

Enhancing the security of message in the QR Code using a Combination of Steganography and Cryptography

MSc Internship

Cyber Security

Rohit Jain

Student ID: x18164455

School of Computing

National College of Ireland

Supervisor: Christos Grecos

National College of Ireland
MSc Project Submission Sheet
School of Computing

Student Name:	Rohit Jain
Student ID:	x18164455
Programme:	Cyber Security
Year:	2019
Module:	MSc Internship
Supervisor:	Christos Grecos
Submission Due Date:	12/12/2019
Project Title:	Enhancing security of message in QR Code using a Combination of Steganography and Cryptography
Word Count:	8385
Page Count:	23

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

I agree to an electronic copy of my thesis being made publicly available on NORMA the National College of Ireland's Institutional Repository for consultation.

Signature:	
Date:	11th December 2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Enhancing security of message in QR Code using a Combination of Steganography and Cryptography

Rohit Jain

X18164455

ABSTRACT

Various cryptographic methods are available for completing the objective of data security across the internet, servers, and local systems. But, there is a continuous requirement of more extra protection, which might not fit in cryptography and steganography. Hence, the vital combination of steganography along with cryptography methods can give an extra level of security. Quick Response (QR) codes are adopted widely due to their useful characteristics. QR Code involves robustness, readability, error correction capability, ample information space than old barcodes, etc. Thus, in our idea, we propose a four-layered design for securing the message distribution mechanism by using the QR code image. This design uses the practical and vital use of cryptography and steganography techniques. The offered method gives a higher level of protection based on results. In this paper, we assess our approach against the performance evaluation, securities as discussed in the article.

Keywords:- Elliptic curve cryptography(ECC), Elliptic curve Diffie-Hellman(ECDH), Authenticated Encryption with Associated Data (AEAD) algorithm, QR code, LSB, One Time Pad, PSNR, MSE, Steganography, Cryptography.

1. INTRODUCTION

In this world, communication with new technologies becomes the priority to share information. The information hiding technique can be secure in different ways to make things safer. Few researchers [1] [2] [3] [4] had used steganography, cryptography, and QR codes as a security technique, but their system is not much reliable. Thus the premise of this thesis is to use ECC, ECDH, AEAD with QR Code, and Steganography. Which offers a novel idea that can use to encode and decode the data sent to a receiver. We are applying a novel approach, i.e., the ECC algorithm embedded in the QR Code, which makes the information more secure while transmitting to the receiver. QR-Code is very useful to gather a piece of information while scanning the barcode. This code is a data matrix type. It can manage the alphabetic characters as well as the numerical model of data. The QR-Code can store vertical and horizontal data. Also, it can hold more than 700 alphabetic characters as compared to a 1D barcode. QR code has many benefits, such as the speed of reading, high precision, and substantial small physical size. Barcodes enable automated work processes without human intervention and are widely deploy because they are fast and accurate, eliminate many errors, and often save time and money. QR Code is responsible for describing the hidden information, which is a device-readable digital data representation. The value of regular QR Codes in terms of bar codes has been better in several characteristics like higher capacity, reduced volume, etc. Paired with the provided versatility and modularity, it helps to make more attractive use of QR code. Numerically, QR codes may signify the same data volume for around a tenth of a standard barcode space. The matrix of two dimensions that contain data such as URL, SMS, and contact information. In a single symbol, a QR code version 40 can hold up to 7,089 numeric data and 8-bit byte data of 2953 characters [5]. We can use the QR Code in many fields because of their speed, reliability, and usability. QR code has become hugely popular in specific applications, especially their use in stores and retail chains for pricing products, tracking items, and identifying customers through membership cards. QR code can trail the movement of the product like courier services, flight luggage, sports or music events, theatre hall, car rental [6]. The problem with the barcode, which we solve through our proposed idea is by using QR Code, which is not just to improve the volume of data, but also to minimize the difficulty of decoding information correctly. Difficulties with accurate and reliable color detection, color debugging of the QR code structure could raise further issues. The problem with QR code is the risk that the QR code itself could cause. QR code is usually injected with the SQL injection command by an attacker. Kieseberg [7] stated it could allow tampering into a backend dataset to add a semicolon accompanied by an SQL query, Command injection attacks, phishing attacks, and social engineering. Moreover, it would also be possible to execute arbitrary commands on behalf of intruders using encrypted data as a command-line variable without sanitizing. The second technique we used in the proposed idea is cryptography. Cryptography changes the data into scrambled code, which can fetch back sent to another person. Many cryptography algorithms ensure the integrity and security while sharing the information. Cryptography needs the key with a message to form the encrypted text. The cryptography algorithm is two types, i.e., symmetric and asymmetric. The third technique we used in this research is steganography. Steganography is a mean of concealing information from others.

Steganography's goal is to conceal the vital presence of communication by implanting messages within other objects. Steganography used to send secret information from one place to another place. There are different steganography techniques used to hide confidential data into images [2]. Hidden data can be encrypted text or plaintext. Any data can

be protected in the steganography; hence, all forms of secret data have to be converted into binary. In the image steganography, classified electronic data is hidden behind the image using the proposed algorithm. In which Color QR-Code is considered a vital example of image steganography. In the past few years, QR codes have frequently been changing with their versions. QR-Code is highly being using for information sharing in various fields like a mobile device, virtual stores, QR code payments, website login, loyalty programs, etc. These codes standards are from 1 (21 x 21 modules) to version 40 (177 x 177 modules). Some vital principles need to be considered the knowledge of QR code technology and electronic image techniques. Different version level defines the storage capacity higher version means larger data payloads. The reliability of QR Code standards offers four correction levels, such as L, M, Q, and H.

In the following sections of this thesis, the details related to QR Code, steganography, and cryptography with the encryption, decryption, problems, and solutions are being specified. Section 2 provides a critical review of the research in the field of steganography, cryptography, and QR code. Section 3 and part 6 shows the Evaluation.

1.2 Research Question and Objectives

How can information be securely transmitted using Steganography and Cryptography in a QR-code?

Objectives:

- How can we communicate securely with the help of the QR Code?
- Evaluate the performance and efficiency of the proposed idea. To check, is it possible to communicate via QR Code using steganography and cryptography techniques.
- To what extent we can recover our data if the attacker tries to compromise our QR Code.

The rest of the structure of the paper describes the sections. Section 2 focused on previous work done in the field of the QR Code, steganography, and cryptography techniques. Section 3 explains the methodology in detail implemented in this model. Section 4 describes the features of the design model in the form of algorithms. Section 5 explains the Implementation detail of the proposed idea. Section 6 shows the evaluation of our proposed idea, and the last section describes the conclusion and future work.

2. RELATED WORK

2.1 Different approaches to protect steganography

In [8], by merging the DNA sequence with hyper-elliptic curve cryptography, the researchers introduced the secure steganographic method. Also, using DNA cryptography and steganography, this method has obtained the advantages for both ways to acquire excellent communication securely. The algorithm covers a hidden image in another cover picture by converting the nucleotide into a binary translation table into the DNA sequence. The encoding process consists of three measures on the recipient side. First, the value of both cover picture and the hidden image pixel is converted to its respective DNA three-fold value using DNA 3-fold character transformation. Second, the threefold values are converted into binary values. Third, the last stage, the XOR principle, applies to create the new photo named steganography-image between binary values for both the hidden image and the cover image.

In [4] the author used a secure information hiding system (SIHS) to provide security service confidentially. They hide a message in an image file with SIHS and focus on the LSB technique. In their method, they took the image size of 800 x 600 pixels, which can hold up-to 60kb message. With the help of SIHS, they embedded messages into cover images without steg- key and passcode. The received Steganography image appears to be similar to the cover image because the cover image should be paired with the text. That's because the amplitude of the change is minimal, and thus the modulation of the LSB would not result in a noticeable difference between humans. Hence, permitting high perceptual clarity of LSB. In their result, they found that the size of the message is smaller than the size of the cover image. A big capacity permits the usage of a smaller cover-image for the communication message of the fixed size and therefore reduces the bandwidth to transmit the steganography image.

The author Mohammad shirali shahreza [9] implemented the steganography on mobile phones using LSB pixel colors. They hide each information in two pixels, and that information divided into eight bits with the three colors (RGB). LSB technique improvement might not be identified due to the human eye's incomplete sensitivity. They used the small size PNG image with lossless compression. In the regular steganography algorithms, the hacker can identify the pattern of modified pixels and retrieve the secret information if the size of the data is small compared to the size of the picture. But in the shirali proposal, they used random order pixels in all blocks. As a result, it is difficult to extract information.

In [10] the author implements and analyzes the three steganographic approaches. The first approach is the LSB method after encryption. The header field is appended to the encrypted message before concealing the secret information

(encryption text). The modified text includes steganography key and file extension. Fifty-four bytes BMP header file used. They encrypted the message before embedding the hidden information to the LSB. The second approach is the Pseudo-Random Number Generator (PSNR) sequence used to create any PSNR with random strings 1 and 0. For a random generator, Blum Blum Shub (BBS) was used. BBS creates one as a PRNS, and the hidden information should be implanted into LSB container image. The third approach is scattered LSB method, which divides the BMP file into blocks. In these different blocks, the secret message has been embedded in LSB of container photos based on PRNS.

Author Prem, Rajat, and Ambika [11] proposed a novel approach of text steganography based on null spaces. These white/blank spaces are placed in the text format of the cover document when the binary bit of coded message is equal to 1. And when the binary bit of the coded message is 0, the white/blank space remains unchanged. This encoding scheme has been implemented in the gap between the words. Two programs compose the approach. The first is the concealing system, which would be responsible for concealing information in wording; this means transforming the data. The second program called extractor; this program extracts data from the steganography text.

