SECURE DATA ENCRYPTION AND DECRYPTION USING CRYPTO-STEGO

A PROJECT REPORT

Submitted by

M. VENKATESH	**316126510028**
G. SATISH	**316126510010**
K. RAM SUDEEP	**316126510018**
M. SUDARSHAN	**316126510051**

In partial fulfillment for the award **of the degree Of**

BACHELOR OF TECHNOLOGY IN

COMPUTER SCIENCE ENGINEERING

Under the Guidance of
Mrs. T.ANITHA
Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(UGC AUTONOMOUS)
(*Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)
Sangivalasa, visakhapatnam - 531162

2019-2020

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGYAND SCIENCES

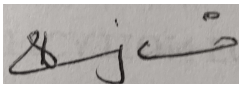
(UGC Autonomous)

(*Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)

Sangivalasa, bheemili mandal, visakhapatnam dist.(A.P)

BONAFIDE CERTIFICATE

Certified that this project report **"SECURE DATA ENCRYPTION AND DECRYPTION USING CRYPTO-STEGO"** is the bonafide work of "M.VENKATESH (316126510028), G.SATISH (316126510010), K.SUDDEP (316126510018), M.SUDARSHAN (316126510051)" who carried out the project work under my supervision**.**



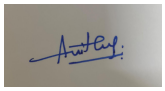
Dr. R.Sivaranjani

HEAD OF THE DEPARTMENT

COMPUTER SCIENCE AND ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

(AUTONOMOUS)



Mrs. T.Anitha

ASSISTANT PROFESSOR

COMPUTER SCIENCE AND ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

(AUTONOMOUS)

DECLARATION

This is to certify that the project work entitled **"SECURE DATA ENCRYPTION AND DECRYPTION USING CRYPTO-STEGO"** is a bonafide work carried out by **M.VENKATESH, G.SATISH, K.SUDDEP, M.SUDARSHAN** as a part of **B.Tech** final year second semester of **Computer Science & Engineering from Anits** during the year 2019-2020.

We **M.VENKATESH(316126510028), G.SATISH(316126510010), K.SUDDEP (316126510018), M.SUDARSHAN(316126510051)**" students of Fourth year second semester B.TECH, **Computer Science & Engineering from Anits**, Visakhapatnam, hereby declare that the project work entitled **"SECURE DATA ENCRYPTION AND DECRYPTION USING CRYPTO-STEGO"** is carried out by us and submitted in fulfillment of the requirements for the award of Bachelor of Technology in **Computer Science and Engineering,** under Anil Neerukonda Institute of Technology & Sciences during the Academic year 2016-2020 and has not been submitted to any other university.

M. VENKATESH	**316126510028**
G. SATISH	**316126510010**
K. RAM SUDEEP	**316126510018**
M. SUDARSHAN	**316126510051**

ACKNOWLEDGEMENT

An endeavor over a long period can be advice and support of many well-wishers. We take this opportunity to express our gratitude and appreciation to all of them.

We owe our tributes to **Dr. R.Sivaranjani, Head of the Department, Computer Science & Engineering for** her valuable support and guidance during the period of project implementation.

We wish to express our sincere thanks and gratitude to our project guide **Mrs. T.Anitha** Assistant professor Department of **COMPUTER SCIENCE AND ENGINEERING, ANITS,** for the simulating discussions, in analyzing problems associated with our project work and for guiding us throughout the project. We express our sincere thanks for the encouragement, untiring guidance and the confidence they had shown in us.

We also thank our project coordinator **Dr. K. Suresh, Assistant Professor,** Department of Computer Science and Engineering, ANITS, for his constant support throughout our project period

We also thank **Principal** and all **non-teaching staff** of the department of CSE, ANITS for providing resources when required.

M. VENKATESH	**316126510028**
G. SATISH	**316126510010**
K. RAM SUDEEP	**316126510018**
M. SUDARSHAN	**316126510051**

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

LIST OF FIGURES

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.

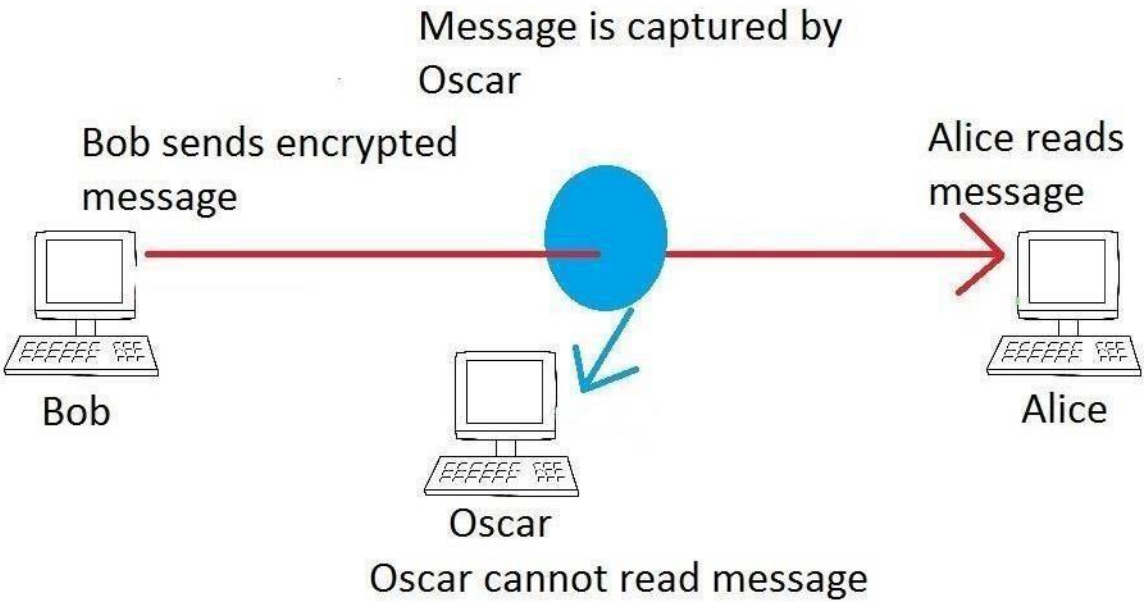


Fig 1.1.1 : Cryptography as a flow model.

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 .

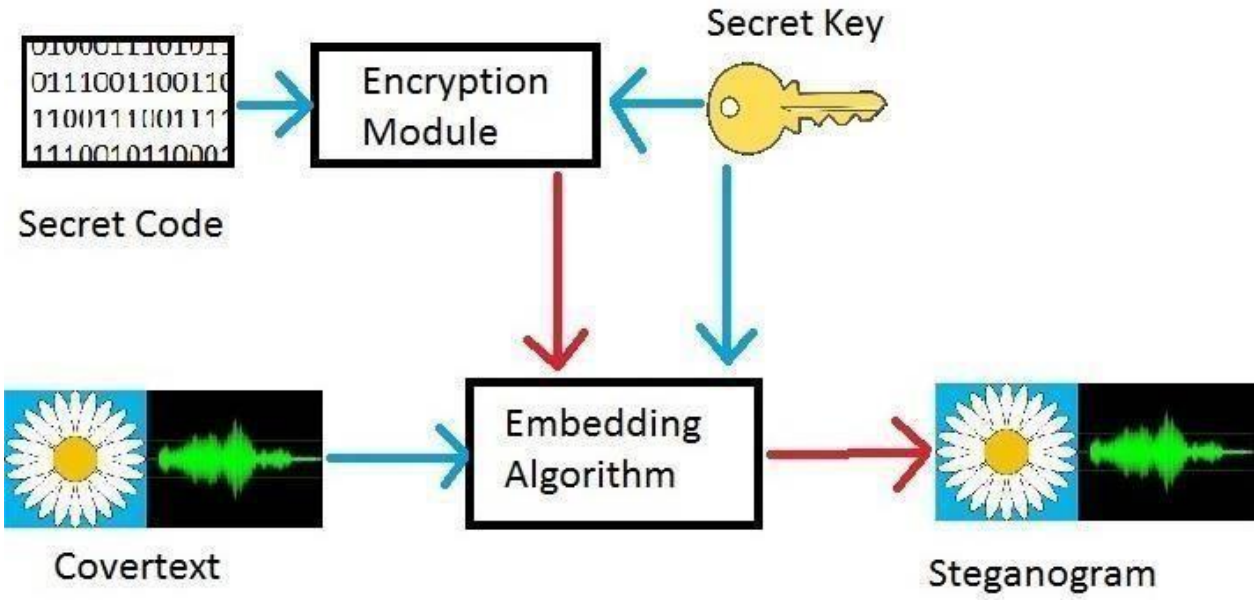


Fig 1.1.2 : Stegonography as a flow model.

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.

1.4 ORGANIZATION OF THESIS

The organization of this thesis is as follows.

Chapter-1 is about introduction which gives an idea about of our project domain i.e Network Security and title is explained i.e., Secure Data Encryption and Decryption Using Crypto-Stego, how the data is transmitted between two private parties.

Chapter-2 is about literature survey where all previous methods and existing models are examined.

Chapter-3 contains methodology, where algorithm is implemented for hiding the data in an image. Even architecture of the system is explained thoroughly.

Chapter-4 consists of design which includes UML diagrams such as class diagram, use-case, sequence and activity diagram.

Chapter-5 consists experimental analysis and results in this sample code, testing results, system configurations such as software and hardware requirements, input and output images are displayed.

Chapter-6 explains conclusion and future work about our project i.e Secure Data Encryption And Decryption Using Crypto-Stego.

2. LITERATURE SURVEY

As we said the significance of network security is increased day by day as the size of data being transferred across the Internet. This issue pushes the researchers to do many studies to increase the ability to solve security issues. A solution for this issue is using the advantage of cryptography and steganography combined in one system. Many studies propose methods to combine cryptography with steganography systems in one system. This Project has been implemented on the basis of the requirements of security i.e. authentication, confidentiality, and robustness.

There has been a continuous rise in the number of data security threats in the recent past and it has become a matter of concern for the security experts. Cryptography and steganography are the best techniques to nullify this threat. The researchers today are proposing a blended approach of both techniques because a higher level of security is achieved when both techniques are used together.

In proposed an encrypting technique by combining cryptography and steganography techniques to hide the data. In cryptography process, we proposed an effective technique for data encryption using one's complement method. It used an Asymmetric key method where both sender and receiver share the Secret key for encryption and decryption. In steganography part, we used the LSB method that is used and mostly preferred.

We present a method based on combining both the strong encrypting algorithm and steganographic technique to make the communication of confidential information safe, secure and extremely hard to decode. An encryption technique is employed for encrypting a secret message into a Cipher text using the Senders Private Key and receiver public key. The Cipher Text is finally embedded in a suitable cover

image and transferred securely to deliver the secret information. They utilized a least significant bit method to accomplish the digital image steganography.

At the receiver's side, the secret data is retrieved through the decoding process. Thus, a three-level security has been rendered for them a secret message to be transferred.

