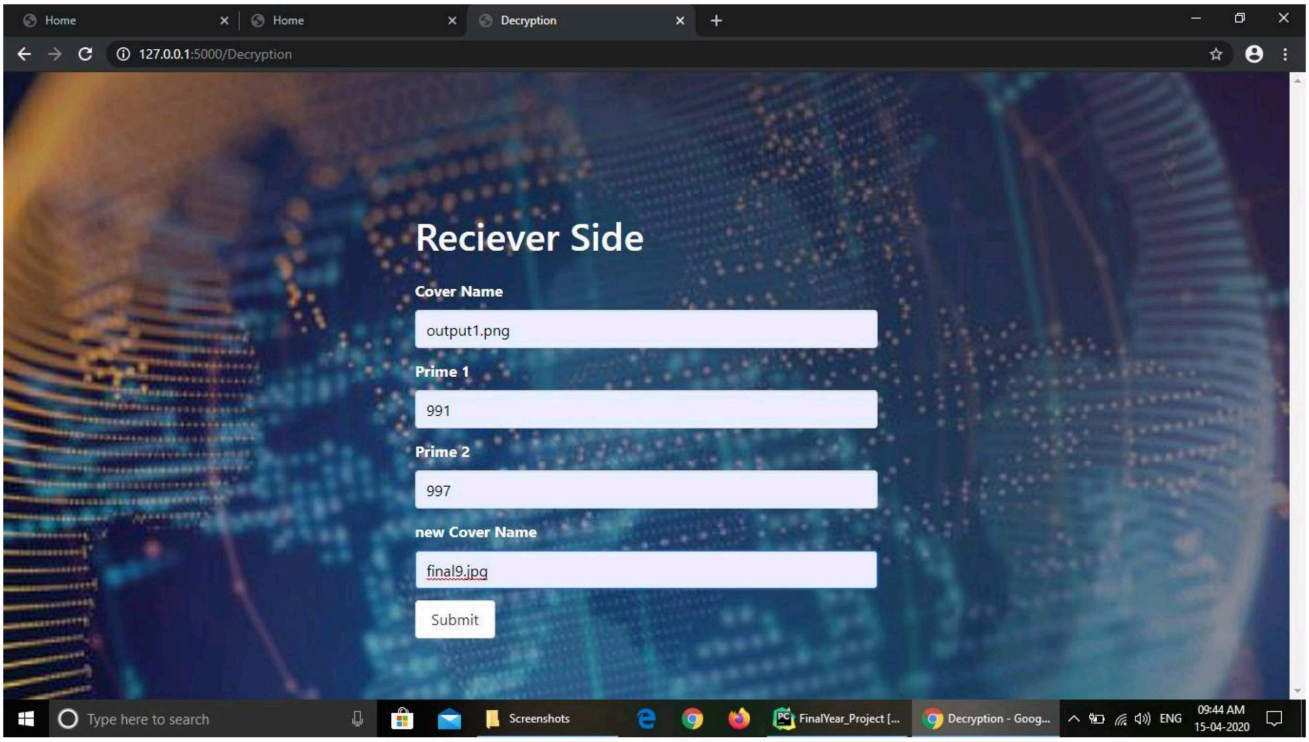
2. Noise in image: A Steganographic image has noise when compared to a regular image. This is the reason why initially little noise is added to the cover image, so that the Steganographic image doesn't appear very noisy when compared to the original cover image.