In [3] the authors presented an image steganography method using the K-means clustering technique. All the text message was encrypted using the DES algorithm. They used 64-bit size 16 round DES. After that, K indicates the clustering of the pixels, which clusters the photo into various sections and integrates information into each section. Several clustering algorithms were used for the separation of images. Separation of an image contains a broad set of pixel details, with each extra pixel having three elements: red, green, and blue (RGB). The encoded content isolates into sections K following the structure of the clusters. In each cluster, these sections are to be concealed. To this end, they used the LSB method.

This work [12] presents a technique that combines strongly encrypted algorithms and steganographic techniques to ensure safe, secure, and extremely hard to decode the interaction of sensitive information. The hidden letter is first encrypted with a cryptography algorithm before it encodes in a QR code. They have used AES-128 to encode and transform the message into the base 64 layouts for more analysis. The encrypted picture scrambles to accomplish a different level of security. A scrambled QR code eventually integrates into an appropriate LSB of the cover picture. They used an LSB approach to achieve steganography for digital pictures. The deciphering procedure collects the top-secret data on the recipient side. Therefore, a four-level safety system is provided for the transfer of a secret message, i.e., first, they encrypting the secret message with the AES algorithm and that encrypted text converted into base64. Second, the base64 encrypted message embedded into the QR Code. The third module creates a random scramble order that is stored together with the obtained RGB values. A scrambled picture is created, which concatenates the three RGB values and afterward generates a picture from it. The fourth module uses a concept known as the least significant bit insertion to obtain digital picture steganography. In the paper [13] Secure QR-Code Based Message sharing System Using Cryptography and Steganography author work on the three-layer to secure message. In the layer 1st RSA algorithm used in cryptography. The second layer ciphertext is hidden into the bar code, and in the third layer, barcode watermarked into an image.

2.1.1 Different approaches have some problems

In their [4] method, the message inserted in the cover image cannot obtain, if some LSB bits are modified in the steganography image. Furthermore, the steganography image alters in any way, e.g., by scaling, rotation, cropping, adding noise, or loss compression, since tampering will also ruin the secret message. They said the human eyes were less sensitive to the color of blue. Therefore more substantial changes could be applied to the color of blue once the changes were recognized. Each byte of information hides, consequently, in two pixels. LSB isn't very strong. It is quite vulnerable to any stego-image filtering or manipulation. Another flaw is that it is the resistance to tampering. An intruder can probably destroy a text by removing or discarding the whole LSB plane.

In this proposed steganography algorithm method [9], the size of the data is small in contrast with the size of the image. The invader can easily discover the design of changed pixels and extract the secret data. On the other hand, if the size of the data is big, The algorithm has reached the end of the image. To solve this issue, it must revert to the original picture and keep secrets in a void pixel. A void pixel defines as a pixel of the main image that does not contain any secret data. For this process, they need a large volume of memory, which the mobile phone does not have. Overall, in the coding and decoding phase, it will take lots of time to find an empty pixel.

Author [10] works on the idea of the Bitmap file. The information can never be retrieved if a steganography image changes by certain image processing technology (rotation, masking, etc.). The first method is easy to crack by the attacker if they know the steganography key. It is very simple for a hacker to gather all the information before received by users.

The main drawback [11] is that it requires a lot of space to encrypt a few bits. For instance, one character is equal to 8 bits and requires an estimated eight inter-spaces to encrypt one character. Also, some challenges occur due to less storage capacity in the cover message. The inclusion of extra gaps to depict data results is an increase in the object size of the

stego-cover image. The secret message can only store in a text file, and change in one bit can result in the difference in ASCII code. In this paper [13] their proposed method talking to much time for encryption and decryption process.

2.1.2. Different approach Solution

Shiraz proposed a solution for the empty pixel. He used the algorithm if the pixel of the block starts with k (Stego-key) and has m (Message) pixels, the last pixel number is $k + m - 1$. An array used to recognize the empty pixel of the recent block. This method uses to cover the info in itself by selecting a block [4].

2.2 QR CODE

2.2.1 Different approaches to protect QR - Code

In this paper, a Security overview of QR Code [14] the author talked about the security overview of the QR Code in which they introduce SEQR – symmetrically encrypted QR code and PKEQR Public key encryption QR code. The SEQR used symmetric key encryption as well as the PKEQR followed the encryption of the public key. For SEQRs, they used asymmetric encryption mechanism in which a secret key is shared both by the reader as well as the EQR writer. Encrypting the message bits, they used the shared secret key AES block cipher. In PKEQR, the authors used the RSA public key, which combined with AES to encrypt it. The next author [15] proposed the novel approach of secret sharing methodology using the QR Code. The suggested technique designed a secure data transfer system, including a QR code built on the secret communication system. The key concept of the process of secrecy splits a secret into N shadows. It is not possible to decrypt the secret message by itself. The secret can only recover if any t ($t=n$) is extracted from n shadows. Their approach based on Shamir's secret scheme. The confidential data split into shares of shadows with the secret sharing method. They created shadows that inserted within each QR-code tag. Anyone who wants direct to read the content from QR codes is impossible if the predefined threshold doesn't achieve the number of accepted shadows.

Author Kuan and Wie-Hsun proposed a user authentication scheme based on the QR Code [16]. They included two parties, i.e., SP – service provider and remote user. The existing customer may request permissions from SP. Every user also has a cell phone with an embedded camera, so they can take a photo of the QR code and decipher it. They divided into two phases, i.e., verification and registration. This method based on the remote authentication model rather than an old smart card. It shows that the suggested QR-code-based verification protocol is effective and practical. Another paper author [17] suggested the new SQR code, which holds data in the encrypted format using the AES algorithm with a 128-bit key. MATLAB version 7.10 has to code the QR Code along with Java 1.8. The proposed method avoids decoding in the case of tampering the QR codes as compared to the old way. The old way had lots of risks for users. During the experiments, they verified that the proposed method gives the actual information after scanning the QR code. Also, if the QR code is damaged or includes any sensitive data, then it would not process the analyze. This is done to avoid the phishing and trojan attack.

2.2.2 Different approaches problems

The keys included in the QR code throughout the encryption and decryption process, and the reliability of the key data must ensure to a larger extent. This suggests that the sender and receiver keys are the same. Another problem is that because of the keys, the processing of such additional information requires the part of the data space. The last problem is to prevent the leakage of information from the native message. The error correction bits must correct errors of the cipher [14]. Paper [17] has a minor problem in their method during the decryption process. It requires more time while creating and reading a QR-Code as they worked on the AES algorithm to secure the QR Code.

2.2.3 Different approaches Solution

The solution for the time process is to use a cipher block chaining approach. To test if the encoded data is unchanged or not, additionally, the hashing can be implemented [17]. The author examined QR code scanners in Android. The scanner is the most commonly downloaded tool and noticed that most scanners could not detect phishing attacks effectively. Furthermore, a much more detailed study of protection, privacy, and accessibility factors needs to build software that represents the user's decision-making process on a URL's confidentiality. Another significant challenge is to establish the design standards for the development of a stable, functional multilayer system for QR code management. The QR code itself and the scanner technology should be designed in such a manner to enable the client to identify a possible threat. They suggested a set of criteria for supporting research in the fields of security and the engagement of humans with the risk scenarios outlined in 3 categories. (1) Requirements for secure QR Code - they recognize safety needs in this portion to protect the QR code strategy. They consider enhancements in the QR code scanning application to be symmetric with the help of a visible QR Code. The visual QR codes assist the user greatly in recognizing the change or replace the QR codes in an attack situation. The more complicated the topic, the more difficult it will be for an intruder to inconspicuously alter QR codes. Digital Signatures - Digital signatures have shown that they are an effective way to improve protection. Furthermore, the emphasis shows integrating digital signatures into the QR code standardization, on verifying the origin of the code and thus to verify if the QR code has been changed. A digital signature makes the attacks on QR code considerably more difficult as the intruder has to change the checksum and authentication

mechanism appropriately. The increasing amount of encrypted information reduces the region in which actual data is encoded. Additionally, QR code readers need to be adapted to check the digital signatures and to show if the verification has been successful, (2) Requirements for service layer, and (3) Usability Requirements. Requirements for service layer - This segment emphasizes the challenges of safeguarding the QR code reader application and is designed to strengthen safe QR codes. The overall aim of the enhancement in the service layer is to increase the security measures of the QR codes and to assess whether it is appropriate for the client to decide whether to escape a malicious code.

2.3 CRYPTOGRAPHY

2.3.1 Different approaches to protect Cryptography

In [18] the authors proposed a new method of the RSA algorithm. The hidden text altered into encryption text, and then the encryption text is hidden in a sound media using the LSB audio steganography methodology. At the recipient end, the ciphertext is obtained from sound media and then decoded by RSA decryption into a text. So, This methodology blends both public-key and steganographic characteristics to ensure a higher level of safety.

In [1] Blowfish algorithm was used by the author to encode a top-secret picture. They said the blowfish algorithm is the quicker, better, and stronger than AES, 3DES, DES, and RC6. They picked and encoded a hidden photo in the BMP layout by Blowfish. The LSB method used to integrate the encoded image into video frames.

In [19], the researchers have suggested a new approach using the 128 key RSA algorithm to encrypt the confidential information before integrating it in a cover picture and to progressively insert the encrypted file into a cover picture by using the F5 steganographic algorithm. They have chosen DCT to integrate the hidden message using the F5 algorithm. The DCT has random parameters. They used the embedding matrix to shorten the length of the text. This method provides a quick, steganographically capable system that prevents observation and analytical threats.

