

IRIS_teg: A NOVEL APPROACH TO IMPROVE THE
SECURITY OF MOBILE BANKING
TRANSACTIONS BY USING IRIS DATA IN
STEGANOGRAPHY

by

Sameen Yasar Ashraf
17101473

Syed Mahmud Hafiz
17101195

Soumik Saha Dip
17101329

Lamia Rahman
14101225

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 2022

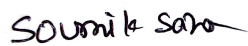
© 2022. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



Soumik Saha Dip
17101329



Syed Mahmud Hafiz
17101195



Sameen Yasar Ashraf
17101423



Lamia Rahman
14101225

Approval

The thesis/project titled “IRISteg: A novel approach to improve the security of mobile banking transactions by using IRIS data in Steganography” submitted by

1. Sameen Yasar Ashraf (17101423)
2. Syed Mahmud Hafiz (17101195)
3. Soumik Saha Dip (17101329)
4. Lamia Rahman (14101225)

Of Spring, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 29, 2022.

Examining Committee:

Supervisor:
(Member)

A.M. Esfar E Alam
Lecturer
Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam Coordinator
Associate Professor
Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi
Associate Professor
Department of Computer Science and Engineering
Brac University

Ethics Statement

- 1) This material is the authors' own original work, which has not been previously published elsewhere.
- 2) The paper is not currently being considered for publication elsewhere.
- 3) The paper reflects the authors' own research and analysis in a truthful and complete manner.

Abstract

In the world of digitization, digital banking has become a coveted commodity. Digital banking is a computerized system based on modern technology that offers clients one of the easiest banking systems with minimal costs and speedy services. However, with the rise of digital banking amongst the masses, fraudulent activities centring on these digital platforms has increased as well. On the other hand, there are almost next to no techniques to trace these digital banking fraudsters. Thus, we have thought about this issue and will be conducting research on the methods of Cyber Forensics to investigate fraudulent activities in digital banking. The main focus of this article is how to improve Cyber Forensic Investigation in Mobile Banking Systems in Bangladesh. After our research, we could come to a conclusion that Steganography is the best Cyber Forensic method that can be implemented. On top of that, to add another layer of security, we have added a layer of IRIS scan. The IRIS data will be taken by the banks to make this process a more secure way of online transactions and to develop a simple and effective digital banking system which resists any kinds of data theft or fraud.

Keywords: IRIS Scan, AES, LSB, Steganography, M-Banking, SMS

Dedication

We would like to dedicate this thesis work to our loving parents. As well as, all the amazing faculties we came across and learnt from in the course of pursuing our Bachelors degree. It's been a worthwhile experience.

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis have been completed without any major interruption.

Secondly, to our Supervisor Mr. A.M. Esfar - E - Alam sir for his kind support and advice in our work. He helped us whenever we needed help.

Thirdly, to all the researchers before us who prospered machine learning field, all the reviews they gave helped us a lot in our later works. And finally to our parents without their throughout support it may not be possible. With their kind support and prayer we are now on the verge of our graduation.

Table of Contents

Declaration	i
Approval	ii
Ethics Statement	iii
Abstract	iv
Dedication	v
Acknowledgment	vi
Table of Contents	vii
List of Figures	1
1 Introduction	2
1.1 Background Study	2
1.1.1 LSB Algorithm	3
1.1.2 AES Algorithm	4
1.1.3 IRIS Scanner Biometric Data	4
1.1.4 Image Steganography	5
1.1.5 Encryption and Decryption	6
1.1.6 Bank User Privacy	7
1.2 Problem Statement	9
1.3 Research Objective	9
1.4 Research Gap	10
1.5 Challenges Faced	10
1.6 Thesis Outline	10
2 Literature Review	11
3 Methodology	15
3.1 Working Process	15
3.2 Used Architecture	16
3.2.1 AES	16
3.2.2 LSB	19
3.2.3 Steganography	20