3. METHODOLOGY

3.1 SYSTEM ARCHITECTURE

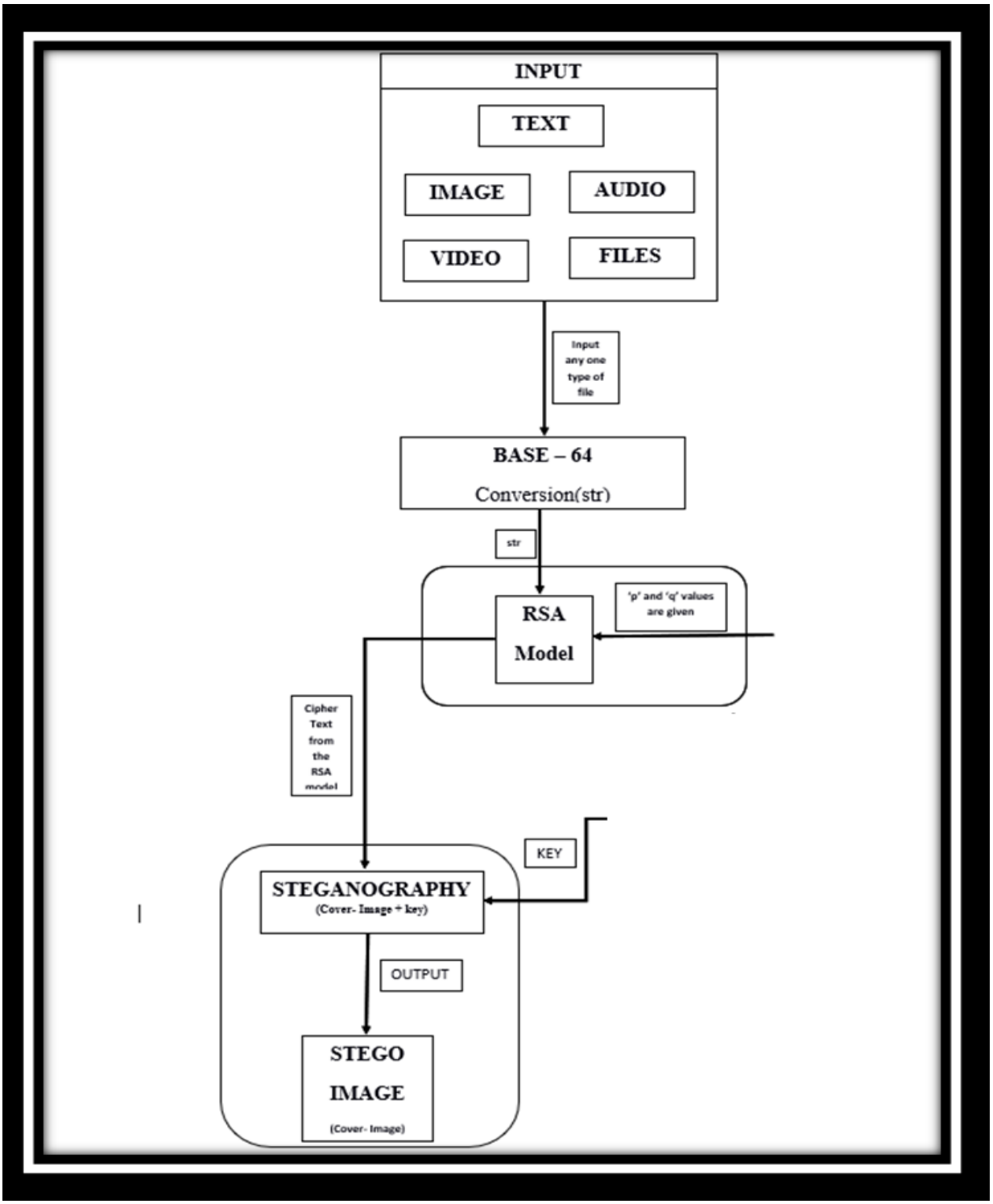
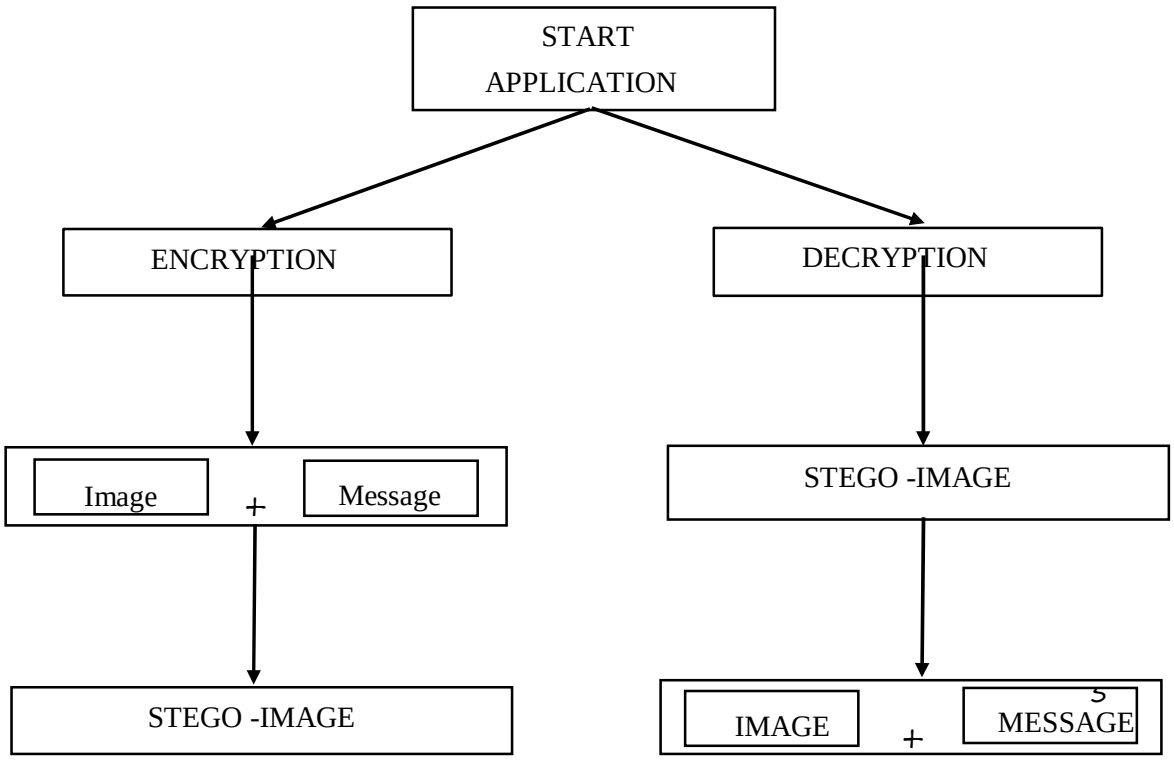


Fig 3.1 System Architecture

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.

3.3 MODULE DIVISION

3.3.1 Base-64

Base 64 is an encoding scheme that converts binary data into text format so that encoded textual data can be easily transported over network un-corrupted and without any data loss. Base64 is used commonly in a number of applications including email via MIME, and storing complex data in XML. Problem with sending normal binary data to a network is that bits can be misinterpreted by underlying protocols, produce incorrect data at receiving node and that is why we use this method. The term Base64 is taken from the Multipurpose Internet Mail Extension (MIME) standard, which is widely used for HTTP and XML, and was originally developed for encoding email attachments for transmission.

3.3.1.1 Why Do We Use Base64 ?

Base64 is very important for binary data representation, such that it allows binary data to be represented in a way that looks and acts as plain text, which makes it more reliable to be stored in databases, sent in emails, or used in text-based format such as XML. Base64 is basically used for representing data in an ASCII string format.

3.3.1.2 Base64 Encoding

Base64 encoding is the process of converting binary data into a limited character set of 64 characters. The characters are A-Z, a-z, 0-9, +, and / . This character set is considered the most common character set, and is referred to as MIME's Base64. It uses A-Z, a-z, 0-9, +, and / for the first 62 values, and +, and / for the last two values. The Base64 encoded data ends up being longer than the original data, so that, for every 3 bytes of binary data, there are at least 4 bytes of Base64 encoded data. This is due to the fact that we are squeezing the data into a smaller set of characters.

3.3.1.3 Base64 Decoding

Base64 decoding is the opposite of Base64 encoding. In other words, it is carried out by reversing the steps described in the Encoding. So, the Each character in the string is changed to its Base64 decimal value. The decimal values obtained are converted into their binary equivalents. The first two bits of the binary numbers are truncated from each of the binary numbers obtained, and the sets of 6 bits are combined, forming one large string of binary digits. The large string of binary digits obtained in the previous step is split into groups of 8 bits. The 8-bit binary numbers are converted into their decimal equivalents. Finally, the decimal values obtained are converted into their ASCII equivalent.

3.3.1.4 Usage

Base64 is most commonly used to encode binary data (for example images, or sound files) for embedding into HTML, CSS, EML, and other text documents. In addition, Base64 is used to encode data that may be unsupported or damaged during transfer, storage, or output.

Some of the applications of the algorithm:

- Attach files when sending emails
- Embed images in HTML or CSS via data URI
- Preserve raw bytes of cryptographic functions
- Output binary data as XML or JSON in API responses
- Save binary files to database when BLOB is unavailable

3.3.2 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

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.

RSA derives its security from the difficulty of factoring large integers that are the product of two large prime numbers. Multiplying these two numbers is easy, but determining the original prime numbers from the total or factoring is considered infeasible due to the time it would take using even today's supercomputers.

3.3.2.1 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.

The private key consists of the modulus n and the private exponent d, which is calculated using the Extended Euclidean algorithm to find the multiplicative inverse with respect to the totient of n.

3.3.2.2 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.

3.3.2.3 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)

3.3.3 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

Image Steganography

Image Steganography deals with the hiding of data within the image, data can be any file, such as an image, audio, text or another file. We have to wrap up the data using an image. We have chosen to bind data in an image. This kind of embedding an audio in an image helps to authenticate the sender, verify whether valid user is receiving the data or not and to find whether a third party attacker is present in the channel of communication or not. Since, image is used as a cover file, we have to make sure that the image must be accountable for the data that is being embedded. Hence a 24-bit image format proved to be the best solution for hiding the data, since it holds a large memory space and convenient to hide a considerable amount of data. Furthermore, the threshold sure of the image must be calculated for the given image size which will be explained in the laier parts.

LSB Positioning Method

This method is the simplest method of hiding data within in the given image. We utilizes the LSB bits of the pixels within the given image. When converting the image to digital format, we usually choose between three different ways of representing colors :

- 24-bit color : every pixel can have one in 2^24 colors, and these are represented as different quantities of three basic colors : red(R), green(G), blue(b) given by 8 bits (256) each.
- 8-bit color : every pixel can have one in 256 (2^28) colors, chosen from a palette, or a table of colors.
- 8-bit gray-scale : every pixel can have one in 256 (2^28) shades of gray.

LSB insertion modifies the LSBs of each color in 24-bit images, or the LSBs of the 8-bit value for 8-bit images. The most basic of LSBs insertion for 24-bit pictures inserts 3 bits/pixel.

For image steganography we are using Spatial methods. In spatial method, the most common method used is LSB substitution method. Least significant bit (LSB) method is a common, simple approach to embedding information in a cover file. In steganography, LSB substitution method is used. I.e. since every image has three components (RGB). This pixel information is stored in encoded format in one byte. The first bits containing this information for every pixel can be modified to store the hidden text. For this, the preliminary condition is that the text to be stored has to be smaller or of equal size to the image used to hide the text. LSB based method is a spatial domain method. But this is vulnerable to cropping and noise. In this method, the MSB (most significant bits) of the message image to be hidden are stored in the LSB (least significant bits) of the image used as the cover image.

The Human visual system (HVS) cannot detect changes in the colour or intensity of a pixel when the LSB bit is modified. This is psycho-visual redundancy since this can be

used as an advantage to store information in these bits and yet notice no major difference in the image.

3.3.3.1 Algorithm

Inputs : Image, Message, Key.

1) Intially Sender consider a Cover Image.
2) Hide the Encrypted message in the image.
3) Hidding can be done using a secret key for confidentiality.
4) After Hidding the Image is considered as a Stego-Image. Which consists of image and data which is encrypted.
5) The Receiver will receive the Stego-Image.
6) Using the Secret Key the receiver can view the data hidden in the image.
7) Thus the receiver can receive the message safely.

3.4 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.

4. 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.