In [20], the authors have proposed a new visual cryptographic technique. This strategy is appropriate for color images of both Grayscale and Bitmap. In this method, the concept of the Residual Number System (RNS) is used for the development of the shares creation algorithm and shares the stacking algorithm of a provided image based on the Chinese Remainder Theorem (CRT). The share creation algorithm separates the hidden image into n number of shares. The shares generated by this algorithm will be in an unreadable form such that it is difficult to reveal the hidden picture. Single share can't show the hidden image. Security is accomplished if certain specific shares are transferred independently through a transmission network. The share stacking algorithm reveals the hidden image by choosing the number of shares as data. Some algorithms may use all shares as data, and some other algorithms may use a subset of shares as data. Decoding done by joining shares, which should take as data. For encoding, additive modulo 255 algorithms are used. Keys are created by using a different method called Mixed Key Generation (MKG). In this design, a block of the size of 8-byte keys is created using the PRN creation algorithm, and the individual bits from each byte is chosen. Since they have an 8-byte word, they showed the parallel operation with 8 bytes of source data. As they created eight keys at a time. Their idea is a rapid, accurate, effective, and easy strategy to implement. Authors Vipul and Madhusudan [21] proposed two approaches for image steganography using cryptography to protect the image from a hacker when the image is being transferred using the S-DES algorithm. The algorithm requires as input, an eight-bit clear text block, and a 10-bit authentication key that generates as the output of an eight-bit ciphertext block. A colored picture, on the other hand, is built up of pixels, each containing three color components, these holding a red, green, and blue part. The color strength of the pixel depends on the dimension of these three parts. These three components that take 8-bit clear text as an input and create an 8-bit ciphertext as an output. A 24 BMP color picture separates into three matrices or rectangular frames, in which each frame or matrix contains pixels correlating to the intensity of the red, green, and blue components. Using the S-DES algorithm, the pixels of the picture were encrypted. The text collected after encryption of the image is also known as the ciphertext. This encrypted text is sent to the recipient along the channel. The recipient party will then use the same unique key to decipher the ciphertext for the image. Rather than transmitting it directly through the stream, the text received can also conceal inside another picture. They used MATLAB for implementation and encrypted photos acquired through the implementation of the S-DES algorithm.

2.3.2 Different approaches problems

There are three major drawbacks to private key cryptography [22]. The first major drawback is that a secure channel is needed for the parties to agree on the key and to transport the key. The second major drawback is that two people are communicating with each other using private key cryptography. They need their unique key, which can quickly add up to an unwieldy number of keys if many people are using the cryptographic scheme. The third major drawback of private key cryptography is that it cannot perform authentication on an open network. The major drawback of public-key cryptography is that it is relatively slow as compared to the private key cryptography. This problem can overcome in the past by combining public-key cryptography with private key cryptography.

3. METHODOLOGY

In persistence, for working on the suggested method for securing the message, a certain method accompanied. Understandings of this method detailed in this section, with a clear knowledge of the mechanisms used along with the details of the idea. The proposed methodology work on cryptography and steganography technique. To increase the security of the secret message transmitting to the receiver through the QR Code. This proposed method works on the four-layer model in which every layer gives extra security. The first layer is the Asymmetric cryptography algorithm, i.e., Elliptic Curve cryptography, which is used to generate shared key and hash that shared key into the secret key. The second layer used the AEAD algorithm for encryption and decryption. The third layer, Steganography technique, is used to hide the ciphertext and nonce into the QR Code. The fourth layer, the QR Code embedded into the cover image using the least significant bits and One Time Pad algorithm to avoid a man-in-the-middle attack.

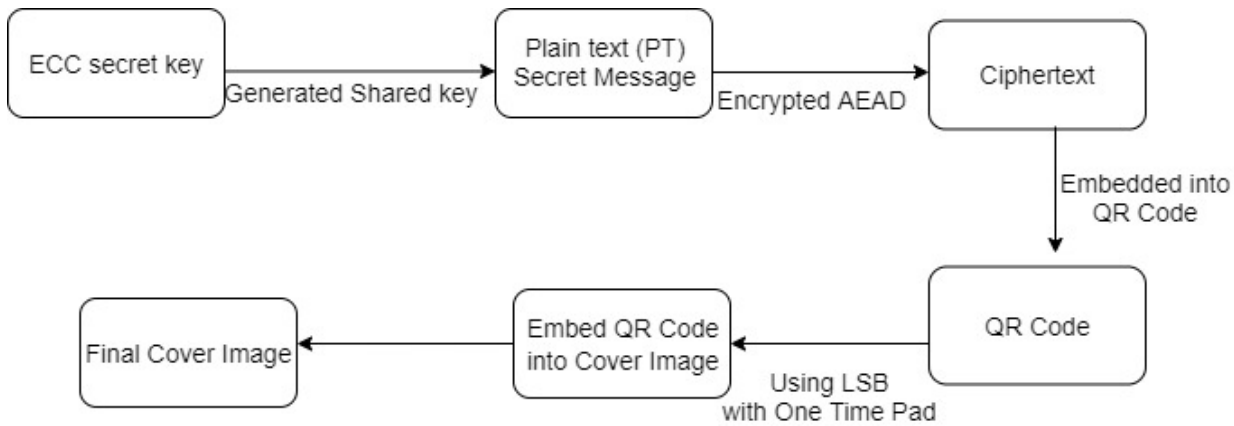


Fig: 1 Encryption Process

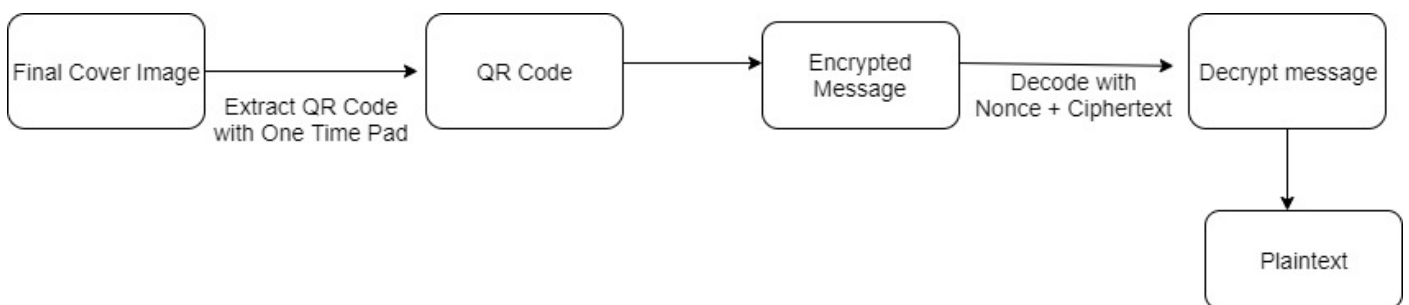
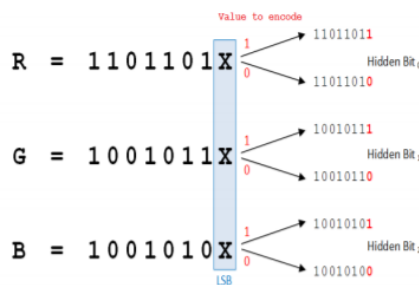


Fig: 2 Decryption Process

LSB: LSB is a simple technique but vulnerable to lossy compression and image handling. Some bits altered directly to hide the data in the image pixel values. Variations in the value of the LSB bit are unnoticeable for human eyes. E.g.:



In the LSB domain method, there is a smaller amount of chance for deprivation of the original image. The additional data can be stored in an image that can be used for secret communication of sensitive data. This method uses to hide the data by a pattern image. This method is valued wherever watermarks become a portion of the image. The data will be

fixed into the most significant part of the image relatively than hiding it into the noisy part. The watermarking methods are additionally integrated into the image, and it can be applied deprived of the fear of demolition of the image. This method is used in 24-bit greyscale images.

Factors Include in Steganography Technique: The usefulness of the steganography method can determine by associating the original cover-image with the stego Image. The following factors of the steganography are:

4.1 Robustness: Robustness denotes the capability of embedding information to remain unbroken if the stego- image undertakes changes, for instance, linear and non-linear filtering, refining or distorting, adding of random noise, spins and scaling, collecting or obliteration, lossy compression.

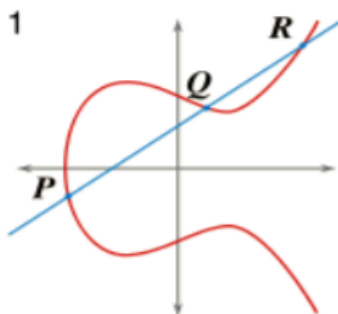
4.2 Imperceptibility: The imperceptibility resources to the indistinctness of a steganography process. Since it is the primary necessity. The strength of steganography deceits in its capability to be unobserved by the human eye.

4.3 Mean Square Error: MSE computed by carrying out byte by byte contrasts of the two images. The illustration of the pixel with 8 bits and the illustration of grey level images up to 256 levels. The alteration in the image can be measured with the Mean Square Error.

4.4 Peak Signal to Noise Ratio: The image steganography scheme must implant the content of hidden data in the image, so as the quality of the image must not alter. PSNR is usually used to calculate the quality of the renovation of lossy compression methods. Larger the value of PSNR better is the quality of an image, i.e., less alteration. PSNR is the relation of the extreme signal to noise in the stego image [23].

ECC Key generation

Our device is for examining the behavior of our secure information transmission; we used the Elliptic Curve Cryptography algorithm. From Fig 1, any plaintext hidden inside the curve. For this design, there is a need secret key to encode the Plaintext to Ciphertext. Furthermore, for decoding the ciphertext to Plaintext. Then insert the same key, which uses for encoding the text.



3.1 Mechanism used for securing the message.

The encryption process uses to secure the message that is transmitting to the receiver. Encoding works in away. Where the encryption is applied to the plain text when the ciphertext is sent over a network, the receiver can decrypt only by ciphertext and nonce. The message we are trying to secure focuses on third party interference. To overcome this interference, we used a random nonce and one-time pad algorithm in our steganography image. One time pad technique used to provide more security.