4	Research Methodology	22
4.1	Data Acquisition	22
4.2	Workflow	24
5	Result Analysis	30
5.1	AES Analysis	30
5.2	IRIS Security Analysis	31
5.3	AES versus Other Encryption Models	31
5.3.1	By the factor of Decryption	31
5.3.2	By the factor of Time	32
6	Future Work and Conclusion	33
6.1	Future Work	33
6.2	Conclusion	33
	Bibliography	35

List of Figures

1.1	Process of Image steganography	6
1.2	Unsecured Communications	8
3.1	Full working process	15
3.2	A possible substitution box	16
3.3	Shifting Rows	17
3.4	Equation for multiplication of columns	17
3.5	Variation of Round Keys	18
3.6	AES Algorithm	19
3.7	LSB Technique	20
3.8	Image Steganography	21
4.1	Acquiring the IrisCode	23
4.2	Equation of Gaussian Filter	24
4.3	Polar form afternormalization	24
4.4	Binary conversion	25
4.5	AES encryption function	25
4.6	AES decryption function	26
4.7	inputs	26
4.8	Outputs of the ciphered text	27
4.9	Encoding function for Image steg	27
4.10	decoding function	28
4.11	Graphical image of encoding	28
4.12	Graphical image of decoding	29
5.1	Chart for Cracking AES	30
5.2	Chart for comparison of AES against other Encryption methods	31
5.3	Chart for comparison of AES rounds against Serpent and Twofish	32

Chapter 1

Introduction

Since the earliest days, banks have been a place where people would go, store their money/valuables, receive interest and send/receive money to/from other accounts. It was a symbol of trust. Everyone, when aged as an adult, opened a bank account.

But with the passage of time, with modernisation and industrialization banks started to follow several methods in order to make the processes easier. And with the help of modern digital technology, banks started to go online i.e. people could access their accounts from digital devices like computers or smartphones. This has eased life for both the customers and authorities. Digital banking is a lot cheaper than physical banking. It also offers comfortability and flexibility for the customers. The goal is to suggest a better and more secure process by using the Cyber Forensic Method; Image Steganography. We have also added a new layer of security; IRIS scan which will personalize security from person to person.

1.1 Background Study

The goal of this project is to become familiar with the many varieties of steganography that are currently in use. In order to recover the message image, image steganography and data decryption are both used. A picture steganography method is used to explain how this can be done, as it can be done in many ways.

Image steganography is performed by utilizing data hiding. We have found two different kinds of methods which is used in image steganography one is Spatial Methods and another is Transform this into spatial methods

Very useful technique to accomplish this thing is to use LSB algorithm to hide important message into another different sort of medium, which is short for "least significant thing." The LSB substitution method is used in steganography. As an example, each image is composed of three distinct parts of a pixel. Each bit of information for this pixel is stored in a single byte. Modifying the first bits of data allows for the insertion of secret text into the data that pixels store. To proceed, it must have a difference between the text that has been stored and the text that will be obscured by the image that is lower or equal in size. After that, we'll use the AES technique to further secure the information. The data is encrypted with the help of this algorithm. This technique utilizes symmetrical block ciphers with keys

of 128, 192, or 256 bits to encrypt plain text in blocks of 128 bits. In other words, we can call this unique established strategy the Rijndael algorithm. As a result of its exceptional track record in data security, the Advanced Encryption Standard (AES) has been adopted as the global standard and also it was the of Advanced Encryption Standard Process which was held in 1997. The DES algorithm, which had a lot of flaws and had to be changed, provided the inspiration for this new approach. In order to get a high-contrast picture of someone's eye, the process of iris recognition, also known as iris scanning, is carried out and maintained by lighting the iris with both visible and near-infrared light. Biometric data obtained from an iris scanner is also utilized. Fingerprinting and facial recognition are both examples of biometric technologies that include voiceprinting as an element. A database of suspect images already exists, according to the designers of the iris scanning technology. This database can be used to validate the identification of a suspect. It's done by comparing the suspect's iris scans to the database's images.