RESULTS

Home Page



: Sender Side



Reciever Side

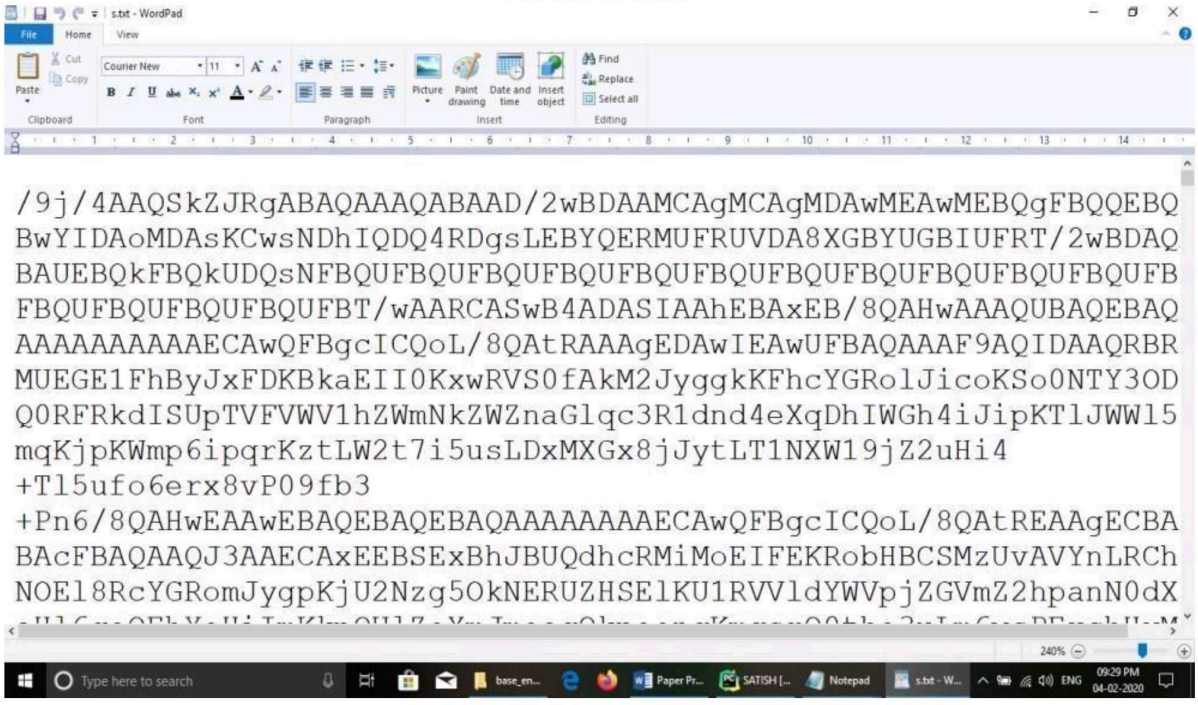
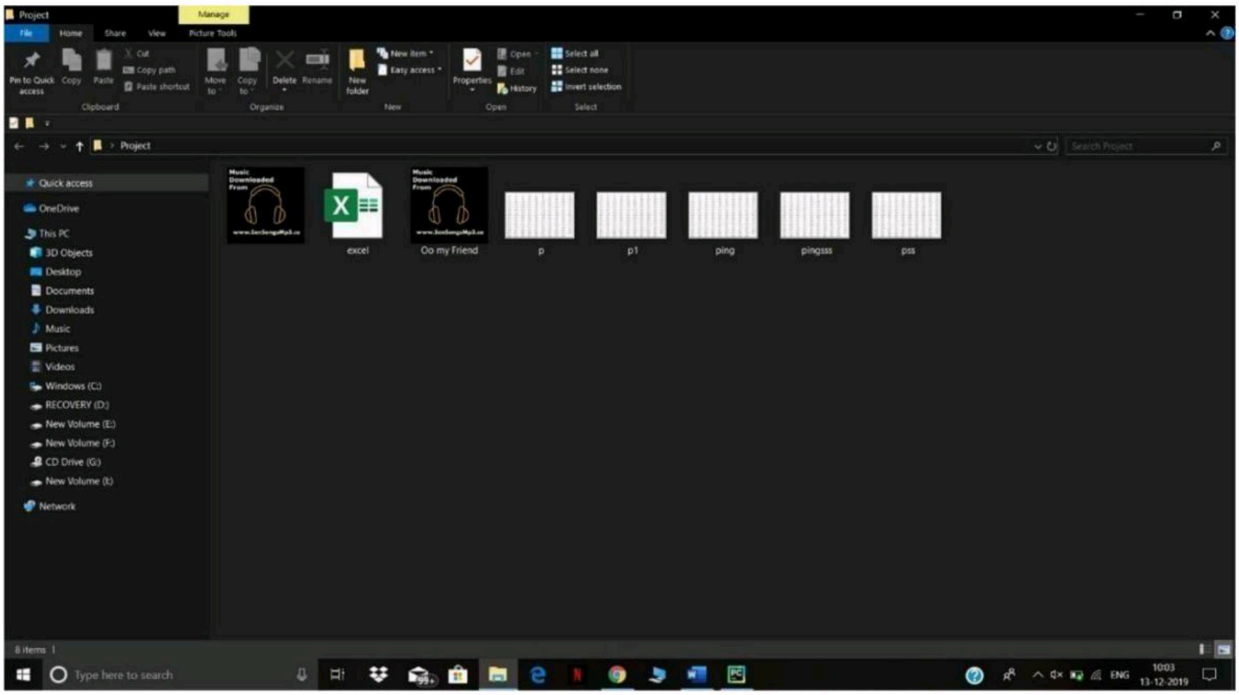