RSA is a type of public-key encoding that requires a private key to decode a text. Likewise, Elliptic Curve Cryptography (ECC) deploys the public key encoding approach. It evaluates a piece of certain information and gives a comparable level of security similar to RSA. Besides that, the Elliptic Curve Cryptography is using smaller key lengths compared to RSA [24]. It demonstrates that it offers smaller key sizes, and ECC performance is better than RSA, that is why we used the ECC algorithm instead of RSA. In the ECC algorithm, we used ECDH and AEAD (chacha20poly1305).

In the fig 3 plaintext encrypted into ciphertext using Nonce, and secret key. The secret key we generated from the ECDH algorithm, and we hash that shared key to form a secret key. Which gives the extra security, and it is difficult for an attacker to break our ciphertext.

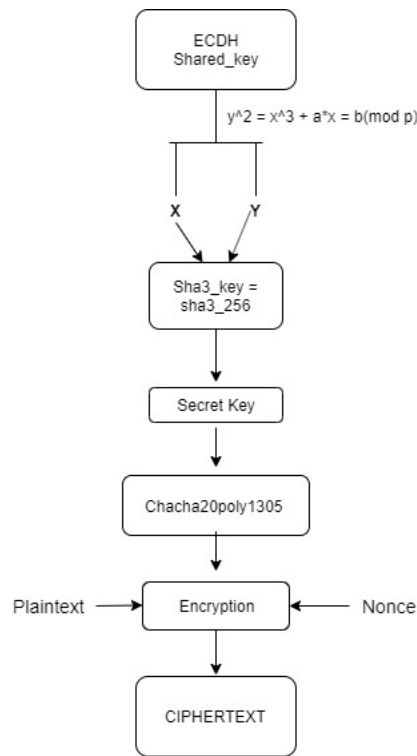


Fig: 3 Message Encryption flow

3.1.1 Algorithm choice

As we have discussed above, Elliptic curves give the quick and robust keys feature which we can use for the encoding and decoding process. This algorithm intensifies end to end safety through various structures encryption process, key exchange in a secure way, and decryption process. We use Elliptic Curve Diffie Hellman (ECDH), Authenticated Encryption with Associated Data (AEAD algorithm) – Chacha20-poly1305. Where Chacha20 is a high-speed cipher that is faster than the AES. Whereas poly1305 is a high-speed message authenticator. In this, AEAD supports two operators, i.e., “seal” and “open.”

Working of “SEAL” operation :

- Plaintext – encrypt the message
- A secret key
- A special IV(initialization value) – it should be unique between calling of the seal process with the same key, else the privacy of the encryption is fully compromised.
- Optionally any other, non-secret, supplementary data. That data won't be encoded, but still it will be authenticated. This is the associated data (AD) in AEAD.
- The “SEAL” process uses the key along with IV to encode the plaintext into the encrypted text of the same length using the cipher. Here cipher is Chacha20 in Chacha20poly1305.
- When the data is encoded, the “SEAL” uses the key to create a 2nd key. The 2nd key can use to produce the hash of the Associated data, the encrypted text, and the separate lengths for each key. Here hash is poly1305 in Chacha20poly1305.
- The last step takes the hash value and the encoded text to create the final message authentication code (MAC) and apply it to the encrypted text.

Working of “OPEN” operation:

This process is the opposite of the SEAL operation. It takes the IV, key, and create the message authentication code of the encrypted text and the associated data. Then reads the message authentication code added to the cipher message, and compares the two. If any difference occurs in the MAC value, it means the AD or the encrypted text tempers and rejects it as unsafe. If the two matches, then the process decrypts the encoded text and gives the original plaintext.

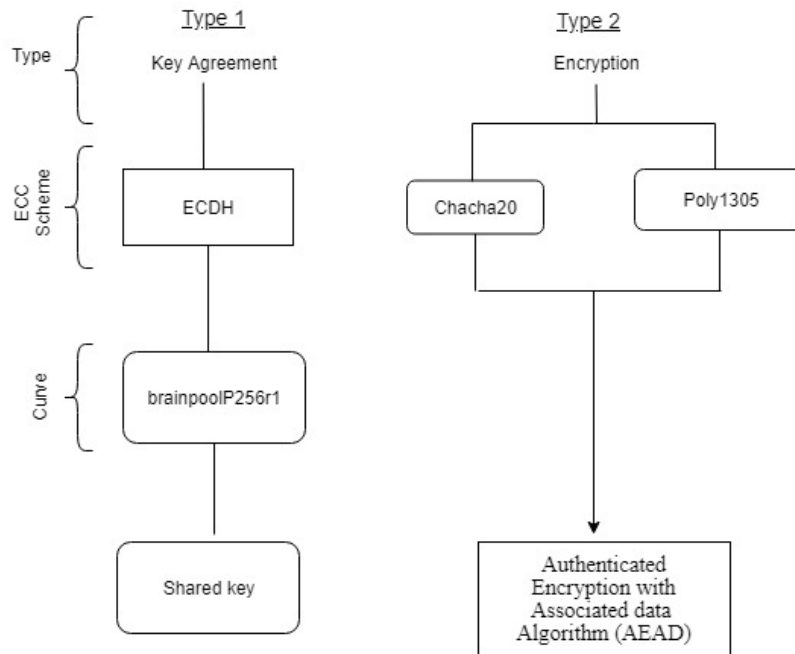


Fig: 4 Mathematical Structure

3.1.2 Transmitting Secure message.

As we all know, in today’s world, nothing is secure. It is very hard to transmit the information to another person in a secure way. Day by day, attackers are trying to hack the information channel through different techniques. In our design, we transmit the message which encodes with ECDH and AEAD. For the development of the algorithm, Python is the universal purpose programming language that we are using for the back end development. It is user-friendly and works a limited number of codes, which gives it desirable by maximum developers. Python gives users with extensive support, and the script is also relatively simple. Therefore it is an excellent option for development. We encode the sent message, and the user will decode at the user end.

3.2 Steganography Attacks

Testing is an important part of development. It shows the effectiveness of the proposed model. Some attacks are made to test the model performance. To see if the attacker received our information at what extend the attacker can damage our data. The following attacks have chosen from a list of threats.

3.2.1 Scaling attack

The input picture A can obtain grayscale, RGB, or binary. If A has more than two dimensions, imresize only resizes the first two dimensions. If the scale is in the range [0, 1] then, B is less than A. If a scale is bigger than 1, B is more significant than A. The stego image is tested in MATLAB using the “imresize” function.

3.2.2 Median filter attack

A widespread manipulation in an image is median filtering. The median filter is a non-linear spatial filter that usually used to reduce noise spikes from an image. When implemented to an image pattern, it works by managing the median of the neighborhood pixels, applying a window that glides pixel by pixel over an image. The stego image is tested in MATLAB using the “medfilt2” function.

3.2.3 Rotation Attack

In this attack, a small change in an image (0.1 degrees) are enough to disorder the whole bits. It can be done clockwise or anti-clockwise. The rotation attack range is between +1 to -1 degrees.

3.2.4 Motion blur attack

Motion blur is the effect that changes the image pixels. Some changes in the image can make an image blur to some extent. For the motion blur attack “imfilter” function used in the MATLAB.

3.2.5 Noise Attack

There are two types of attacks (salt and pepper & Gaussian noise) that can add to the stego image. In MATLAB imnoise function used to perform the noise attack. To understand the attack, 10% means that 10% of the Image pixel (1 in the 5 pixels) are changed. 50% means half a pixel of the image is changed.

3.2.6 Contrast Adjustment

Contrast adjustment remaps image depth values to the complete chain of the data type. An image with excellent contrast has visible contrasts between black and white.

3.2.7 Crop

Image cropping is a lossy method usually used in practical life. Unnecessary cutting will cause the image useless. Hence the degree of a cropping attack, in general, won't be enough. In MATLAB “imcrop” function used for cropping the image in a different size.

4. DESIGN MODEL

This section highlights the proposed method used to design this structure with a complete explanation of its architecture. In section 3, we have discussed that the ECDH, chacha20poly1305 that are used in the Elliptic curve cryptography encryption and decryption process.

4.1 Symbols List

PT	Plain text
ET	Encrypted text or Ciphertext
FO	Final output
EQR	Encrypted QR Code
DQR	Decrypted QR Code
OTP	One time pad
CI	Cover Image
PUK	Public Key
PK	Private Key
QRI	QR Code Image
LSB	Least significant bits

4.2 Elliptic Curve Diffie Hellman

Elliptic curve equation

$$Y^2 = x^3 + a*x + b \pmod{p}$$

1: Sender private key and Receiver private key. Sender and Receiver public key generated by multiplying curve g (generator point) with both private keys. We get:-

- Sender private key * curve.g = Sender public key
- Receiver private key * curve.g = Receiver public key

2: Shared key calculation

- Sender private key * Receiver public key = Sender shared key $\Rightarrow y^2 = x^3 + a*x + b \pmod{p}$
- Receiver private key * Sender public key = Receiver shared key $\Rightarrow y^2 = x^3 + a*x + b \pmod{p}$

3: Sender `_shared_key` = Receiver `_shared_key`

In this proposed method, we have used the “brainpoolP256r1” curve g (generator point).

4.3 Algorithm's

4.3.1 The following algorithm `Secure_Message_Method` presents the methodology of the proposed idea. As the list of symbols displays the information components in the proposed idea.

Encryption_Secure_Message_Method()

1: Begins

2: `PT` \leftarrow User Input Message

3: `ET` \leftarrow `Call_Encryption_ECC_(PT)`

4: `QRI` \leftarrow `Call_QR Code generate`

5: `QR_Code_Image` \leftarrow `Call_Embed_ET_ECC`

6: `QRI_ET` \leftarrow `Call_Embed_CI`

7: `CI` \leftarrow `Call_Embed_QRI(Size)`

8: Shows FO to the User `CI`

9: End()

- The procedure begins with providing the Plaint text (Top-secret message) in the proposed method.
- Next, the ECC encryption technique with chacha20poly1305 uses to encrypt the plain text into ciphertext. Here the 256-bit key is used in the ECC algorithm.
- The encrypted text provided to the next stage. The encrypted message embedded into the QR Code.
- The QR Code with the encrypted message embeds into the cover image with LSB
- The final step is that the QR code size is embedded into the cover image using the OTP technique.