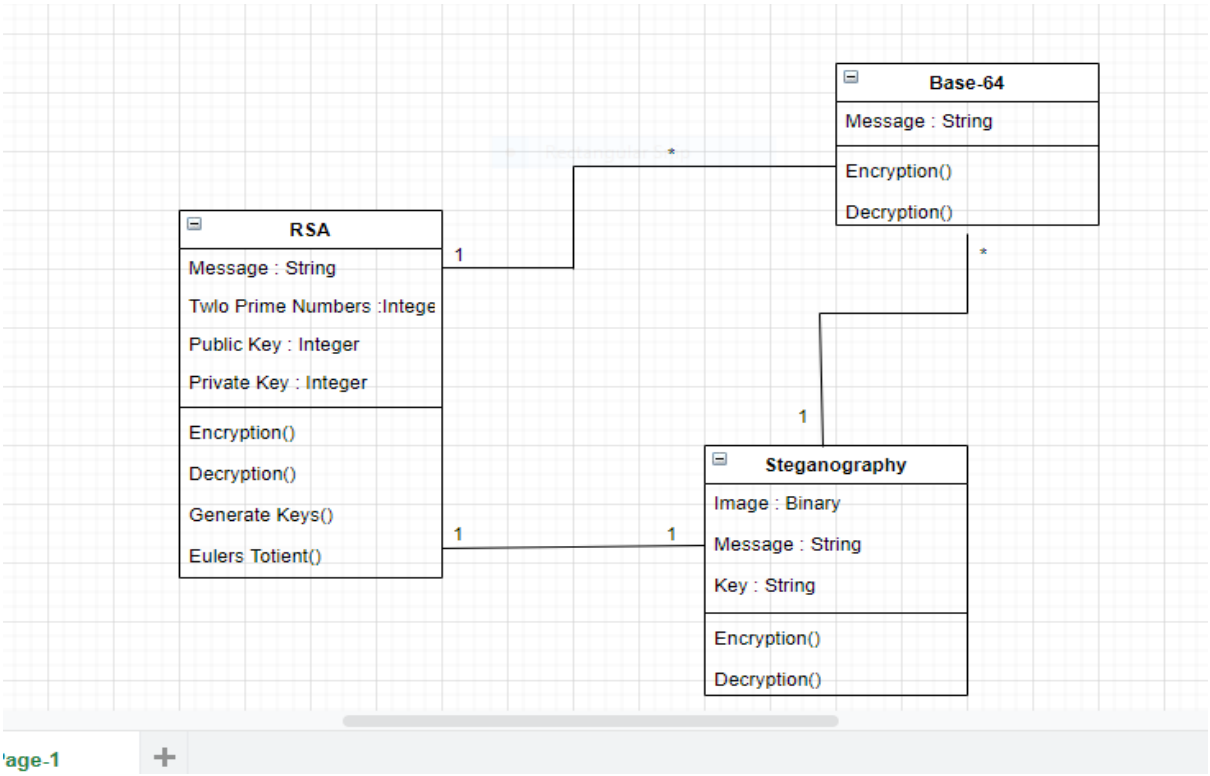


Fig 4.1 Class Diagram

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.

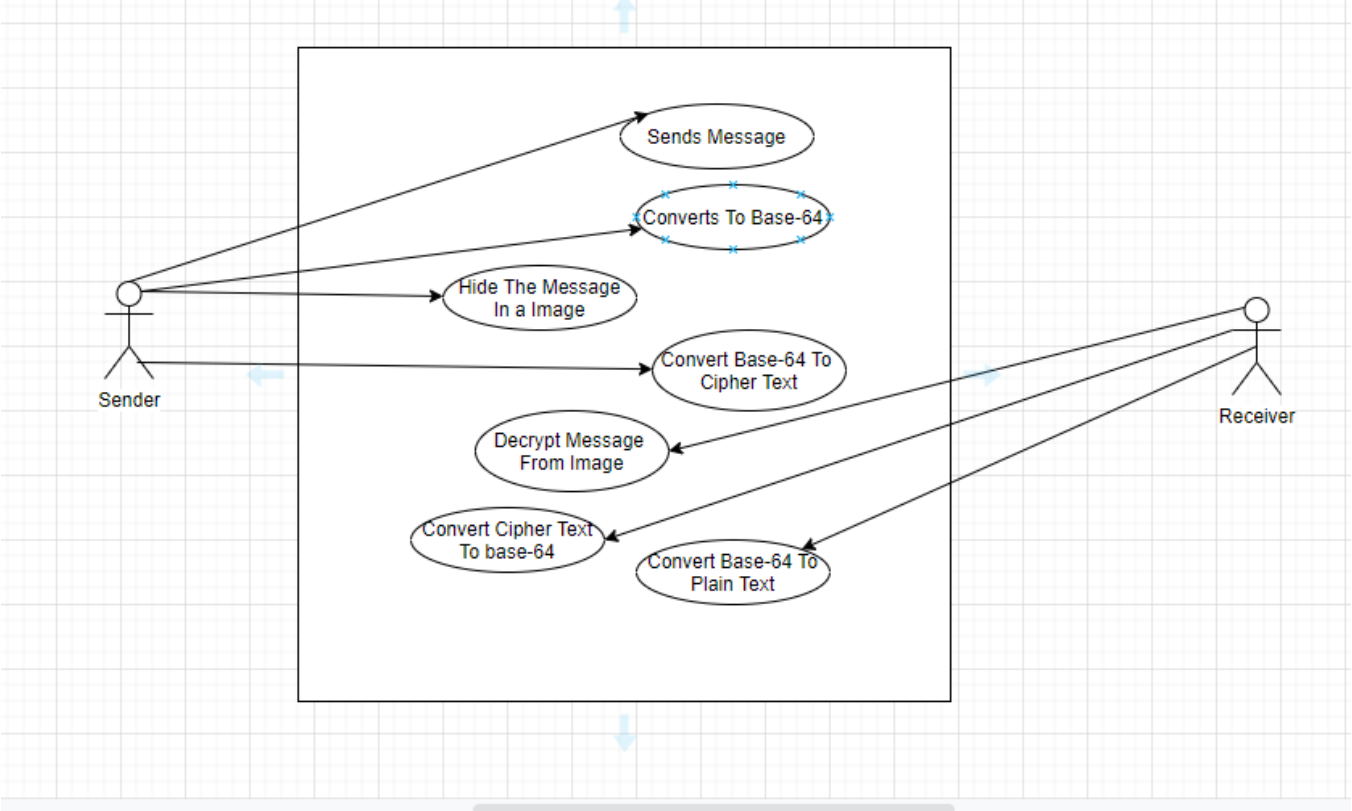


Fig 4.2 Use Case Diagram

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.

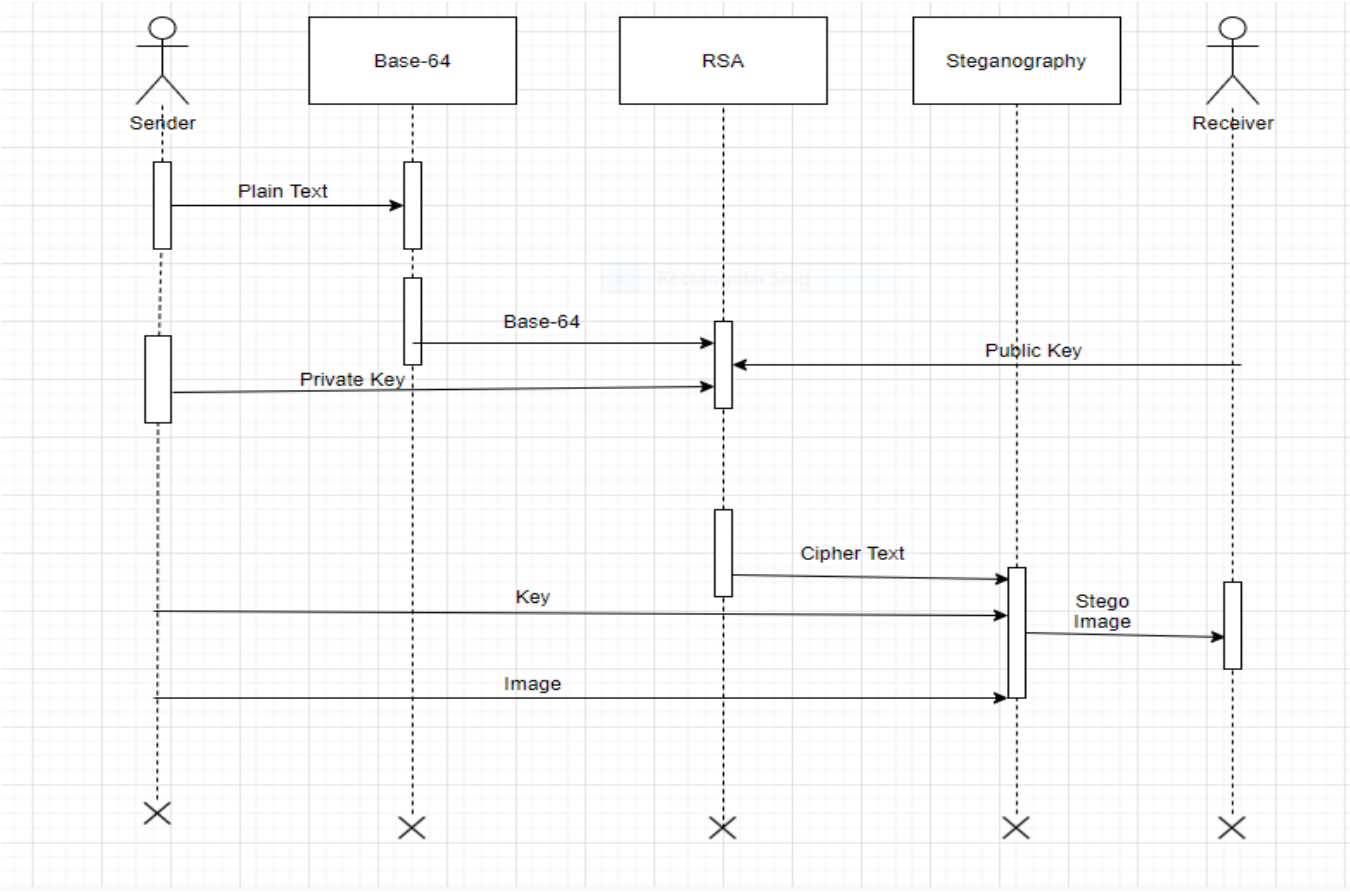


Fig 4.3 Sequential Diagram

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.