1.1.1 LSB Algorithm

The least significant approach (LSB) is a popular and simple method for storing information in a cover file. LSB substitution is a strategy used in steganography. For example, every picture is made up of three different parts. The information about this pixel is saved in a single byte. Each pixel's information may be modified to incorporate hidden text by altering the first bits of the data. There must be a lesser or equal size difference between the text being stored and that which will be hidden by the image before you can proceed. It is well known that the individual picture elements, or pixels, of an image are saved as individual bits. Although Each grayscale pixel's intensity is stored in eight bits (1 byte). For a color image which consists of three layers, each pixel needs 24 bits (8 bits for each layer). There are some steps which we used in LSB image steganography : The steps for hiding the message picture are as follows: [3]

- 1) The cover art for the book should be examined. As a consequence of integrating the message graphic, it is easier to hide the changes that have been done.
- 2) Examine the message picture attentively.
- 3) Create separate layers for each image's bit planes.

Up to four of the cover image's least significant bitplanes can be replaced with the image's upper four bitplanes without changing the picture. Using fewer bitplanes from the message picture might distort and lose information from the returning image. (1) Replace the cover image's lowest 4 bitplanes with the message image's highest four. (2) Recombining bitplanes yields the Steganographic image.

Each pixel which was originally in a grayscale image is afterwards depicted by an array of eight bits. The final bit of a pixel is referred to as the "Least Significant Bit" since its value has no bearing on the pixel's overall value. Consequently, this property is used to conceal picture data. If the last two bits were classified Least Significant Bits, they would only alter the pixel value by "3." This provides more space for storage. One steganographic method involves substituting a data bit for the image's least significant bit (LSB). This approach is susceptible to steganalysis, thus we encrypt the raw data prior to embedding it in the image. However, even

while the encryption process increases the necessary time, it also provides a higher level of security. This is an extremely simple way. The least significant bits of a portion or all of the bytes in an image are replaced with bits containing the secret message. The LSB embedding approach serves as a basis for a number of strategies for concealing messages in multimedia. It is feasible to apply LSB embedding in particular data domains, such as hiding a message within the color values of RGB bitmap data or the frequency coefficients of a JPEG image. LSB embedding can be utilized with a variety of data types and formats. Consequently, LSB embedding is one of the most significant steganographic techniques now in use.

1.1.2 AES Algorithm

The Advanced Encryption Standard (AES) is a digital encrypted communication specification published in 2001 by the National Institute of Standards and Technology (NIST). However, Despite its more complicated implementation, AES has gained widespread popularity because it provides more security than both DES and triple DES. The encryption process which is divided in to four different parts are:

- 1)SubBytes : After this step, the replacement will go into effect. Each byte will be replaced with a different byte at this point. S-boxes are used to perform this task. A byte will never be replaced by itself or by the complement of the byte that is currently being used because of the manner this substitution is carried out.
- 2) Shiftrows : Each of the four rows of the matrix is shifted to the left. Any entries that ‘fall off’ are re-inserted on the right side of the row.
- 3)MixColumns : This step essentially involves multiplying a matrix by itself. A certain matrix is multiplied across each row, affecting each byte’s position on each row’s data.
- 4)AddRoundKeys : A total of 128 bits have been created by combining the 128 bits of the round key with the 128 bits of the matrix. The ciphertext will be generated if this is the final iteration. if this doesn’t happen, the 128 bits are read as 16 bytes and we begin a new sequence of operations like this one.

The AES standard is widely used in modern encryption and is supported by both hardware and software. There are currently no known cryptanalytic methods that can be utilized against the AES. Keys that use the Advanced Encryption Standard (AES) are also ”future-proofed” against hypothetical advances in the capacity to do exhaustive key searches, thanks to their built-in flexibility.