Decryption_Secure_Message_Method()

1: Begins

2: `CI` \leftarrow `Call_Decryption`

3: `CI` \leftarrow `Call_Extract_QRI(Size)`

4: `DQR` \leftarrow `call_ET`

5: Show PT

6: End()

The following algorithm, `Encoding_Image (QR Code)`, shows the steganography image. The algorithm shows the QR Code encoding process into the cover image using Least significant bits and a One-time pad.

4.3.2 Encoding_Image (QR Code)

- 1: Begin
- 2: Get the dimensions of the QR image
- 3: Create a random binary code with the same QR image size
- 4: Perform the one time pad by (XOR) of the binary code and QR image
- 5: Embed the encrypted image from step 4 in the selected cover image plane on two steps
 - 5.1: Firstly, embed the plane with zeros in the cover image.
 - 5.2: find the one's locations and then embed the ones into the plane
- 6: Save the resulting image (Stego image) to a file to be sent to the decoder.

Decoding_Image (EQR)

- 1: Start with reading the stego image sent from the encoder.
- 2: Go to the plane where the hidden QR image is obtained.
- 3: Reshape the result to the size that was agreed both in encoder and decoder (195x195)
- 4: Create the same binary code as the encoder
- 5: Use the one time pad again by (XOR) operation of the resulted code and the retrieved QR encrypted image obtaining the decoded image.

4.3.3 Message_encryption_process

- 1: Start()
- 2: Shared_key ← call_generated_key
- 3: Sha3_key ← call_sha3_256(hash)
- 4: Sha3_key ← call_x_y_component (Shared_key)
- 5: Secret_key ← call_sha3_key
- 6: object ← call_secret_key * Chacha20poly1305
- 7: nonce ← random_nonce
- 8: Ciphertext ← call_PT + nonce + object
- 9: Ciphertext_End()

- The process begins with the shared key, with we explain in the above process of creating a shared key.
- We get the x and y component from the deffie-hellman shared key process. That x and y component we updated with sha3_256() to create a secret key. With the secret key, we will encrypt our plaintext.
- We used Chacha20poly1305 multiply with a secret key.
- For ciphertext, we need plaintext, nonce, and secret key.

Message_decryption_process

1: Start()

2: Plaintext ← Ciphertext + secret_key + nonce

3: Plaintext_end()

- For decrypting the encrypted message. The receiver needs the ciphertext and secret key. The receiver can read the encrypted text after providing the secret key and ciphertext.

4.4 Steganography Attacks

4.4.1 Scaling attack

```
stegoImage = imresize(stegoImage,1);
```

4.4.2 Median filter attack

```
stegoImage = medfilt2(stegoImage(:, :, 1), [1 1]);
```

4.4.3 Rotation Attack

```
stegoImage = imrotate(stegoImage,.01);
```

4.4.4 Motion blur attack

```
stegoImage = imfilter(stegoImage, fspecial('motion', 1.25, 0));
```

4.4.5 Noise Attack

```
stegoImage = imnoise(stegoImage, 'Salt & Pepper', 0.02);
```

4.4.6 Contrast Adjustment

```
stegoImage = imadjust(stegoImage(:, :, 1));
```

4.4.7 Crop

```
stegoImage = imcrop(stegoImage);
```

4.5 Crptography attacks

4.5.1 Known plaintext attack

Attacker knows the plaintext. Allows the attacker to find secret key and nonce.

- For secret_key: Urandom (32) is required. The probability of finding secret_key is $1 / (16^2)^{32} = 1/256^{32}$ so $1 / 1.1579208e77$

- For nonce: urandom (12) is required. The probability of finding a nonce is $1 / (16^2)^{12} = 1/256^{12}$ so $1 / 7.9228162e28$

#This is the number of ways to be sure that we can find the correct secret key

$n1 = 256^{32}$

#This is the number of ways to find the correct nonce

$n2 = 256^{12}$

The complexity of this algorithm is $O(n1 * n2)$ as $n1$ and $n2$ is too large, this algorithm will never end.

4.5.2 Known Ciphertext attack

Attacker knows the ciphertext and nonce. Allows the attacker to find secret key and plaintext. The probability of finding secret key is $1/256^{32} = 1/1.1579208e77$

$$n1 = 256^{32}$$

The complexity of this algorithm is $O(n1)$ as $n1$ is too large, this algorithm will never end because the probability to find secret_key is $1/n1$.

4.6 System Flow Diagram

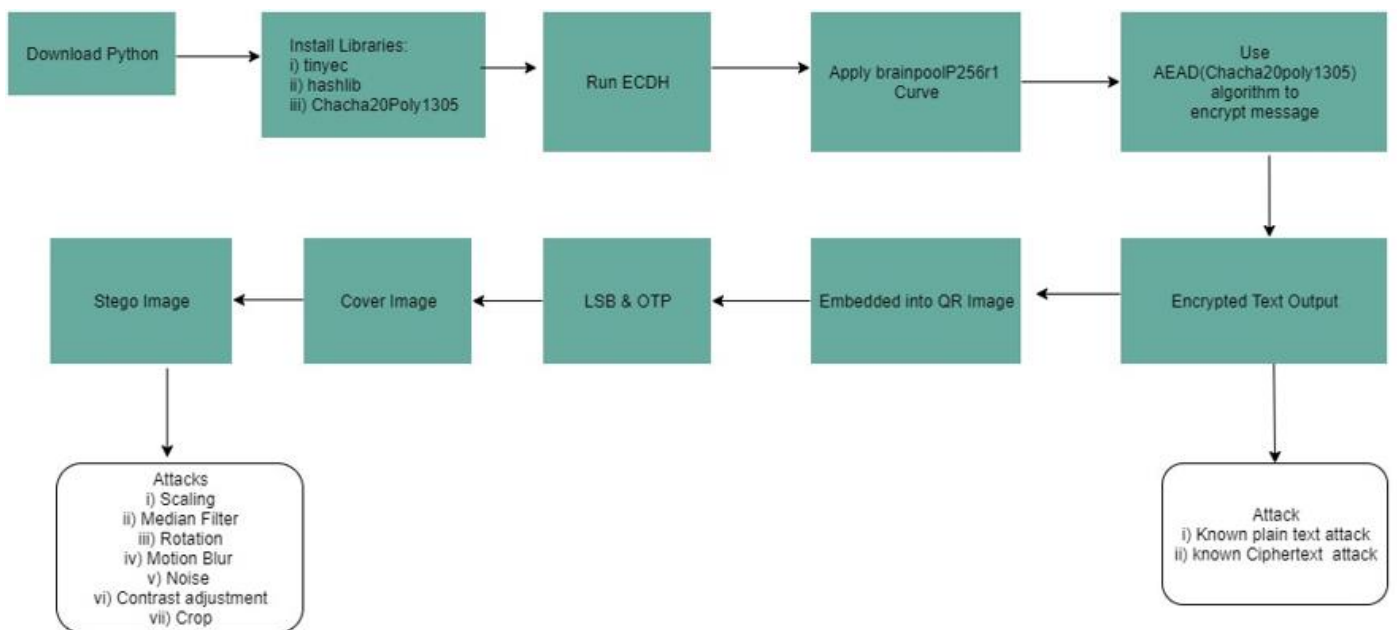


Fig: 5 Flow diagram

5. IMPLEMENTATION

5.1 System Work Flow

The figure shows the process of encryption of plaintext into ciphertext. Here ECDH used for Shared_key obtained from the Alpha private key and Beta public key and message stored on the brainpoolP256r1 curve. The message to B encoded with the Chacha20poly1305. Additionally, the encrypted message M is acquired by $M = e(\text{Secret_key} + \text{Nonce} + m)$. This ciphertext is sent to the receiver where the encrypted message is decrypted using the given formula below.