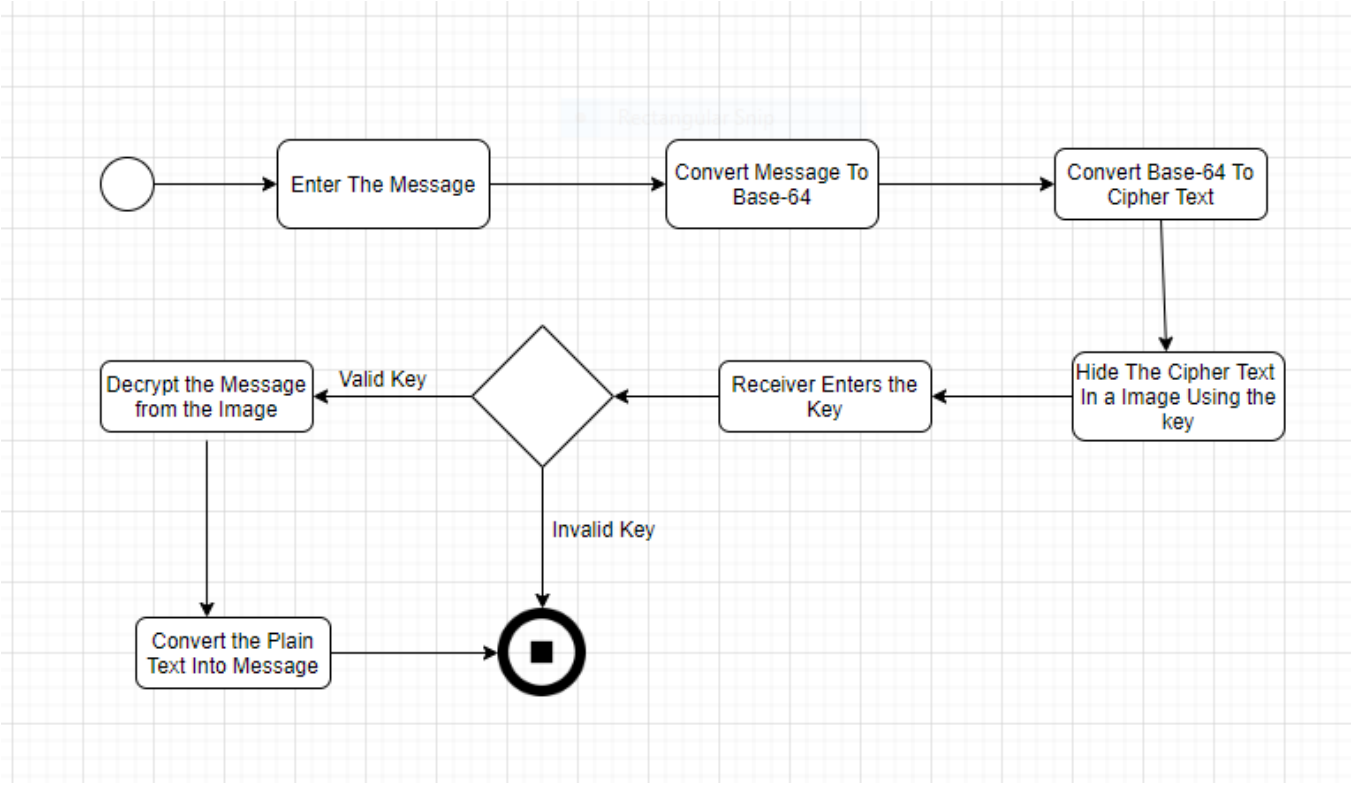


Fig 4.4 Activity Diagram

5. 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>
```

5.2.1.3 Connectivity

```
from flask import Flask, render_template, request


app = Flask(__name__)


@app.route('/')
def home():
    return render_template('home.html')


@app.route('/Encryption')
def encrypt():
    return render_template('Encryption.html')


@app.route('/Decryption')
def decrypt():
    return render_template('Decryption.html')


@app.route('/Encryption', methods=['POST'])
def getdata_enc():
    import os
    source_name = request.form['source_name']
    p = int(request.form['prime_1'])
    q = int(request.form['prime_2'])
    cover_name = request.form['cover_name']
    new_img_name = request.form['new_name']

    import base_enc
    base_enc.base_enc(source_name)

    import rsa_enc
    rsa_enc.call_rsa('s.txt', p, q)

    import stego_enc
    cover_name = os.path.dirname(os.path.abspath(__file__))+'/static/coverimages/'+cover_name
    stego_enc.encode(cover_name, new_img_name)
    return render_template('thank.html')
```

```
@app.route('/Decryption', methods=['POST'])
def getdata_dec():
    cover_name = request.form['cover_name']
    p = int(request.form['prime_1'])
    q = int(request.form['prime_2'])
    new_cover_name = request.form['new_cover_name']

    import stego_dec
    stego_dec.decode(cover_name)

    import rsa_dec
    rsa_dec.rsa_dec(p,q)

    import base_dec
    base_dec.base_dec(new_cover_name)

    return render_template('thank.html')


if __name__ == '__main__':
    app.run(debug=True)
```

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)
```

5.2.3 RSA

5.2.3.1 Encryption:

```
def convert(txt):
if (txt == "A"):
k = 1
elif (txt == "B"):
k = 2
elif (txt == "C"):
k = 3
elif (txt == "D"):
k = 4
elif (txt == "E"):
k = 5
elif (txt == "+"):
k = 74
elif (txt == "/"):
k = 75
```

```
elif (txt == "!"):
        k = 63
    elif (txt == "@"):
        k = 64
    elif (txt == "#"):
        k = 65
    elif (txt == "$"):
        k = 66
    elif (txt == "%"):
        k = 67
    elif (txt == "^"):
        k = 68
    elif (txt == "&"):
        k = 69
    elif (txt == "*"):
        k = 70
    elif (txt == "("):
        k = 71
    elif (txt == ")"):
        k = 72
    elif (txt == "-"):
        k = 73
    elif (txt == "+"):
        k = 74
    elif (txt == "/"):
        k = 75
else:
k = "ERROR"
return k
```

```
def revconvert(num):
    if (num == 1):
        k = "A"
    elif (num == 2):
        k = "B"
    elif (num == 3):
        k = "C"
    elif (num == 4):
        k = "D"
    elif (num == 5):
        k = "E"
    elif (num == 6):
        k = "F"
elif (num == 63):
        k = "!"
    elif (num == 64):
        k = "@"
    elif (num == 65):
        k = "#"
    elif (num == 66):
        k = "$"
    elif (num == 67):
        k = "%"
    elif (num == 68):
        k = "^"
    elif (num == 69):
        k = "&"
    elif (num == 70):
        k = "*"
```

```
    elif (num == 71):
        k = "("
    elif (num == 72):
        k = ")"
    elif (num == 73):
        k = "-"
    elif (num == 74):
        k = "+"
    elif (num == 75):
        k = "/"
    else:
        k = "Error"


    return k
def gcd(a, b):
if b == 0:
return        a
else:
return gcd(b, a % b)
if name ==" main ":
p = int(input('Enter the value of p = '))
q = int(input('Enter the value of q = '))
# Input Text.....

file = open("s1.txt","r")
text=file.read()
file.close()
lk = []
#text = input('Enter the value of text = ')
l1 = len(text)
k10 = ""
k20 = ""
for i in range(0, l1):
```

```
no = convert(text[i])
n = p * q
if (no > n):
print("Please enter correct text.....")
else:
t = (p - 1) * (q - 1)
for e in range(2, t):
if gcd(e, t) == 1:
break
for i in range(1, 10):
x = 1 + i * t
if (x%e==0)
d = int(x /e)
break
ctt=Decimal(0)
ctt = pow(no, e)
ct = ctt % n
lk.append(ct)
ct1 = ct % 75
print('n = ' + str(n) + ' e = ' + str(e) + ' t = ' + str(t) + ' d = ' + str(d) + ' cipher text = ' +
str(ct1)) k1 = revconvert(ct1)
k10 = k10 + k1
print("Cipher Value", k10)
print("Original Value : ",lk)
def get_lk():
        return lk
file = open("sample1.txt","w")
file.write(k10)
file.close()
file = open("sample2.txt","w")
for i in lk:
        file.write(str(i)+"      ")
file.close()
```

5.2.3.2 Decryption:

```
 def convert(txt):
if (txt == "A"):
k = 1
 elif (txt == "B"):
k = 2
 elif (txt == "C"):
k = 3
 elif (txt == "D"):
k = 4
 elif (txt == "E"):
k = 5
 elif (txt == "+"):
k = 74
 elif (txt == "/"):
k = 75
 elif (txt == "!"):
        k = 63
 elif (txt == "@"):
        k = 64
elif (txt == "#"):
        k = 65
elif (txt == "$"):
        k = 66
elif (txt == "%"):
        k = 67
elif (txt == "^"):
        k = 68
elif (txt == "&"):
```

```
        k = 69
elif (txt == "*"):
        k = 70
elif (txt == "("):
        k = 71
    elif (txt == ")"):
        k = 72
    elif (txt == "-"):
        k = 73
    elif (txt == "+"):
        k = 74
    elif (txt == "/"):
        k = 75
 else:
 k = "ERROR"
 return k

 def revconvert(num):
    if (num == 1):
        k = "A"
    elif (num == 2):
        k = "B"
    elif (num == 3):
        k = "C"
    elif (num == 4):
        k = "D"
    elif (num == 5):
        k = "E"
    elif (num == 6):
```

```
        k = "F"
    elif (num == 63):
        k = "!"
    elif (num == 64):
        k = "@"
    elif (num == 65):
        k = "#"
    elif (num == 66):
        k = "$"
    elif (num == 67):
        k = "%"
    elif (num == 68):
        k = "^"
    elif (num == 69):
        k = "&"
    elif (num == 70):
        k = "*"
    elif (num == 71):
        k = "("
    elif (num == 72):
        k = ")"
    elif (num == 73):
        k = "-"
    elif (num == 74):
        k = "+"
    elif (num == 75):
        k = "/"
    else:
        k = "Error"
```

```
    return k


def gcd(a, b):

if b == 0:

return         a
else:
return gcd(b, a % b)

p = int(input('Enter the value of p = '))
q = int(input('Enter the value of q = '))
n = p * q
file=open("sample2.txt","r")
s=file.read()
ct=list(map(int,s.split()))

for i in range(0, len(ct)):
t = (p - 1) * (q - 1)
for e in range(2, t):

if gcd(e, t) == 1:

 break

for j in range(1, 10):

           x = 1 + j * t
           if(x%e ==0)
          d  =  int(x  /  e)
           break
dtt  =Decimal(0)
print(ct[i])
dtt  =  pow(ct[i],  d)
dt = dtt % n
print('n = '+str(n)+' e = '+str(e)+' t = '+str(t)+' d = '+str(d)+'decrypted text = '+str(dt))
k2 = revconvert(dt)
k20 = k20 + k2

print("Original Message: ", k20)
```

5.2.4 Steganography

5.2.4.1 Encryption :

```
from PIL import Image

def genData(data):
    # list of binary codes
    newd = []

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


def modPix(pix, data):
    datalist = genData(data)
    lendata = len(datalist)
    imdata = iter(pix)

    for i in range(lendata):
    pix = [value for value in imdata._next_()[:3] + imdata.__next__()[:3] +
            imdata.___next___()[:3]]


        for j in range(0, 8):
            if (datalist[i][j] == '0') and (pix[j] % 2 != 0):

                if (pix[j] % 2 != 0):
                    pix[j] -= 1

            elif (datalist[i][j] == '1') and (pix[j] % 2 == 0):
            pix[j] -= 1

        if (i == lendata - 1):
            if (pix[-1] % 2 == 0):
                pix[-1] -= 1
        else:
            if (pix[-1] % 2 != 0):
                pix[-1] -= 1
```

```
        pix = tuple(pix)
        yield pix[0:3]
        yield pix[3:6]
        yield pix[6:9]


def encode_enc(newimg, data): w
    = newimg.size[0]
    (x, y) = (0, 0)
    for pixel in modPix(newimg.getdata(), data):
        newimg.putpixel((x, y), pixel)
        if (x == w - 1):
            x = 0
            y += 1
        else:
            x += 1


def encode():
    img = input("Enter image name(with extension) : ")
    image = Image.open(img, 'r')
    file = open("sample1.txt", "r")
    data = file.read()
    file.close()
    print(len(data))


    if (len(data) == 0):
        raise ValueError('Data is empty')

    newimg = image.copy() encode_enc(newimg,
    data)

    new_img_name = input("Enter the name of new image(with extension) : ")
    newimg.save(new_img_name, str(new_img_name.split(".")[1].upper()))




def main():
    a = print(":: Welcome to Steganography ::\n")
```

```
    encode()



if__name____== '__main__':
    # Calling main function
    main()
```

5.2.4.2 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:
                binstr += '1'

        data += chr(int(binstr, 2)) if
        (pixels[-1] % 2 != 0):
            k = data
            print(k)
            print(len(k))
```