Like DES, the security of AES can only be ensured through proper implementation and the use of an appropriate key management strategy.[4]

1.1.3 IRIS Scanner Biometric Data

The method of iris recognition, which is often referred to as iris scanning, involves capturing a high-contrast snapshot of a person’s iris while lighting it with both visible and near-infrared light. This is done in order to recognize the iris. It is a form of biometric technology that, together with fingerprinting and facial recognition, is classified in the same category as this particular method. It is claimed by developers of the iris scanning technology that it enables law enforcement agents to establish

or authenticate the identity of a subject by comparing the iris scans of a suspect to an already existing database of photos of suspects. They also claim that iris scans are both faster and more trustworthy than fingerprint scans. This is due to the fact that it is simpler for an individual to hide or change their fingers than it is to improve the appearance of their eyes.

Iris recognition works by taking a photograph of a person's iris while lighting it with both spectrum of light which is visible and also not visible with eyes. This technology that falls into the same category as fingerprinting and recognizing people's faces. Developers of the iris scanning technology claim that it enables law enforcement agents to establish or authenticate the identity of a subject by comparing the iris scans of a suspect to an already existing database of photos of suspects. They also claim that iris scans are faster and more trustworthy than fingerprint scans. This is due to the fact that it is simpler for an individual to hide or change their fingers than it is to improve the appearance of their eyes.

A technique called Iris Scanning measures the eye's irises, which are the colored circles in the eye. Biometric iris recognition scanners use infrared light to illuminate the iris and pick up unique patterns that the human eye can't see. There are a number of factors that can interfere with an iris scan, including eyelashes, eyelids and specular reflection. Pixels that only include the iris are the end result. In order to decipher the iris's information, the eye's lines and colors are studied to derive a bit pattern. This bit pattern is digitally encoded and compared to templates in a database (one-to-one consistent and reproducible) to make sure it is correct or to identify it. In our paper we use iris scanner biometric data of an user. Approximately 240 biometric features are captured by iris scanners, each of which is unique to a person's eye. The data is then digitally represented by the scanners. The computer has a record of the numeric representation of the iris-derived data in a database. For users, the procedure of signing up and verifying their identity is really simple: the system scans their eye to authenticate their identity. However, a four-step enrollment process transforms iris data into a biometric ID before it can be used as an identification method.[9]

1. Image Capture: This is basically the scanner takes high quality images from the users left and right eye using near infrared light
2. Compression: It is then compressed using JPEG 2000 methods to preserve the original's excellent quality. This helps in the removal of artifacts, such as visual distortions.
3. Template creation: Biometric templates created from the image data can be utilized in future scans to verify the identity of the user.

1.1.4 Image Steganography

Image Steganography is a technique for hiding valuable information in an image file. Firstly, the image send to a user which is called cover image after that to decrypt the cover image steganography process started and this process after steganography is named stego image. $A*B$ or $A*B*3$ matrixes are used to keep an image in memory, with each pixel's intensity value being represented by a row and column in each matrix. By changing the values of selected pixels in an image. image steganography can be used to hide a message. An encryption algorithm is used to narrow the field

of candidates. To decipher the message, the individual receiving the image must be familiar with the same process in order to select the appropriate pixels. This has been reported to be the case. As per it was read here [1]

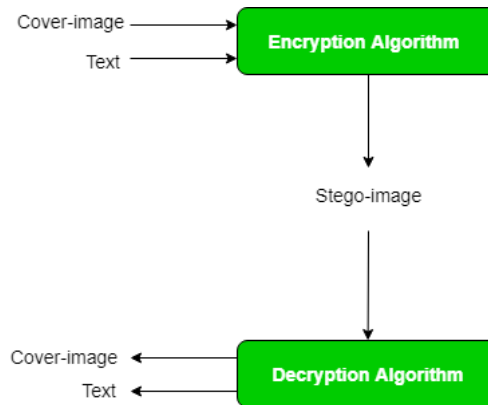


Figure 1.1: Process of Image steganography

The following is a brief description of how the steganographic process works: Hidden message and the stego key make up the stego object. In the graphic above, the Stego function and stego-key are shown as examples of ways in which the message can be encrypted by the person sending it. Hidden messages can be retrieved from receivers using this method. In this article, we will explain how picture steganography is used and how a user can decrypt a secret text using image steganography.