$$\text{Plaintext (PT)} = d(m + \text{Secret_Key} + \text{Nonce})$$

$$PT = d \{e(\text{Secret_key} + \text{Nonce}) + m\}$$

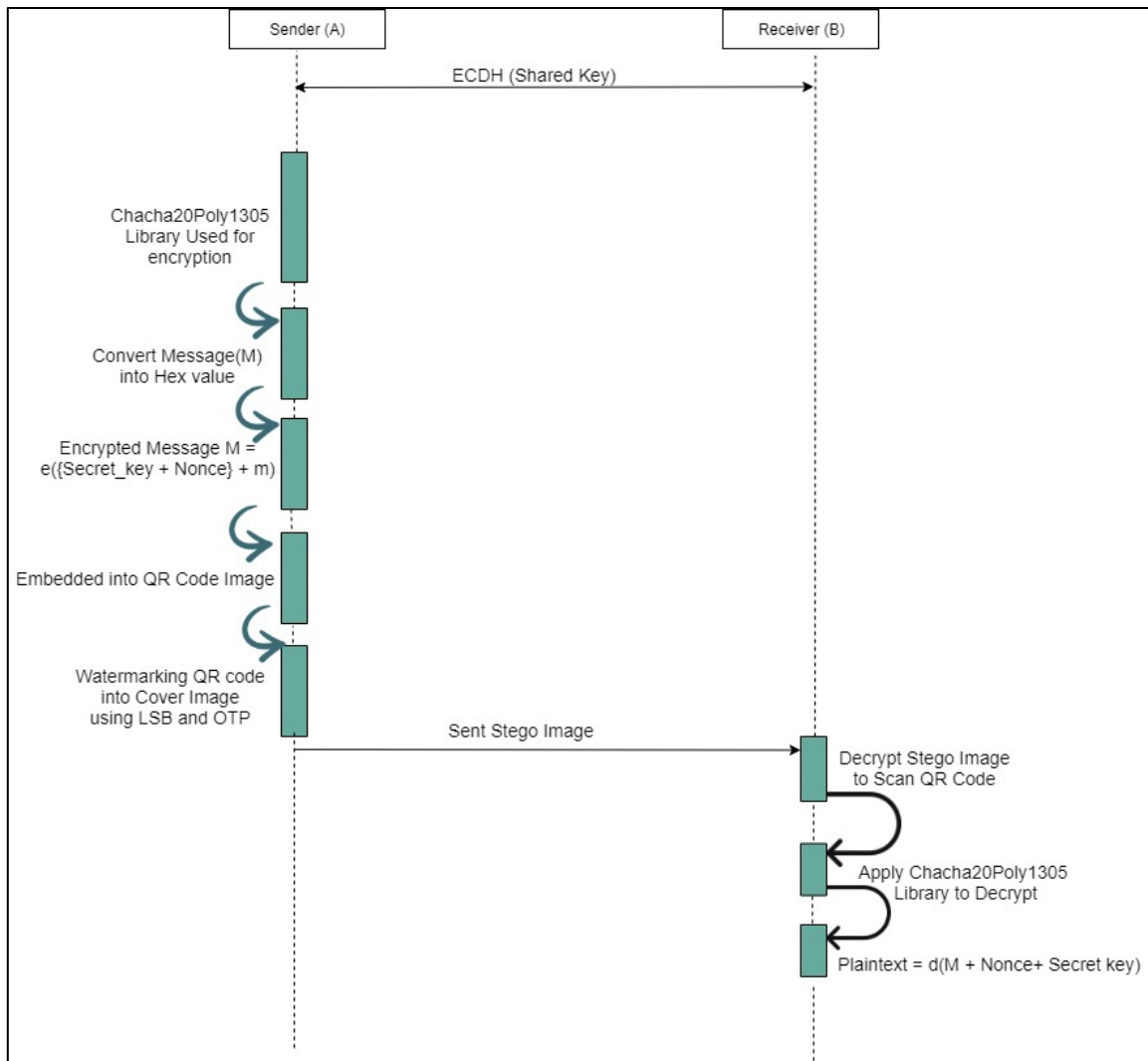


Fig:6 Work Flow Diagram

5.2 Implementation of Work Flow

5.2.1 ENCRYPTION OF MESSAGE AND IMAGE

5.2.1.1 Generating Secret key

As discussed in section 4.2, the process of generating the shared key is obtained from the sender's private key as well as the receiver's private key, as shown in figure 7.

```
V:\steg image\Rohit\final code>python sender.py
Sender's private key: 3050153037081679705873446868240637296222919987970697501180320008272762634273

Receiver's private key: 61434279641050395131503952861655812658217324401819568372243784515977539648558

Sender's public key: (35793709582294552822593442399953466920547479706755927290409931023740522404010, 6005869939592010207234173490752419997020023129183333676050125559868265160504) on "brainpoolP256r1" => y^2 =
x^3 + 56698187605326110043627228396178346077120614539475214109386828188763884139993x + 17577232497321838841075697789794520262950426058923084567046852300633325438902 (mod 76884956397045344220809746629001649093037950200943055203735601445031516197751)

Receiver's public key: (54702440472700154275614974153704769008342760259400974155860027817675025532369, 17728856685602330533060745100429176984204907309699179168000008837950920455289) on "brainpoolP256r1" => y^2 =
x^3 + 56698187605326110043627228396178346077120614539475214109386828188763884139993x + 17577232497321838841075697789794520262950426058923084567046852300633325438902 (mod 76884956397045344220809746629001649093037950200943055203735601445031516197751)

Sender's shared secret key: (25262540663143304835930806194018078864227212491933201526314844725472656111939, 23988298877672413039544694620054100106486815645788910106005237088327567621862) on "brainpoolP256r1" =>
y^2 = x^3 + 56698187605326110043627228396178346077120614539475214109386828188763884139993x + 17577232497321838841075697789794520262950426058923084567046852300633325438902 (mod 76884956397045344220809746629001649093037950200943055203735601445031516197751)

Receiver's shared secret key: (25262540663143304835930806194018078864227212491933201526314844725472656111939, 23988298877672413039544694620054100106486815645788910106005237088327567621862) on "brainpoolP256r1" =>
y^2 = x^3 + 56698187605326110043627228396178346077120614539475214109386828188763884139993x + 17577232497321838841075697789794520262950426058923084567046852300633325438902 (mod 76884956397045344220809746629001649093037950200943055203735601445031516197751)

Shared secret keys match each other

(25262540663143304835930806194018078864227212491933201526314844725472656111939, 23988298877672413039544694620054100106486815645788910106005237088327567621862) on "brainpoolP256r1" => y^2 = x^3 + 56698187605326110043627228396178346077120614539475214109386828188763884139993x + 17577232497321838841075697789794520262950426058923084567046852300633325438902 (mod 76884956397045344220809746629001649093037950200943055203735601445031516197751) <class 'tinyec.ec.Point'>
```

Fig: 7 Shared key

5.2.1.2 Encryption process of Plaintext

The fig below shows that the Sender encrypted the plaintext with Nonce and secret key. The receiver got the encrypted text.

```
nonce: 6bfd128d8a53816ac7b1cc6e
Enter text to encrypt: My name is Rohit. This is a test.
Encrypted Text: 94f4b4506337281e87cf29ff0da3a921cbefe6a1960be577687e8b870f8ba7a9203ed178d0df279f801670a13175cd1d30
V:\steg image\Rohit\final code>
```

Fig: 8 Random nonce and encrypted text

5.2.1.3 Encryption Process of Image

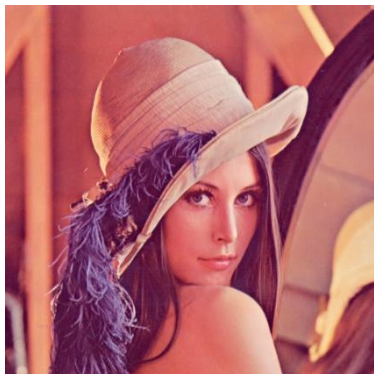


Fig 9 Original Cover Image



Fig 10 Original QR Code

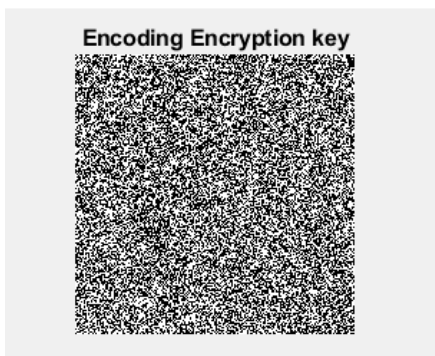


Fig 11: OTP key

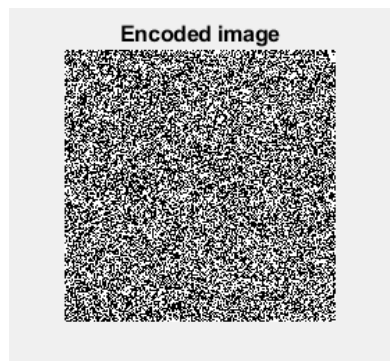


Fig 12 OTP encoded image

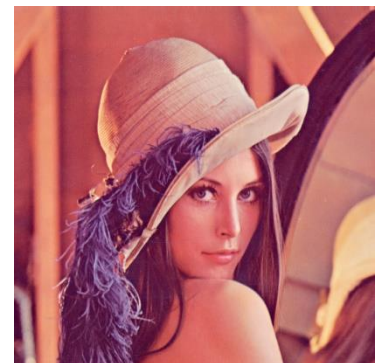


Fig 13 Stego Image

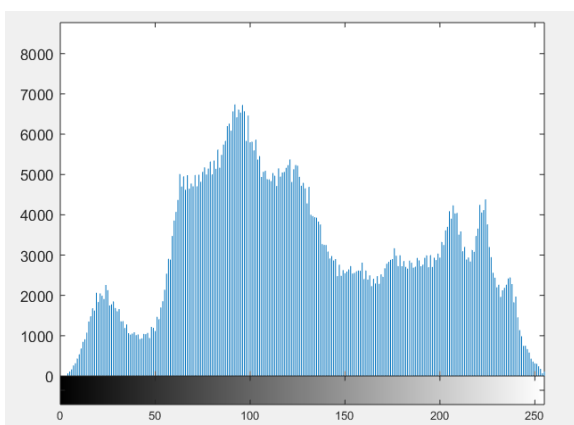


Fig 14: Histogram of Original Cover Image

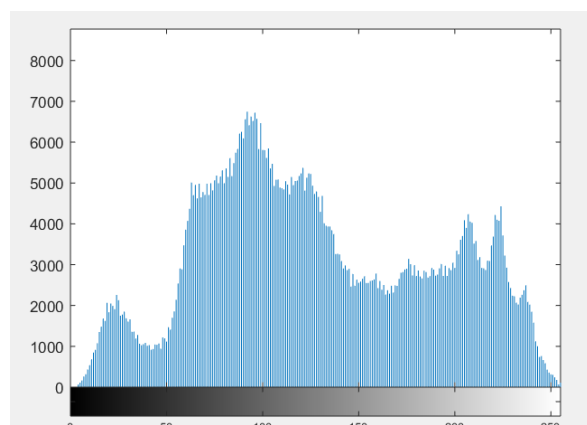


Fig 15: Histogram of Stego Image



Fig 16: Comparing Original cover Image and Stego Image

PSNR = 64.2705

MSE = 0.024324

Fig 9 Shows the Original cover image in which we tried to hide the QR code using MATLAB. Fig 10 shows the QR code, and the QR code includes the ciphertext (encrypted message). In the QR code, we used the one-time pad algorithm to make QR Code more secure. In fig 13, we successfully embedded encoded QR code with the ciphertext in the Original cover Image. Fig 14, fig 15 Histogram of both the Images. It shows there is no difference after hiding the QR code into the Cover Image.