```
        return data


def main():
    print(":: Welcome to Steganography ::\n") sq
    = decode()
    print("Decoded word- " + sq)
    file = open("s09.txt", "w")
    file.write(sq)
    file.close()


if__name___== '__main__':
    # Calling main function
    main()
```

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	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.

5.4 RESULTS

Fig 5.4.1 : Home Page

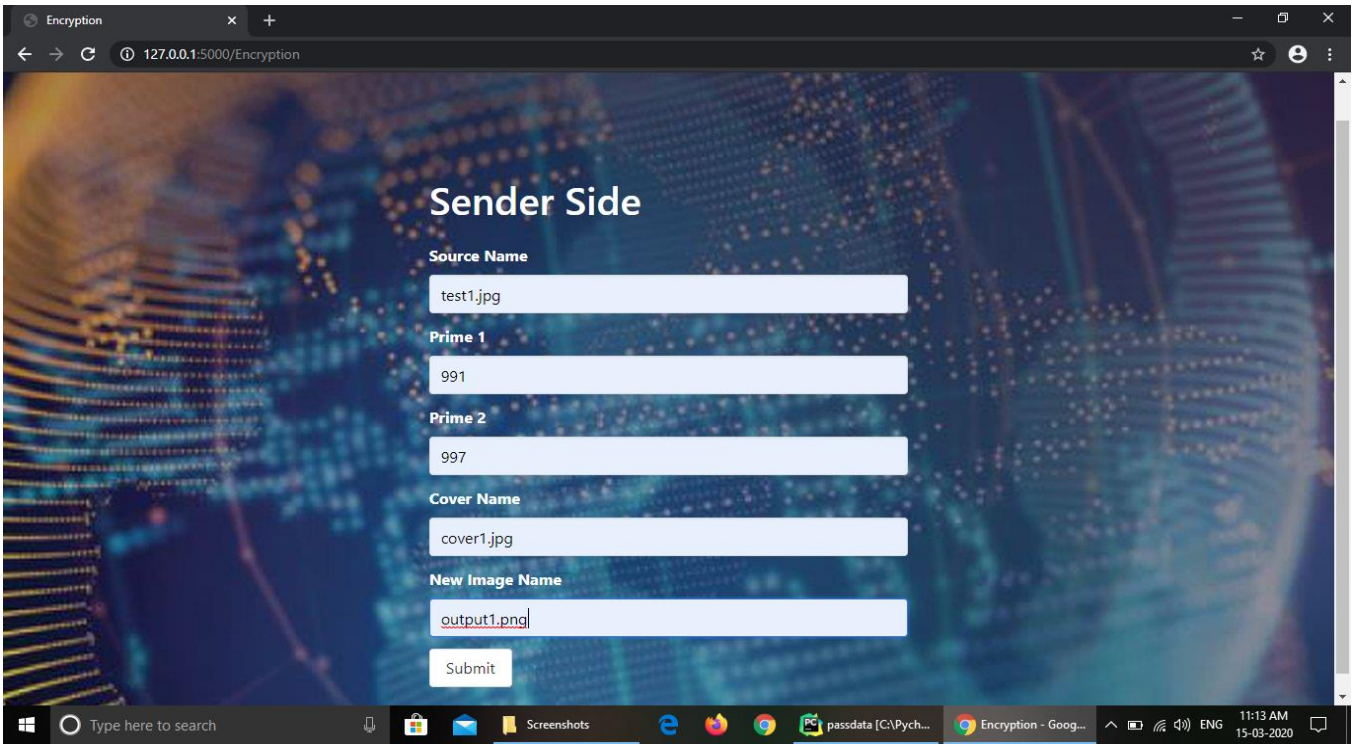


Fig 5.4.2 : Sender Side

Fig 5.4.3 : Reciever Side

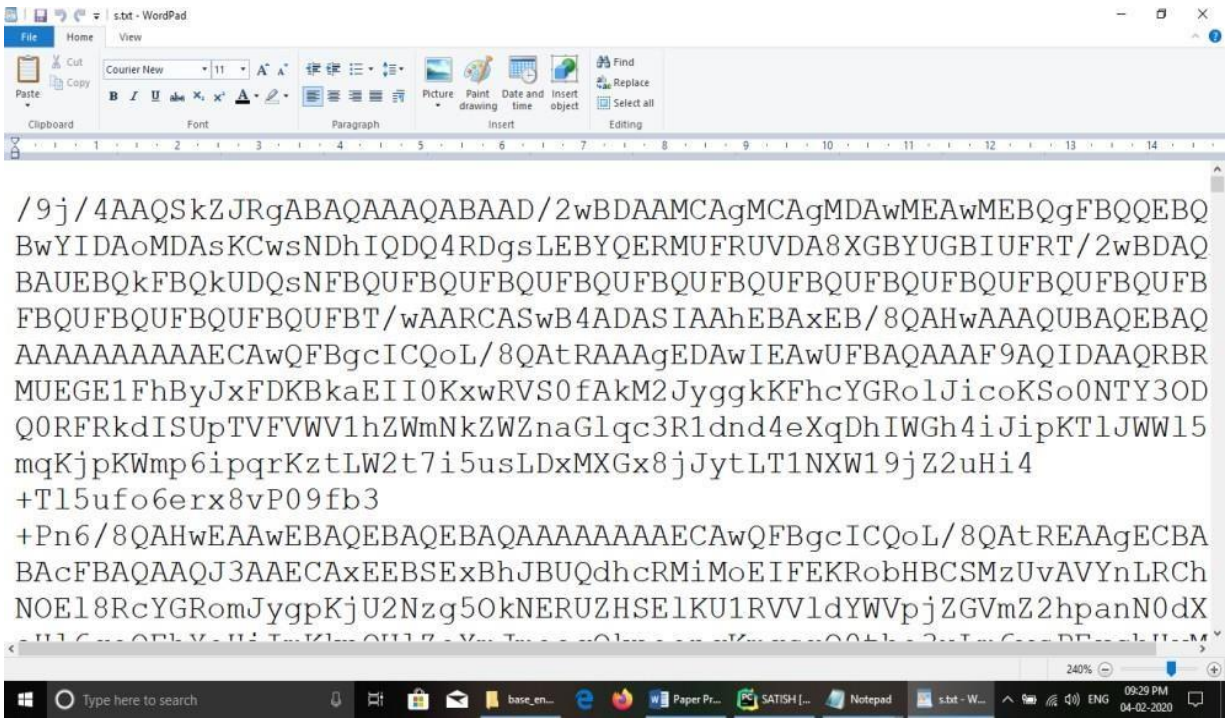


Fig:5.4.4 Image Encryption

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%
```