1.1.5 Encryption and Decryption

The use of encryption while transmitting confidential or personal data over the Internet is a must for anyone concerned about their privacy. Even if plain text is intercepted before it reaches its intended users, it is impossible for hackers, cyber-criminals, and other online snoops to decipher the secret code used by encryption. The recipients of the communication will be able to decipher the scrambled data once they receive it since they will have their own unique key. Encryption protects data you send, receive, and preserve on a computer or other device. Smartphone, fitness watch, and online banking data are examples of this type. Encryption scrambles text so only the person with the secret code can read it. It secures critical data.

Encryption scrambles plaintext, such as an email or text message, into ciphertext. Digital data stored on computers or transmitted across networks like the Internet is private. If someone sends an email, the recipient receives it in its original form when they open it. This process is known as decryption. Encryption keys are required by both the sender and recipient in order to decrypt a sent communication. In our Paper we will use encryption method in a highly effective algorithm. Two of the different types of encryption are (1) Symmetric, which is one single key for both ends and another is (2) Asymmetric, which uses two different keys for both ends.[14]

Decryption is the process of recovering encrypted data and re-encrypting it. As a general rule, it is a decryption procedure in reverse. A secret key or password is required to decrypt the encrypted data, so it can only be decrypted by an authorized user. Data security is an important consideration when putting in place an encryption

and decryption scheme. It's important to keep an eye out for illegal organizations or individuals while sending or receiving information over the Internet. Data is encrypted to prevent theft or corruption. Text files, photographs, e-mails, user data, and directories are often encrypted. Decryption recipients see a password prompt or window to access encrypted data. In order to decrypt the data, the system extracts and converts it into text and graphics that can be understood by both a reader and a computer. Automated and human methods exist for decryption. It can also be done with a set of passwords or keys.[13]

1.1.6 Bank User Privacy

It's not just IT companies that need to be concerned about cybersecurity. For every business, even non-profits, it is essential. However, financial security is of critical importance. Millions of transactions are processed every day by banks and other financial institutions. Digital payment transfer platforms handle the majority of these transactions. As a result, banks have become a popular target for cybercriminals.

Financial institutions, like most people, have type A or type B personalities when it comes to banking security. In the banking industry, data security is a strategic issue for those who are Type A. The board of directors is the first line of defense in Type A security. They are adamant in adhering to all administrative and data security requirements. It's especially important now, when fraud and deception are on the rise and data theft from within organizations is a real possibility. Type B financial institutions, on the other hand, approach data security in the banking business in a more relaxed manner. They do it because they think it's what the regulators expect them to do. They don't take initiative. If there is no evidence of a data breach, the situation should be fine. Because they are more trusting of insiders, this is their worst flaw. They are more concerned with those who come from outside their group.

In order to keep customers safe, computer vision systems and digital identity verification algorithms must be trained on a diverse variety of profiles, lighting, and environments. Technological innovation should be able to distinguish between a fake and the actual thing in an image or video, but it should not obstruct people from using important financial goods and services. By virtue of its operations, the banking industry is nevertheless vulnerable to cyber-attacks. Bank websites will remain a target for cybercriminals, and financial institutions must do everything they can to prevent this. You can take certain precautionary measures when banking online. You may protect your personal information and your money by using an Internet security software that incorporates Internet banking security and being alert of probable scams.