5.2.2 DECRYPTION PROCESS OF MESSAGE AND IMAGE

5.2.2.1 Decryption process of Stego Image



Fig 17: Stego Image

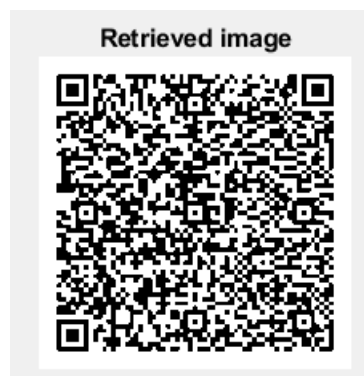


Fig 18: Retrieved Image

As you can see in fig 18 that we can successfully retrieve our QR code.

5.2.2.2 DECRYPTION PROCESS OF RECEIVER

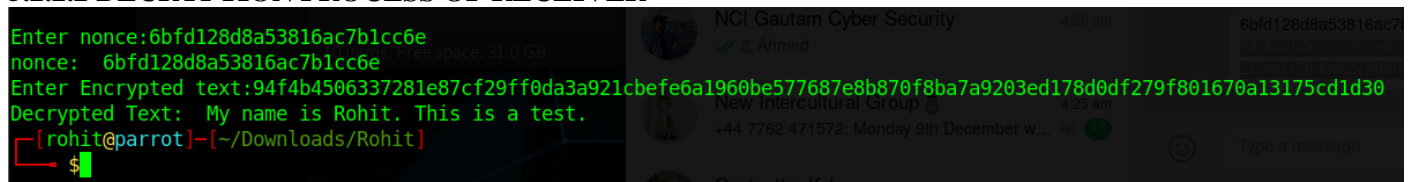


Fig: 19 Decrypting plaintext

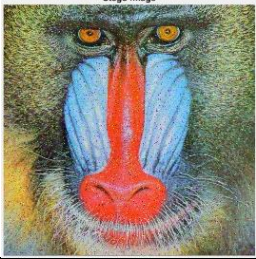


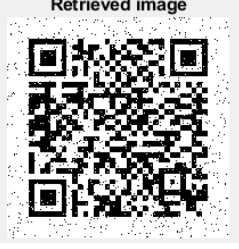

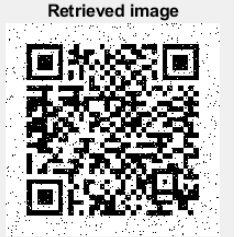
In the fig: 19, receivers will decode the encrypted message with Nonce and ciphertext. As shown in the fig receiver able to decrypt the message.

6. EVALUATION

In the experiments, we worked on the 1 bit plane of the Image. As you can see in the tables of the experiments.

6.1 Experiment 1 Noise Attack

Table 1

PNG	STEGO IMAGE	RECOVERED IMAGE	PSNR	MSE
Baboon.png			18.2978	962.2681
Lenna.png			18.1815	988.4015
Peppers.png			17.9394	1045.0595

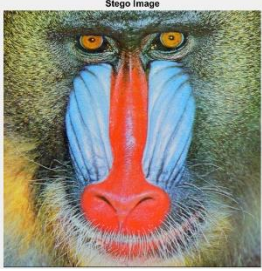
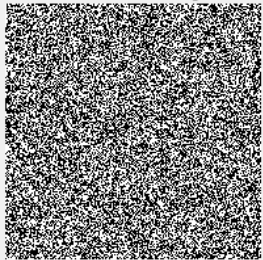

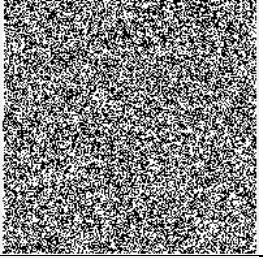

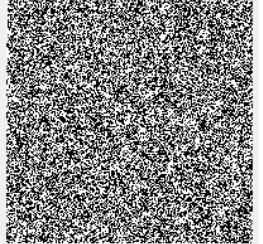
6.2 Experiment 2 Motion blur attack

Table 2

PNG	STEGO IMAGE	RECOVERED IMAGE	PSNR	MSE
Baboon.png			54.2454	0.24465
Lenna.png			60.1764	0.062436
Peppers.png			60.3382	0.060154

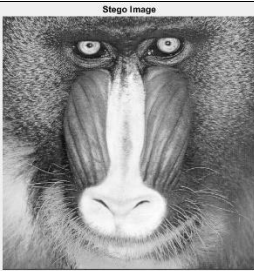



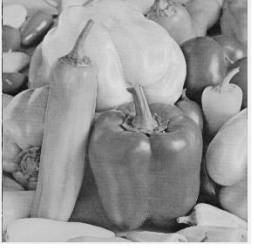

6.3 Experiment 3 Scaling attack

Table 3

PNG	STEGO IMAGE	RECOVERED IMAGE	PSNR	MSE
Baboon.png			26.6543	140.4911
Lenna.png			34.711	21.9775
Peppers.png			32.9813	32.7301

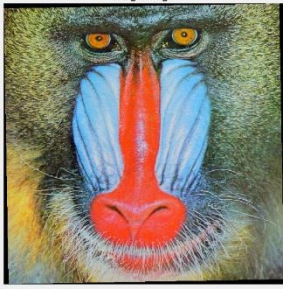
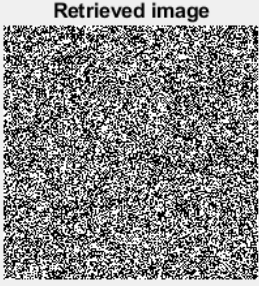

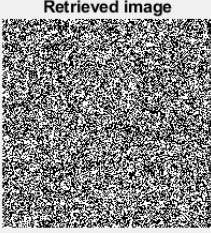
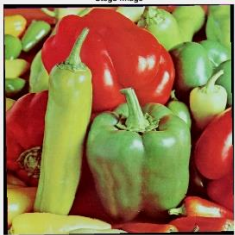
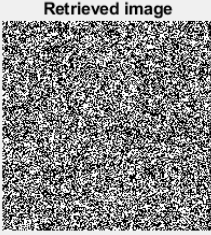
6.4 Experiment 4 Median filter attack

Table 4

PNG	STEGO IMAGE	RECOVERED IMAGE	PSNR	MSE
Baboon.png			12.8272	3391.2586
Lenna.png			11.3987	4712.0656
Peppers.png			11.3987	4712.0717

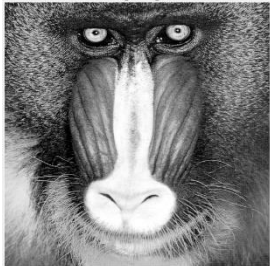
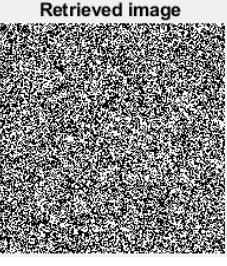

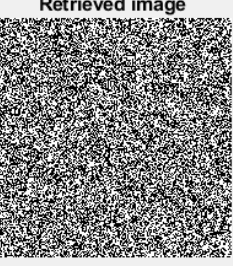
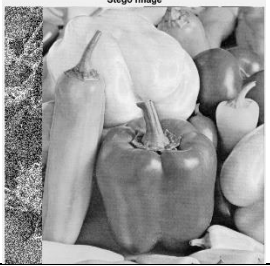

6.5 Experiment 5 Rotation Attack

Table 5

PNG	STEGO IMAGE	RECOVERED IMAGE	PSNR	MSE
Baboon.png	 A color image of a baboon's face with a prominent red and blue nose. The image is labeled "Stego Image" at the top.	 A square image filled with random black and white noise, labeled "Retrieved image" at the top.	15.1392	1991.4014
Lenna.png	 A color image of a woman wearing a hat, labeled "Stego Image" at the top.	 A square image filled with random black and white noise, labeled "Retrieved image" at the top.	17.2565	1223.0152
Peppers.png	 A color image of various peppers, labeled "Stego Image" at the top.	 A square image filled with random black and white noise, labeled "Retrieved image" at the top.	16.4551	1470.8498

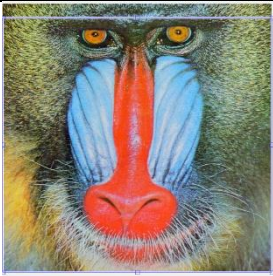
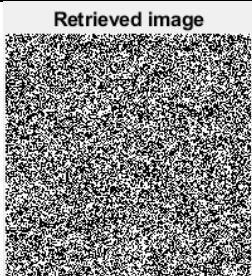

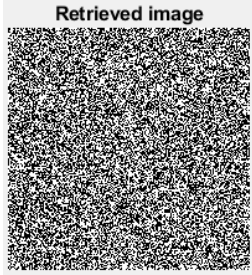

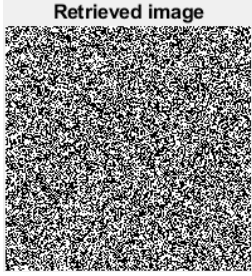
6.6 Experiment 6 Contrast Adjustment

Table 6

PNG	STEGO IMAGE	RECOVERED IMAGE	PSNR	MSE
Baboon.png	 A grayscale image of a baboon's face, labeled "Stego Image" at the top.	 A square image filled with random black and white noise, labeled "Retrieved image" at the top.	12.0963	4012.8718
Lenna.png	 A grayscale image of a woman wearing a hat, labeled "Stego Image" at the top.	 A square image filled with random black and white noise, labeled "Retrieved image" at the top.	12.277	3849.2772
Peppers.png	 A grayscale image of various peppers, labeled "Stego Image" at the top.	 A square image containing a QR code, labeled "Retrieved image" at the top.	9.4013	7463.6168

6.7 Experiment 7 Crop

Table 7

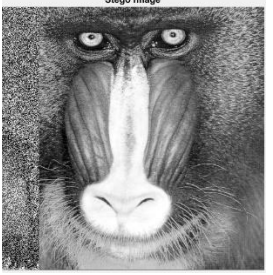
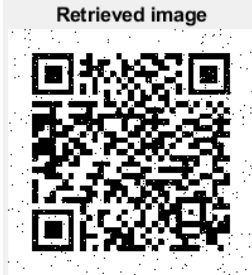


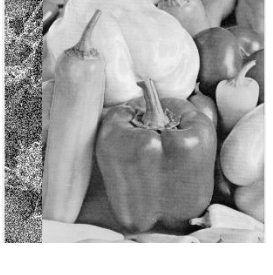

PNG	STEGO IMAGE	RECOVERED IMAGE	PSNR	MSE
Baboon.png		Retrieved image 	15.284	1926.1
Lenna.png		Retrieved image 	16.52	1449.0263
Peppers.png		Retrieved image 	12.3331	3799.8519

Experiment 1 Contrast Adjustment

We tried on a different plane. So, we get our QR code back on the 8-bit plane.