Fig.5.4.6 Audio Encryption

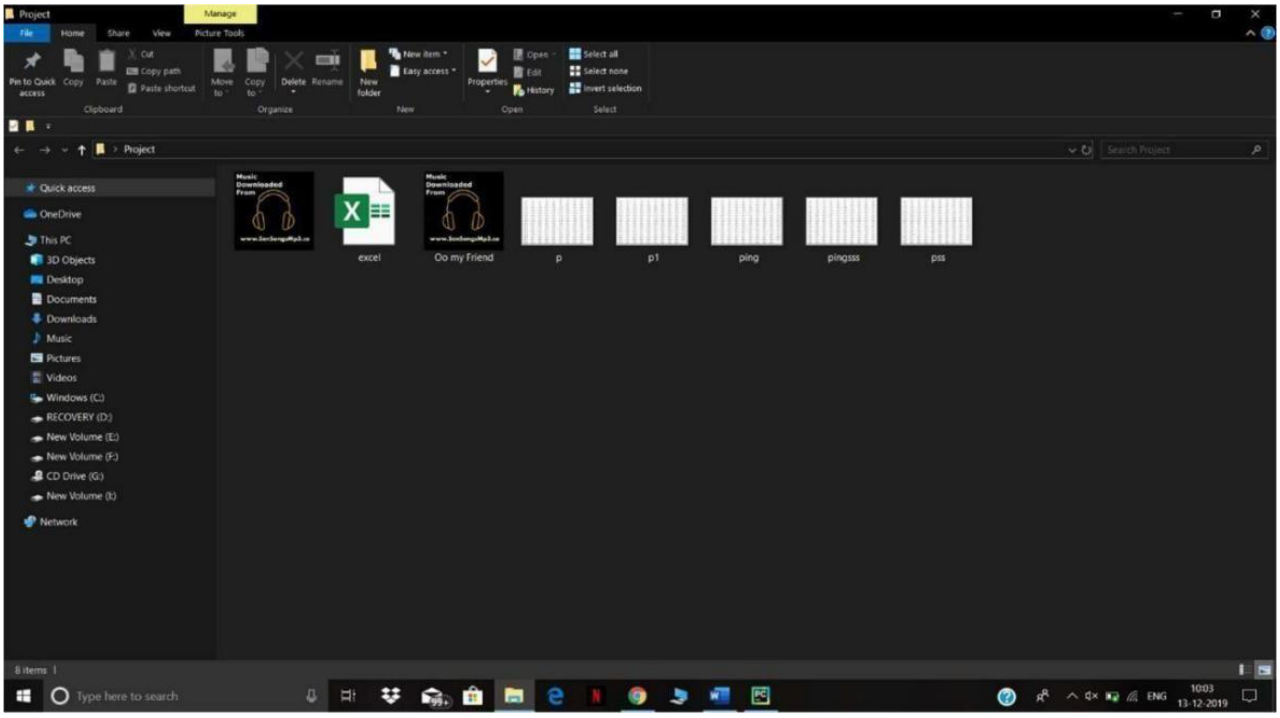


Fig. 5.4.7 Audio Decryption Folder

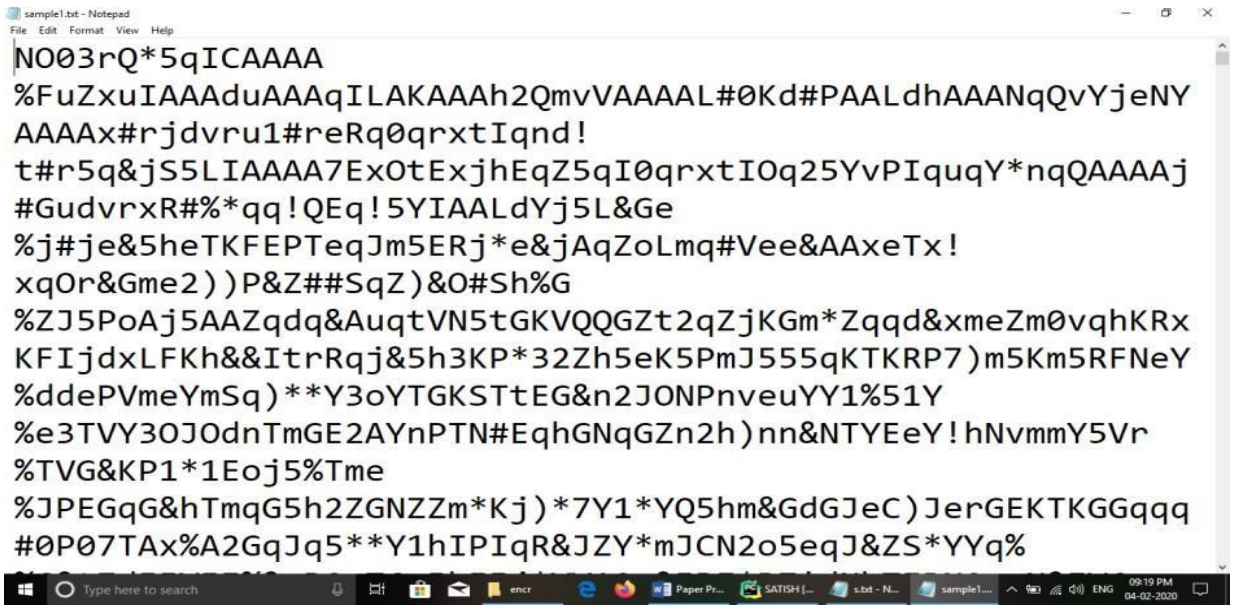


Fig 5.4.8 RSA Encryption

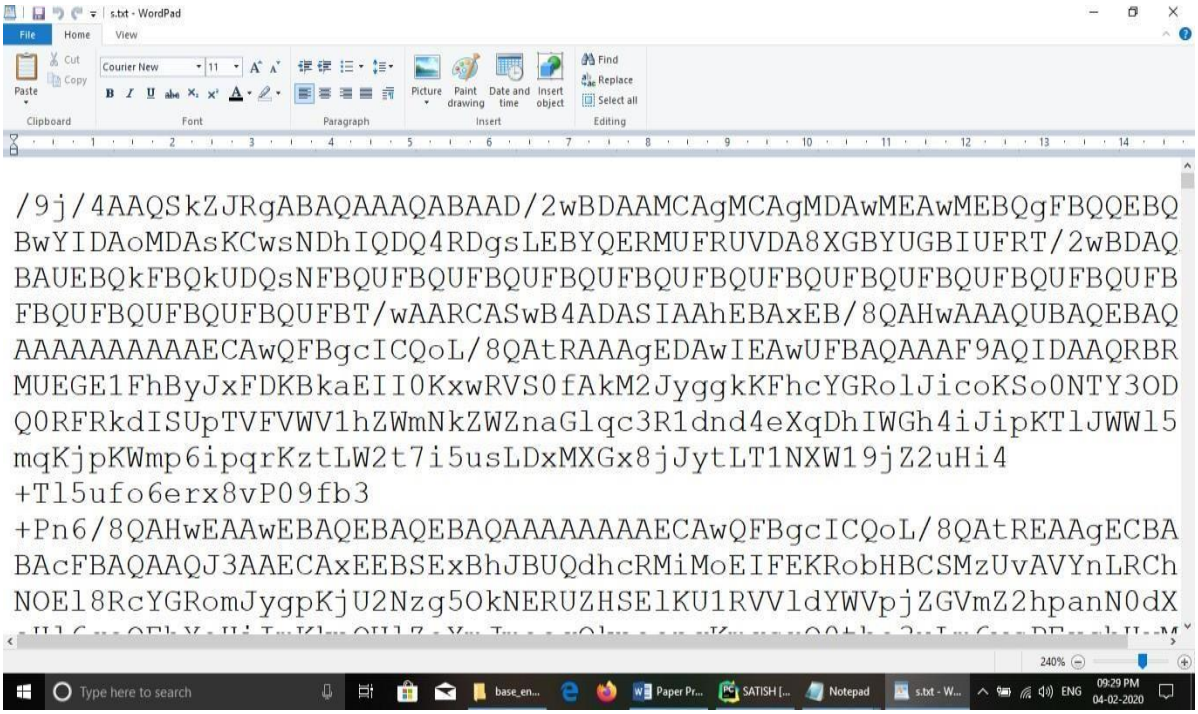


Fig 5.4.9 RSA Decryption

+

=

Fig 5.4.10 Stego-Object

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.

[5] M. K. I. Rahmani and N. P. Kamiya Arora, "A crypto-steganography: A survey," International Journal of Advanced Computer Science and Application, vol. 5, pp. 149–154, 2014.

[6] Mr. Vikas Tyagi(2012), "Data Hiding in Image Using least significant bit with cryptography", International Journal of Advanced Research in computer science and Software Engineering, Volume 2, Issue 4.

[7] P. R. Ekatpure and R. N. Benkar, "A comparative study of steganography & cryptography," 2013.

[8] R. Poornimal and J. Iswarya (2013) "An Overview of Digital Image Steganography", International Journal of Computer Science & Engineering

Survey Vol.4,NO.1,February.

[9] R Praveen Kumar, V Hemanth, MShareef, Securing Information Using Sterganoraphy, 2013 International Conference on Circuits, Power and Computing Technologies.

Secure Data Hiding using Elliptical Curve Cryptography and Steganography

Hemanta Kumar Mohanta
M. Tech, CST, GITAM University
Visakhapatnam, India

ABSTRACT
Now these days information are passing by internet. Hence the security of information has become a fundamental issue. Cryptography is the well-known technique to secure data over network. Steganography is the technique to hide the message in digital media. The elliptical curve cryptography is more secure than the existing cryptography models. This paper describes a proposed hybrid model using public key Elliptical Curve Cryptography (ECC) and Steganography. Which provide more security than a Single ECC or Steganography methods. The main aim of this project is to hide crucial information of internet users, military, different corporate sectors those which are frequently using public network for communication.

Keywords
Cryptography, Steganography, ECC, RGB, LSB, CNOT gate and PSNR

1. INTRODUCTION
The network security is becoming more important as the number of data being exchanged on the Internet increases. Therefore, the confidentiality and data integrity are required to protect against unauthorized access. This has resulted in an explosive growth of the field of information hiding.

Cryptography is the process by which the data to be transmitted is hidden in a manner such that only the intended recipient can understand it. The initial data is called as plaintext and the encrypted data is called as cipher text. A key is used to hide the data [1]. There are different types depending on the number and way in which the keys are used.

There are two types of cryptographic techniques:

(i) Symmetric key cryptography

(ii) Asymmetric key cryptography

Symmetric Key Cryptography is actually the technique by which identical cryptographic keys are used for the purpose of both encryption and decryption. The receiver can get back original data by using the key. The symmetric key cryptography provides high data rates, usage as primitives to construct various cryptographic mechanisms and can be combined to produce stronger ciphers. The main fact here is that the security of data depends on the security of the key. So, care should be taken while exchanging keys between the sender and the receiver [2].

Symmetric cryptosystem have a problem of key transportation. The secret key is to be transmitted to the receiving system before the actual message is to be transmitted. Every means of electronic communication is insecure as it is impossible to guarantee that no one will able to tap communication channels. So the only secure way of exchanging keys would be exchanging personally. Symmetric cryptosystem can't provide digital signatures that can't be repudiated [3].

Asymmetric key cryptography is the technique where two keys are used. One key is used to lock or encrypt the plaintext, and another to unlock or decrypt the cipher text. Neither key can do both the functions. One of these keys is published or made public and the other is kept private. This technique has comparatively slower data rate throughputs than the symmetric key technique [2].

Steganography is the art and science of hiding information such that its presence cannot be detected. A secret information is encoded in a manner such that the very existence of the information is concealed. Paired with existing communication methods, Steganography can be used to carry out hidden exchanges.

This proposed hybrid model is a combination of Elliptical curve cryptography (ECC) and Steganography. As per previous study, key size of ECC is very less in comparison to RSA [11]. The comparison between ECC and RSA algorithms are stated in table 1. Using Steganography we can send multiple messages inside a cover image. The proposed model is described in section 3 and the experimental result is in section 4.

Table 1. Comparison between ECC and RSA

ECC key size in bits	RSA key size in bits
106	512
112	768
132	1024
160	2048
210	3072
283	7680

2. RELATED WORK
2.1 Elliptical Curve Cryptography
Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields [13].

2.1.1 Different Operation on Elliptic Curve
Let E be the elliptic curve over finite field P over equation $y^2 = x^3+ax+b$ and satisfy $4a^3+27b^2 \neq 0 \pmod{p}$. The operations are point addition, point doubling and scalar multiplication.

2.1.1.1 Point Addition:
1. let $A(x_1,y_1)$ point ∞ is the point at infinite are in E(P)

$$A(x_1,y_1)+\infty = \infty + A(x_1,y_1) = A(x_1,y_1) . \quad (1)$$

2. let $A(x_1,y_1)$ and $B(x_2,y_2)$ are Two points and the resultant

point is $R(x_3,y_3)$ for all points in E(P)

$A(x_1,y_1)+B(x_2,y_2)=R(x_3,y_3)$

Where $x_3=\left(\frac{y_2-y_1}{x_2-x_1}\right)^2-x_1-x_2$, and $y_3=\left(\frac{y_2-y_1}{x_2-x_1}\right)(x_1-x_3)-y_1$ (2)

2.1.1.2 Point Doubling

let $A(x_1,y_1)$ be the point in E(P) then

$2A=R(x_3,y_3)$

Where $x_3=\left(\frac{3x_1^2+a}{2y_1}\right)-2x_1$ and $y_3=\left(\frac{3x_1^2+a}{2y_1}\right)-(x_1-x_3)-y_1$ (3)

2.1.1.3 Point Subtraction

Let $A(x_1,y_1)$ and $B(x_2,y_2)$ are Two points and the resultant point is $R(x_3,y_3)$ for all points in E(P)

$R(x_3,y_3) = A(x_1,y_1) –B(x_2,y_2) = A(x_1,y_1) +\{– B(x_2,y_2)\}$

$= A(x_1,y_1) +B(x_2, –y_2)$

For any point $–A\ (x_1,y_1) = A(x_1, –y_1)$

2.1.1.4 Point Multiplication

Let A be any point on the elliptic curve (E). Then the operation multiplication of the point A is defined as repeated addition. kA = A+ A + ...k times.

Where k the integer in the field P.

2.1.2 The ElGamalcryptosystem [4] using an elliptic curve over F (p) or F (2^n)

A. Generating public and private keys

1. Bob chooses E (a, b) with an elliptic curve over F (p) or F (2^n).
2. Bob chooses a point on the curve, $e_1(x_1, y_1)$.
3. Bob chooses an integer d.
4. Bob calculate $e_2(x_2, y_2)$ =d × $e_1(x_1, y_1)$.Note that multiplication here means multiple addition of points.
5. Bob announce E (a, b), $e_1(x_1, y_1)$, and $e_2(x_2, y_2)$ as his public key;
6. Bob keeps d as his private key.

B. Encryption

1. Alice select P, a point on the curve, as her plaintext, P.
2. She then calculates a pair of points on the text as cipher texts:
   - I. C_1= r ×e_1
   - II. C_2= P + r ×e_2

C. Decryption

1. Bob after receiving C_1 *and* C_2 ,calculates P, the plaintext using the following formula.
   - P = C_2- (d× C_1)

2.2 Steganography

The simplest approach to hiding data within an image is called least significant bit (LSB) insertion [5][14]. For 24-bit true color image, the amount of changes will be minimal and indiscernible to the human eye. As an example, suppose that we have three adjacent pixels (nine bytes) with the following RGB encoding:

10010101	00001101	11001001
10010110	00001111	11001010
10011111	00010000	11001011

Now suppose we want to hide the following 9 bits of data101101101. If we overlay these 9 bits over the LSB of the 9 bytes above, we get the following (where bits in bold have been changed) pixels:

10010101	0000110**0**	11001001
1001011**1**	0000111**0**	1100101**1**
10011111	00010000	11001011

The following formula provides a very generic description of the pieces of the steganographic process:

Stego-image = cover image + information

Information maybe text OR image etc.

2.2.1 CNOT Gate

CNOT gate is also called as Controlled not gate [7][12]. It comes under quantum computer. It is essential for constructing a quantum computer. Inside the CNOT gate, first qbit is control bit and the second bit is a target bit [7].

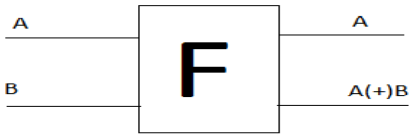


Figure 1: CNOT gate

Where, A be the control qbit, B be the target qbit, and (+) represents as EXOR. CNOT gate is completely different from the EX - OR gate. The EX-OR gate is irreversible gate, then the CNOT gate is reversible gate [7].

2.2.2 Wavelet Transformation

Wavelet compressions are two types lossless or lossy[12]. In lossless compression, the original data can be reconstructed from the compressed data, but in lossy compression the partial data can be reconstructed. Using wavelet transformation the data can be stored in less space, by doing so the memory space will be reduced and the data can be transferred easily [8]. Steps in wavelet compression: Load the image, perform wavelet decomposition of the image, and compress using fixed threshold.