In our work, we intend to improve the bank security system by employing some algorithm and utilizing an additional layer of biometric data from IRIS scanners, as we previously discussed.[13]

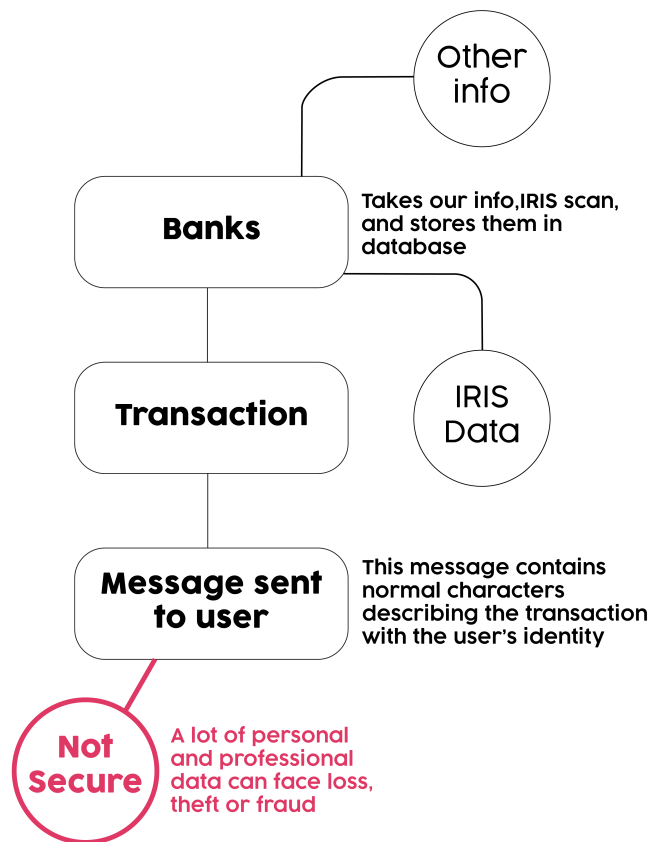


Figure 1.2: Unsecured Communications

1.2 Problem Statement

Due to the rise of users in digital banking due to its mobility and cloud services, there has been a huge rise in data theft as well. Several types of fraudulent activities take place.

1. Phishing: The hacker tries to redirect the user to a website that is hosted by the hacker. Phishing is normally done through clicking on malicious links found in emails, websites and even phone SMS. The fraud can take all kinds of personal information like PINs and passwords from the user and use it to their advantage.
2. Social Engineering hacks: By baiting users into enticing offers and rewards the hacker can converge traffic to their trap and in turn, get access to their personal information.
3. Identity Theft: The fraud takes the identity of any digital platform authority and calls random users to know their PIN number. It's very hard to distinguish because they act as professional as possible to make it seem like an official authority.

There is nothing called enough privacy. And in this age, when the term 'privacy' is thrown around just like another problem, it is not taken seriously as most of us do not think about which part of our information is open to random organizations. Thus in this paper, our motive is to suggest a process of adding a new layer of security as an attempt to fight against any kind of cyber threats on a personal level. This will give ultimate protection against any kinds of hacks during the mobile SIM carrier messages or any other methods.

1.3 Research Objective

This paper examines Digital Transaction security and data theft/loss. We suggest a new layer of security which will be on a personal level for every user. A simple text is sent to the user after any (failed / successful) transaction. This text contains all the major information regarding the account and the transaction. Thus, our suggestion is based on the Cyber Forensic method called Steganography, particularly, image based steganography.

The simple, generic text sent by the bank is encrypted through the AES algorithm. For the AES algorithm to work, it requires a certain Key. For this key, we are going to use the IRIS data of the user. The encrypted data is now passed through LSB embedding and added to the pixels of an image called the Stego Image. The Image will look like just another random image to anyone looking at it. However, the information is encrypted in the image's pixels. The user receives the image instead of the plain text containing all the information. Thus, there will be very little loss of data or data theft / fraud, user privacy is ensured to the fullest.

1.4 Research Gap

This level of security will only be available to the users with smartphones as normal bar phones don't have the capability to run AES encryption and LSB based Image Steganography. Also it is going to add another block of input during opening bank / mobile financial accounts as the users would need to give their IRIS data.