Image Encryption

Fig:5.4.5 Image Decryption

```
t#r5q&jS5LIAAAA7ExOtExjhEqZ5qI0qrxtIOq25YvPIquqY*nqQAAAAj
#GudvrxR#%*qq!QEq!5YIAALdYj5L&Ge
%j#je&5heTKFEPTeqJm5ERj*e&jAqZoLmq#Vee&AAxeTx!
xqOr&Gme2))P&Z##SqZ)&O#Sh%G
%ZJ5PoAj5AAZqdq&AuqtVN5tGKVQQGZt2qZjKGm*Zqqd&xmeZm0vqhKRx
KFIjdxLFKh&&ItrRqj&5h3KP*32Zh5eK5PmJ555qKTKRP7)m5Km5RFNeY
%ddePVmeYmSq)**Y3oYTGKSTtEG&n2JONPnveuYY1%51Y
%e3TVY3OJOdnTmGE2AYnPTN#EqhGNqGZn2h)nn&NTYEeY!hNvmmY5Vr
%TVG&KP1*1Eoj5%Tme
%JPEGqG&hTmqG5h2ZGNZZm*Kj)*7Y1*YQ5hm&GdGJeC)JerGEKTKGGqqq
#0P07TAx%A2GqJq5**Y1hIPIqR&JZY*mJCN2o5eqJ&ZS*YYq%
```

Audio Encryption



6. CONCLUSION AND FUTURE SCOPE

In this project, we deal with the concepts of security of digital data communication across the network. This project is designed for combining the steganography and cryptography features factors for better performance. We performed a new steganography method and combined it with RSA algorithm. The data is hidden in the image so there will be no chances for the attacker to know that data is being hidden in the image. We performed our method on image by implementing a program written in Python language. The method proposed has proved successful in hiding various types of text, images, audio and videos in color images. We concluded that in our method the Image files and RSA are better. Because of their high capacity.

This work presents a scheme that can transmit large quantities of secret information and provides secure communication between two private parties. Both steganography and cryptography can be woven in this scheme to make the detection more complicated. Any kind of text data can be employed as secret msg. The secret message employing the concept of steganography is sent over the network. In addition, the proposed procedure is simple and easy to implement.

The Embedding of data is done such as Audio, Video, Image is done in the image, by choosing a distinct and new image, we can prevent the chance for the attacker to detect the data being hidden. Results achieved indicate that our proposed method is encouraging in terms of security, and robustness.

7. REFERENCES

[1] D. Seth, L. Ramanathan, and A. Pandey, "Security enhancement: Combining cryptography and steganography," International Journal of Computer Applications (0975–8887) Volume, 2010.

[2] H. Abdulzahra, R. AHMAD, and N. M. NOOR, "Combining cryptography and steganography for data hiding in images," ACACOS, Applied Computational Science, pp. 978–960, 2014.

[3] J. V. Karthik and B. V. Reddy, "Authentication of secret information in image stenography," International Journal of Computer Science and Network Security (IJCSNS), vol. 14, no. 6, p. 58, 2014.

[4] M. H. Rajyaguru, "Crystography-combination of cryptography and steganography with rapidly changing keys," International Journal of Emerging Technology and Advanced Engineering, ISSN, pp. 2250–2459, 2012.