2.2.3 Random Number Generators

Blum Blum Shub generator, is the pseudo random number generator [12]. By using this random numbers are generated. The formula has shown below [9],

$$X_{i+1} = (X_i)^2 \ mod\ n \quad (4)$$

Where, X_i is the seed, and n be the range.

The pseudo random bit generator is used for generating random numbers in cryptography. Seed, two large prime numbers, and the range is the inputs for the pseudo random bit generators. The mathematical formulae has shown below,

$$X_{i+1} = (PX_i + Q)^2 \ mod \ n \quad (5)$$

Where P, Q are two large prime numbers, X_i is the seed. n be the range.

2.2.4 Encryption Algorithm

The secret information may be in any form like text, image etc. is compressed by wavelet transforms [10]. The compressed text is converted into its corresponding ASCII value, next the ACSII is converted into its 8-bit binary value. By using Control NOT gate, the 8-bit binary value is encoded. Now these bits are ready to embedded into an image using LSB insertion. The encrypted message is ready to be embedded in the cover image. Before embedding the message, the image is converted into its corresponding pixel values. These values are arranged in the r × c matrix form, r and c represent rows and columns respectively. The bit of the secret information has to be embedded in the random positions in the cover image. To identify the random positions, Random number generator is used. Random numbers act like a key in this technique. Blum/blum/shub generator and Pseudo random generator are used to se1ect the random rows and columns respectively. Random numbers are generated by the generator, using the key (seed). Randomness will be varying from generator to generator. The randomness is achieved by padding the bits in the sequence. After selecting the random positions in the image (pixel values) now the secret message is embedded in the corresponding bits using the LSB insertion technique.

2.2.5 Decryption Process

Decryption is the repeal process of the encryption process[10]. After receiving the stego image, the receiver will convert the image into its corresponding pixels (matrix form). With the help of Key (seed) the receiver will be generating the random number using the random generators to identify in which positions the bits have been embedded. After getting the pixel positions, applying reverse LSB insertion technique will give the encoded bits. Applying the Control NOT gates on the encoded bits, the compressed text is retrieved. By applying wavelet, transformation technique (decompression) the original secret information is retrieved.

3. PROPOSEDMODEL

In the proposed model elliptic curve parameters are(p, E, P, n,) where p is the prime number F_p denoted as field of integers modulo p. E is the elliptic curve over F_p is defined by the equation $y^2=x^3+ax+b$ where (a,b) are the real numbers over F_p and satisfy $4a^3+27b^2 \neq 0$ (mod p). The point infinity ∞ is also in the curve. The abelian subgroup of $E(F_p)$ generated by p is

P={∞, *P,2P,3P,……(n-1)P*}

3.1 Key generation

Input:

Elliptic Curve Domain parameters (p, E, P, n,)

Output:

Public key Q and private key d.

1. Select $d \in_R [1, n-1]$
2. Select $e_1(x_1, y_1)$.
3. Compute $e_2(x_2, y_2) = d \times e_1(x_1, y_1)$
4. Q = $\{e_1, e_2, E\}$
5. Return(Q, d)

3.2 Encryption and LSB Embedding

INPUT:

Elliptic Curve Domain parameters (p, E, P, n), public key Q, Plaintext m, message image I, Cover image C.

OUTPUT: Stego-image CI, Stego key

1. Represent the message 'm' as a point M in E (F_p).
2. Select $K \in_R [1, n-1]$.
3. Compute $C_1 = k \times e_1(x_1, y_1)$
4. Compute $C_2 = M + k \times e_2(x_2, y_2)$.
5. RGB cover image=C.
6. Hide (C_1, C_2) into I using LSB Steganography.
7. Hide I into C using Steganography.
8. Return (CI)

3.3 Decryption

INPUT:

Elliptic Curve Domain parameters (p, E, P, n), Private key d, stego-image CI, stego key.

OUTPUT: (message m, image I)

1. Extract I from CI Extract C_1, C_2 from I
2. Compute $M = C_2 - d \times C_1$ and compute m from M
3. Return (m, I)

3.4 Background work

Consider the elliptical curve is $y^2 = x^3 + 4x + 20$ over finite field F_{29}

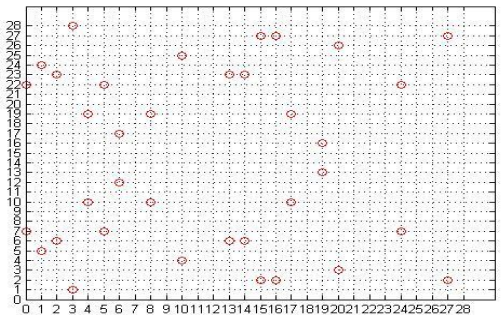


Figure 2. Points on the curve $y^2 = x^3 + 4x + 20$ mod 29

Points are[(∞),(1,5),(4,19),(20,3),(15,27),(6,12),(17,19), (24, 22),(8,10),(14,23),(13,23),(10,25),(19,13),(16,27),(5,22)(3,1),(0,22),(27,2),(2,23),(2,6),(27,27),(0,7),(3,28),(5,7),(6,2),(19,16),(10,4),(13,6),(14,6),(8,19),(24,7)(17,10),(6,17),(15,2),(20,26),(4,10),(1,24),(20,3)]

Table 2: Different points in elliptical curve mask different characters.

∞	1,5	4,19	20,3	15,27	6,12	17,19
a	b	c	d	e	f	g
24,22	8,10	14,23	13,23	10,25	19,13	16,27
h	i	j	k	l	m	n
5,22	3,1	0,22	27,2	2,23	2,6	27,27
o	p	q	r	s	t	u
0,7	3,28	5,7	6,2	19,16	10,4	13,6
v	w	x	y	z	0	1
14,6	8,19	24,7	17,10	6,17	15,2	20,26
2	3	4	5	6	7	8
4,10	1,24	20,3				
9	space	.				

3.4.1 Key generation

For elliptic curve $y^{2} = x^{3} + 4x + 20$ over finite field F_{29}

1. Bob chooses E (a, b) with an elliptic curve over F_p a =4, b=20,p=29
2. Bob chooses a point on the curve, $e_1(x_1, y_1)$.

   let(1,5)
3. Bob chooses an integer d. Let d = 3
4. Bob calculate $e_2(x_2, y_2)$ =d × $e_1(x_1, y_1)$.Note that multiplication here means multiple addition of points.$e_2(x_2, y_2)$=(20,3)
5. Bob announce E (a, b), $e_1(x_1, y_1)$, and $e_2(x_2, y_2)$, p as his public key;E(4,20),e_1(1,5),e_2(20,3)
6. Bob keeps d as his private key.

3.4.2 Encryption

INPUT :(cover image C, image I, message 'm')

OUTPUT: (stego object CI)

1. Alice selects P, a point on the curve, as her plaintext, P. EXAMPLE Message=hello P1=(24,22),P2=(15,27),P3=(10,25),P4=(10,25), P5=(5,22)
2. She then calculates a pair of points on the text as cipher texts:

   III. $C_1 = r \times e_1$

   IV. $C_2 = P + r \times e_2$

Table 3: Encryption process of above example

points	r	$C_1 = r \times e_1$	$C_2 = P + r \times e_2$	(C_1, C_2)
h=P1(24,22)	2	(4, 19)	(13, 23)	(c, k)
e=P2(15,27)	6	(17, 19)	(3, 28)	(g, w)
l=P3(10,25)	5	(6, 12)	(10, 4)	(f, 0)
l=P4(10,25)	15	(3,1)	(2,6)	(p ,t)
o=P5 (5,22)	3	(20, 3)	(5, 7)	(d, x)

3. Separate all C1, C2 and make array of

   C1* ∈{$C1_1$, $C1_2$, $C1_3$……}

   C2* ∈{$C2_1$, $C2_2$, $C2_3$……} respectively.
4. Embedded C1* and C2* into the image I
5. Choose one cover image C
6. CI = Embedded I into C

3.4.3 Decryption

INPUT: stego-object, stego key

OUTPUT: message 'm'

To extract the cipher text and image from stego object ,the stego key is used that used to construct a stego object.

To get the text message from cipher text we use private key that is

$M = C_2 - d \times C_1$

Consider the above example

Table 4: Decryption process of the example

(C_1, C_2)	$C_2 - d \times C_1$	P	M
(c, k)	(13, 23) – 3 × (4, 19)	(24, 22)	h
(g, w)	(3, 28) – 3 × (17, 19)	(15, 27)	e
(f, 0)	(10, 4) – 3 × (6, 12)	(10, 25)	l
(p ,t)	(2,6) – 3 × (3,1)	(10,25)	l
(d, x)	(5, 7) – 3 × (20, 3)	(5, 22)	o

Output message m = (hello)

4. EXPERIMENTAL RESULT

The above stated hybrid method was applied to the message as shown in figure (3).The cover image used for this process is shown in figure (4). Total process of the entire method is shown in figure (5).

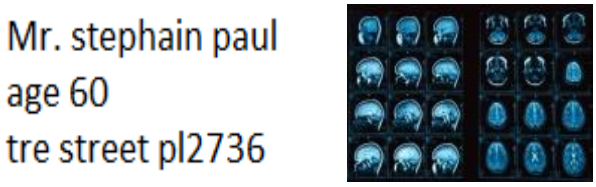


Figure 3. Patient information and brain MRI scan

Figure 4. Cover image (animal.jpg)

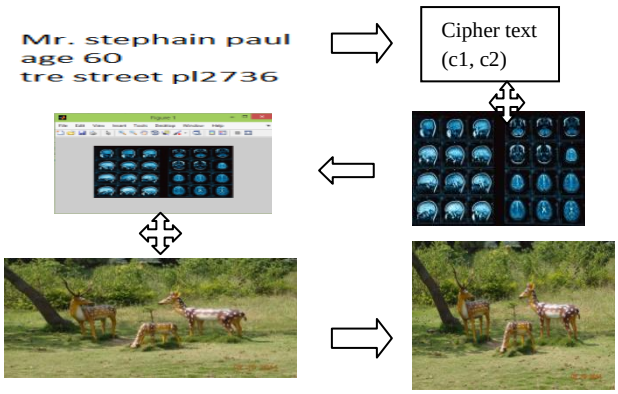


Figure5. Total process of proposed model

The cover image is the main image in which the hidden information will be embedded. The resultant image is the stego image which is the same type of image as the cover image. To measure the quality of stego image, Peak Signal-to-Noise Ratio (PSNR) is calculated. PSNR is a statistical measurement used for digital image or video quality assessment [6]. PSNR is most easily defined via the mean squared error (MSE) which for two m×n monochrome images I and K where one of the images is considered a noisy approximation of the other is defined as:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - k(i,j)]^2 \qquad (6)$$

The PSNR is defined as:

$$PSNR = 10\log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20\ \log_{10}\left(\frac{MAX_1}{\sqrt{MSE}}\right) \qquad (7)$$

Larger PSNR indicates better quality of the image or in other terms lower distortion. The larger the PSNR value the smaller the possibility of visual attack by human eye.

Table 3 Represent the PSNR value after embedding different size of input data and image into cover image.

Table5. PSNR table

Cover image size(pixel)	input		PSNR
	Text(bytes)	Image(pixel)	
800x600	84	294x184	56.3741
800x600	168	294x184	56.3747
800x600	168	256x256	55.5312
800x600	848	256x256	55.5314
800x600	2539	256x256	55.5310

5. CONCLUSION

The proposed model introduced above is a combination of cryptography and Steganography. The goal of the technique is to put the unauthorized person in a difficult position to determine the presence of information. The dual security makes the information more secure. With this model any one can easily send multiple information to the receiver using public network. This model is very useful for defense, corporate, banking, communication and different government portals where information exchange is more crucial. The data hiding capacity in audio and video is more than image, so in future using audio or video steganography and cryptography huge amount of data will transmit in public network without security violence.

6. ACKNOWLEDGEMENT

The author would like to thank Ms. J. Hyma, assistant professor, cse department, GITAM University, for various help and guidance.