Most sig bit- changing in the plane.

Table 8

PNG	STEGO IMAGE	RECOVERED IMAGE	PSNR	MSE
Baboon.png	<small>Stego Image</small> 	Retrieved image 	11.4317	4676.3387
Lenna.png	<small>Stego Image</small> 	Retrieved image 	11.1782	4957.4371
Peppers.png	<small>Stego Image</small> 	Retrieved image 	9.4799	7329.7263

6.8 Limitations

- Message encryption and Image encryption done through the device carried out at the sender end. Message encryption is done with Python, Image encryption done using MATLAB, and encrypted hidden text with the help of the StegHide tool.
- ECC, ECDH, AEAD gives protection in phases of key exchange and the encryption method. The two procedures can be executed using One Time Pad & LSB in steganography. Also, using specific libraries in Python to implement the Hashing and message authentication in cryptography. The compound of accurate results is a much more useful system, but the primary constraint that we came over was the decryption process at the receiver end in some steganography attacks.
- Three attacks were vulnerable for our steganography Image.
- The Elliptic Curve equation is used to drive the secret key from the SHA3 hash function given as:-

$$y^2 = x^3 + ax + b \pmod{p}$$

Using the hash function in our secret key gives the extra security feature in our ECC algorithm.

6.9 Discussion

The design for securely transmitting the information using cryptography and steganography is implemented. Still, many constraints confronted throughout the implementation of the plan. A few modifications should be made from the suggested idea. Even the concept for transmitting the information completes from the design. Some of the findings are identified.

- Some of the attacks are conducted on our design, which proved evident outcomes. Where the noise attack, contrast adjustment attack, median filter attack, motion blur attacks are performed. In these attacks, we are getting our QR code back. Whereas in the crop, rotation, and scaling, we are losing our QR code.
- If an attacker tries to disrupt the Stego Image with some attacks, then it is difficult to retrieve the information from the QR code. We performed some attacks on the stego image. But still, our system is secure because the attacker can only disrupt stego Image. An attacker cannot read our QR Code Image. One-time pad algorithm is used to stop the Man-in-the-middle attack. Without knowing the size of the QR Code, no one can decode our stego Image.
- If an attacker tries to find plaintext and ciphertext it is not possible to find the right plaintext and ciphertext. We used sha3 to create shared key between sender and receiver and random nonce for the communication. Which makes the algorithm more secure.
- In our proposed method, encryption time and decryption time is faster than [17] [13]. The mentioned author worked on three layers of security, but we worked on the four-layer security method.

Comparing	Application field	Robust	Encryption Time	Decryption Time	Security layers level
[13]	Secret sharing	High	8.0152 sec	3.6952 sec	Three layers
Proposed	Secret Message sharing	High	1.3011 sec	0.77157 sec	Four layers

- In the previous paper, the author [22] talked about the two issues. The first issue occurs during communication between two parties. Their key should be unique. Second, the issue is that the private key doesn't perform authentication, and the public key is slow as compared to a private key. In our proposed method, resolve these two issues. We multiplied the sender and receiver private key with a Curve and created a public key. After getting the unique public key, we again multiplied the public with their private key to get the secret key. With this process, we created a strong key for communication.
- In the paper author [4] talks about the issue when we change in the pixel; we can't get our image back. But in our proposed method, we got our images back on the 1-bit plane, as you can see in the table 1,2,4. Also, we got our image back on the 8-bit plane, as shown in table 8.

7 Conclusion and Future Work

The goal of this design is to examine the Elliptic Curve cryptography scheme and steganography for securing the message in the QR code. Our design achieved the idea of securing the text from being caught while being transmitted to the receiver. Implementation of the ECDH curve kept the security of the secret key. AEAD algorithm process used for encryption and decryption. The seven attacks performed in our research and two attacks on cryptography. Four steganography attacks validated that we can get our image back. While in the cryptography attacks it is not possible for an attacker to break the encrypted text. We are decrypting ciphertext manually with the help of python IDE. Also, we implemented the LSB technique using the One Time Pad algorithm.

7.1 Future work

- Built scanning application to read the QR Code and decode the QR code through the scanner.
- Use a different technique for watermarking so that you can get your image back in all the attacks.
- Use color QR Code to store large information, which is easy to transmit more information in the small size Image.

References

- [1] M. Arya, "Secure Image Hiding Algorithm using Cryptography and Steganography," in *IOSR Journal of Computer Engineering (IOSR-JCE)*, India, 2013.
- [2] K. Joshi and R. Yadav, "A new LSB-S image steganography method blend with Cryptography for secret communication," in *2015 Third International Conference on Image Information Processing (ICIIP)*, Wanknaghat, India, 2015.
- [3] B. Pillai, M. Mounika, P. J. Rao and P. Sriram, "Image steganography method using K-means clustering and encryption techniques," in *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Jaipur, India, 2016.
- [4] M. M. Amin, M. Salleh, S. Ibrahim, M. R. Katmin and M. Z. Shamsuddin, "Information hiding using steganography," in *4th National Conference of Telecommunication Technology, 2003. NCTT 2003 Proceedings.*, Shah Alam, Malaysia, Malaysia, 2003.
- [5] K. H. Pandya and H. J. Galiyawala, "A Survey on QR Codes: in context of Research and Application," vol. 4, no. 3, March 2014.
- [6] A. S and J. R. L, "Secure Color QR Codes," pp. 77-85.
- [7] P. Kieseberg, M. Leithner, M. Mulazzani, L. Munroe, S. Schrittwieser, M. Sinha and E. Weippl, "QR Code Security".
- [8] p. Vijayakumar and V. Vijayalakshmi, "An Improved Level of Security for DNA Steganography Using Hyperelliptic Curve Cryptography," *Wireless Personal Communications*, vol. 89, no. 4.
- [9] M. S. Shahreza, "An Improved Method for Steganography on Mobile Phone," vol. 4, 2005.
- [10] W. W. Zin and T. N. Soe, "Implementation and analysis of three steganographic approaches," in *2011 3rd International Conference on Computer Research and Development*, Shanghai, China, 2011.
- [11] P. Singh, R. Chaudhary and A. Agarwal, "A Novel Approach of Text Steganography based on null spaces," *IOSR Journal of Computer Engineering (IOSRJCE)*, vol. 3, no. 4, pp. 11-17, 2012.
- [12] B. Karthikeyan, A. C. Kosaraju and S. G. S, "Enhanced security in steganography using encryption and Quick Response code," in *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, Chennai, India, 2016.
- [13] A. Mendhe, D. K. Gupta and K. P. Sharma, "Secure QR-Code Based Message Sharing System Using Cryptography and Steganography," in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, India, 2018.
- [14] K. Peng, . H. Sanabria, D. Wu and C. Zhu, "Security Overview of QR Codes".
- [15] C. J. Chou, Y. C. Hu and K. H. Ju, "A Novel Secret Sharing Technique Using QR Code," *International Journal of Image Processing*, vol. 4, 2010.
- [16] K.-C. Liao, W.-H. Lee, M.-H. Sung and T.-C. Lin, "A One-Time Password Scheme with QR-Code Based on Mobile Phone," in *2009 Fifth International Joint Conference on INC, IMS and IDC*, Seoul, South Korea, 2009.
- [17] N. Goel, A. Sharma and S. Goswami, "A way to secure a QR code: SQR," in *2017 International Conference on Computing, Communication and Automation (ICCCA)*, Greater Noida, India, 2017.
- [18] A. Gambhir and A. R. Mishra, "A New Data Hiding Technique with Multilayer Security System," *International Journal of Innovations & Advancement in Computer Science IJIACS*, vol. 4, May 2015.

- [19] M. Mishra, G. Tiwari and A. K. Yadav, "Secret communication using Public Key steganography," in *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, Jaipur, India, 2014.
- [20] R. K. H S, P. K. H R, S. K B and G. Aithal, "ENHANCED SECURITY SYSTEM USING SYMMETRIC ENCRYPTION AND VISUAL CRYPTOGRAPHY," in *International Journal of Advances in Engineering & Technology*, 2013.
- [21] V. Sharma and Madhusudan, "Two new approaches for image steganography using cryptography," in *2015 Third International Conference on Image Information Processing (ICIIP)*, Wanknaghat, India, 2015.
- [22] E. J. Radlo, "Legal issues in cryptography," in *Financial Cryptography*, Berlin, Heidelberg, 1997.
- [23] H. Kaur and J. Rani, "A Survey on different techniques of steganography," in *MATEC Web of conferences 57, 02003 (2016) ICAET*, India, 2016.
- [24] M. Bafandehkar, R. Mahmood and Z. M. Hanapi, "Comparison of ECC and RSA Algorithm in Resource Constrained Devices," in *2013 International Conference on IT Convergence and Security (ICITCS)*, Macao, China, 2013.