7. REFERENCES

[1] M. M Amin, M. Salleh, S .Ibrahim, M.R.K atmin, and M.Z.I.Shamsuddin, Information Hiding using Steganography, National Conference on Telecommunication Technology Proceedings, Shah Alam, Malaysia, 2003 IEEE.

[2] S Ushll , G A SathishKumal, K Boopathybagan,A Secure Triple Level Encryption Method Using Cryptography and Steganography, 20 II International Conference on Computer Science and Network Technology, 978-1-4577-1587-7/111$26.00 ©20111EEE, December 24-26, 2011

[3] X. Zhang and S. Wang, Steganography using multiple-base notational system and human vision sensitivity, IEEE Signal Process. Lett., vol.12, no. I, pp. 67-70, Jan. 2005.

[4] BehrouzA.Forouan, Debdeep Mukhopadhyay, 2nd edition Cryptography and network security, McGraw Hill Education, pp.295-296

[5] S. M. Masud Karim, Md. Saifur Rahman, Md. Ismail Hossain, A New Approach for LSB Based Image Steganographyusing Secret Key987-161284-908-9/11/$26.00 2011 IEEE

[6] M. Hossain, S.A. Haque, F. Sharmin, Variable RateSteganography in Gray Scale Digital Images Using Neighborhood Pixel Information, Proceedings of 200912th International Conference on Computer and Information Technology (ICCIT 2009) 21-23 December 2009, Dhaka, Bangladesh.

[7] Controlled NOT gate, From Wikipedia, http://en , wikipedia . org/wiki/Controlled_NOT _gate .

[8] Ivan W. Selesniek "Wavelet Transforms A Quick Study", Physies Today magazine, üetober, 2007.

[9] "Blum Blum Shub", From Wikipedia, http://en.wikipedia.org/wiki/Bluffi_Bluffi_Shub

[10] R Praveen Kumar, V Hemanth, MShareef, Securing Information Using Sterganoraphy, 2013 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2013]

[11] Ipsita sahoo , SEMINAR REPORT SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS http://www.facweb.iitkgp.ernet.in/~isg/ICTSEMINAR/REPORT-Ipsita.pdf

[12] M Venkteswara Reddy, M Lakshman Naik, Securing Information Using Steganography, International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064

[13] Darrel Hankerson, Alfred Menezes, Scott Vanstone, Guide to elliptic curve cryptography, springer

[14] Ahaiwe J. Document Security within Institutions Using Image Steganography Technique, International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064

SECURE DATA ENCRYPTION AND DECRYPTION USING CRYPTO-STEGO

[1]Matcha Venkatesh, [2]T.Anitha, [3]G.Satish, [4] M.Sudarshan, [5]K.Ram Sudeep
[1]Student, [2]Assistant Professor, [3,4,5]Student
[1]Department of Computer Science & Engineering,
[1]Anil Neerukonda Institute Of Technology And Sciences, Visakhapatnam, India

Abstract : Securing data encryption and decryption using Cryptography and Steganography techniques. This paper introduces a new kind of approach for covert communications between two private parties. The approach introduced in this paper makes use of both steganographic as well as cryptographic techniques. In Cryptography we are using Rivest-Shamir-Adleman (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. We are using RSA for encryption of data and Steganography concept to hide the data in an image. Such that any other person in the network cannot access the data. Only the sender and receiver can retrieve the message from the data.

Index Terms : **Rivest-Shamir-Adleman(RSA), Cryptography, Steganography.**

1. INTRODUCTION

Digital communication witnesses a clear and continuous development in many applications within the Internet. Hence, secure communication sessions must be provided. The security of data that is transmitted across a world wide network has become a key factor on the network performance measures. So, the confidentiality and integrity of transmitted data are needed to stop from accessing and using transmitted data. Steganography and Cryptography are 2 techniques that are provided for network security. The aim of this paper is to develop an approachto cover secret data in an image, by taking advantage of combination of cryptography and steganography.

Cryptography

Cryptography is one of the secure method used to guarantee the privacy of communication between parties. This method is used for 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 are often decrypted to the given plaintext. Cryptography provides a secure communication across an insecure channel, like: confidentiality, privacy, non-repudiation, key exchange, and authentication. There are two kinds of Cryptography techniques :

i) Symmetric / Secret Key Cryptography
ii) Asymmetric / Public Key Cryptography

Symmetric / Secret Key Cryptography

The other name for Secret key encryption is symmetric-key, shared key, single-key, and eventually private-key encryption. By using the key we can encrypt the given plain text, similary by using the same key at receiver side we can decrypt the message to obtain the plaintext. The key will be known only by a people who are authorized to the encryption or decryption.

Asymmetric / Public Key Cryptography

We can call this technique as asymmetric cryptography or public key cryptography, here we use two keys which are mathematically associated, used 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 public key, and the decryption key is private key.

Steganography

It can be defined as the science of concealment 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 cover image. If an individual views the given cover which the information is hidden inside of he or she will have no clue that there is any covering data. The proposed model is a combinational of Rivest-Shamir-Adelman(RSA) and Image Steganography.

2. RELATED WORK

2.1 Rivest-Shamir-Adelman

Rivest-Shamir-Adelman(RSA) is an approach to public-keyCryptogrophy.

2.1.1 Different Operations on RSA

Let P and Q be the two prime numbers. The operations are "e" and "d". Product of P and Q is considered as "n". Product of (P-1) and (Q-1) is considered as $\phi(n)$.

2.1.1.1 Public Exponent

let "e " be the public Exponent, it can be calculated using GCD between e and $\phi(n)$, e values varies from 1 to n. If Gcd of e and $\phi(n)$ is "1" then that value can be considered as "e".

2.1.1.2 Secret Exponent

let "d" be the secret exponent and ed mod $\phi(n) = 1$. The Extended Euclidean is based on the formula $\gcd(e, \phi(n)) = 1$, where d should be equal to $s + \phi(n)$ in order to satisfy the ed mod $\phi(n) = 1$ condition.

2.1.2 RSA algorithm

a) Key Generation

Inputs : 2 Prime Numbers p and q.

1. Select p and q such that both are the prime numbers, $p \neq q$.
2. Calculate n=p*q,
   (n) -> Euler's totient function
3. Calculate q(n) = (p-1)*(q-1)
4. Select an integer e such that gcd (q(n), e)) =1 & 1< e < (n)
5. Calculate d; d= e-1 mod (n)
6. Public Key, PU= {e, n}
7. Private Key, PR ={d,n}

b) Encryption

Input : PlainText

1. Let the Plaintext be : "M".
2. Ciphertext be : "C".

C = M^e mod n, Where 1 < M > n.

c) Decryption

Input : CipherText

1. Ciphertext: C
2. Plaintext : M= C^d mod n

2.2 Steganography

It can be defined as the science of concealment and communicating data through apparently reliable carriers in attempt to hide the existence of the data. A secret message are often 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 and 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 .We are having 4 kinds of steganography , Text steganography, Image steganography, Video steganography, Audio steganography. In this paper we are using Image steganography.

Encryption Algorithm

Inputs: Image,Data,Key

1. Intially consider an Image
2. And consider Data and a Key.
3. Now hide the data in the given image by using the secret key.
4. Now send the Stego-Image to the Receiver.

Decryption Algorithm

Inputs : Stego-Image, Key

1. Consider the input be Stego-Image.
2. Using the Secrert Key ,Obtain the hidden message from the Stego-Image.

3. METHODOLOGY

In proposed model RSA , the inputs are two prime numbers P and Q , and a plain text. For Steganogtraphy the inputs are text, key and image.

3.1 Encryption

Inputs : Message, Prime Numbers, Image, Secret Key

1. Consider an Input , It can be :
   a) Text
   b) Image
   d) AudioVideo
2. Convert the input to Base-64 using Base-64 conversion Algorithm.
3. After converting into Base-64 we will be getting a String.
4. Store the entire string in a Text File and save the file.
5. From that file consider each character and apply Rivest Shamir Adelman(RSA).
6. For RSA we must generate two different prime numbers. Using that prime numbers, Calculate Eulers Toient and q(n).
7. Select integer e such that gcd (q(n), e)) =1 & 1< e < (n), Where e is called as Public Exponent.
8. Now sender “A” will Encrypt the message using the Public Exponent.
9. Let the encrypted message be Cipher Text (cm).
10. Consider an image, And hide the encrypted message(cm) in the given image with the secret key Using Stegnography Algorithm.
11. And the secret key will be considerd from RSA algorithm.
12. Now send the Stegno-Image to the Receiver.

3.2 Decryption

INPUT : Stego-Image, key

1. Consider the input be Stego-Image.
2. Using the Secrert Key , Obtain the hidden message from the Stego-Image.
3. The Secret Key can be obtained using the RSA Algorithm.
4. And the obtained message is a Cipher Text. We must decrypt the message.
5. The Decyption of the message can be done using RSA Algorithm.
6. For Decryption the receiver will use the Secret Exponent.
7. And thus the receiver will decrypt the message and it is in the form of Base-64

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

4. EXPERIMENT RESULTS

The above stated hybrid method was applied to the data such as image, video, audio as shown in figure(4). The cover image used for this process is shown in figure(3). Total process of entire method is shown in figure(5).

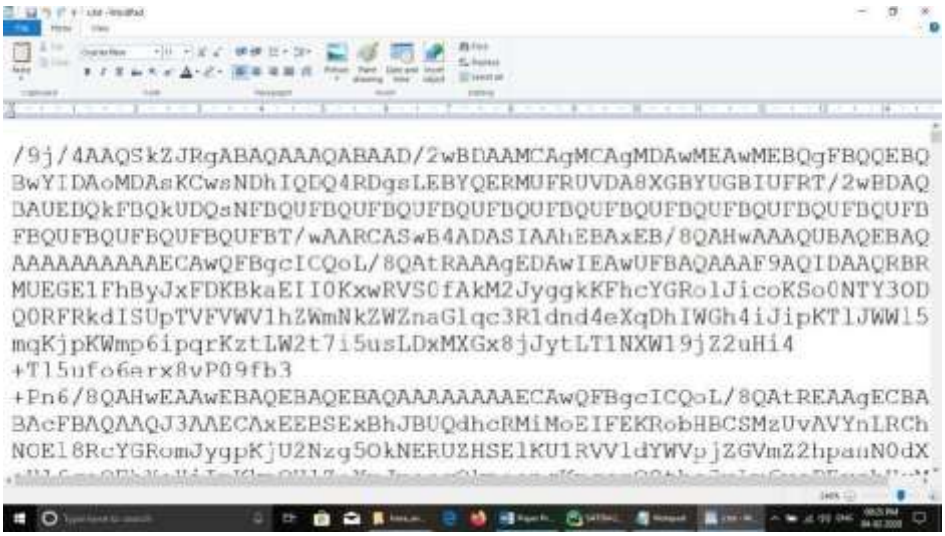


Figure 1 : Image Encryption

Figure 2 : Rsa Encryption

Figure 3 : Cover Image

Figure 4 : Encryption Data

Figure 5 : Stego_Object

5.FUTURE SCOPE AND CONCLUSION

In this project, we deal with the concepts of security of digital data communication across the network. This project is designed by using the steganography and cryptography features factors for better performance. We performed a new steganography method and combined it with RSA Encrption algorithm. 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 colour images. We concluded that in our method the Image files and RSA Cryptography are better . Because of their high capacity. Results achieved indicate that our proposed method is encouraging in terms of security and robustness.

6. REFERENCES

[1] H.Abdulzahra, R. AHMAD, and N. M. NOOR, "Security enhancement; Combining cryptograhy and steganography for data hiding in images," ACACOS, Applied Computational Science,pp.978-960,2014.

[2] P. R. Ekatpure and R. N.Benkar, "A comparative study of steganography & cryptography,"2013.

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

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

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