1.5 Challenges Faced

It is worth noting that conducting the whole thesis was quite challenging for us. As the world is recovering from a global pandemic from COVID 19, we could not develop this Security Layer in real life, which was our initial plan. Rather we had to stick to theorizing the processes and methods of this security layer. In addition to that, when we were researching on how to add another layer of security in addition to the AES encryption, we couldn't come to a conclusion but after thorough research we finally found it out.

1.6 Thesis Outline

In Chapter 1, we have discussed our background study and related work where we explained our necessary ideas about different algorithms and technique and also some used architectures.

In Chapter 2, we described the literature review which we have found by researching other papers about out related topics.

In Chapter 3 and 4, Methodology and Research Methodology is given where we discussed our whole procedure and experiments.

In Chapter 5, The Analysis were given which we have found by our thorough research in the methodology part.

In Chapter 6, We depicted the Conclusion.

Chapter 2

Literature Review

Steganography is known as the art and science of concealment. It is the most secure way one can think of when it comes to covert communication. People started using the term "Steganography" in the 15th century after it was mentioned in the Trithemius book on Steganographia. Steganography, which can be loosely translated as "covered writing" or "hidden writing," has been around since 440 BC. The practice of steganography can be traced all the way back to the very beginning. During periods long ago, secret messages were sometimes hidden on the reverse side of wax writing tables, penned on the stomachs of rabbits, or tattooed on the scalps of those who were held in servitude. Microdots and invisible ink have been famous for generations, both for the amusement they provide to youngsters and scholars and for the laborious work they perform for spies, infiltrators, and terrorists. [2]

What makes Steganography different from Cryptography is, In Cryptography, a person is aware that there is a message that can be extracted through an appropriate method. In contrast, Steganography does not let anybody know even the presence of the news, therefore making a piece of information more secure. This thesis paper covers the various aspects of Steganography. The sole focus is on the modern-day application in mobile banking through intelligent devices, which includes maintaining the confidentiality of users' data. Basically, it takes end-to-end encryption to a whole new level. We researched text steganography and how a stego-text will be created by applying an embedding method to a cover-text containing a secret message (or embedded data). Communication channels such as the internet or a mobile device can deliver the message afterwards. The receiver must apply a recovery method parameterized by a stego-key to extract the secret message provided by the sender to recover it. For speech separation, Beza [7], In his paper Secure Mobile Banking Framework by Using Cryptography and Steganography Methods, Mobile banking uses IVR, SMS, WAP, and Independent Smartphone App Customers. Security and cost-effectiveness must be offered on all types of mobile devices. This paper aims to expand a cost-effective and secure Mobile-Banking solution by merging various security algorithmic methodologies. The receiver must apply a recovery method parameterized by a stego-key to extract the secret message delivered by the sender to recover it. Computer science and other related subjects can make extensive use of these methods. They're used to safeguard everything from military communications to emails, credit card numbers, Steganography is known as the art and science of concealment. It is the most secure way one can think of when it comes to covert communication. People started using the term "Steganography" in the 15th century

after it was mentioned in the Trithemius book on Steganographia. Steganography means covered or hidden writing, the existence of which dates back to 440 BC. The use of Steganography has always been there from the beginning. In ancient times, messages were concealed on the back of wax writing tables, written on the stomachs of rabbits or tattooed on the scalps of enslaved people. Invisible ink microdots have always been famous for centuries for fun by children students and for backbreaking work by spies, infiltrators terrorists [2].

What makes Steganography different from Cryptography is, In Cryptography, a person is aware that there is a message that can be extracted through an appropriate method. In contrast, Steganography does not let anybody know even the presence of the news, therefore making a piece of information more secure. This thesis paper covers the various aspects of Steganography. The sole focus is on the modern-day application in mobile banking through intelligent devices, which includes maintaining the confidentiality of users' data. Basically, it takes end-to-end encryption to a whole new level. We researched text steganography and how a stego-text will be created by applying an embedding method to a cover-text containing a secret message (or embedded data). Communication channels such as the internet or a mobile device can deliver the message afterwards. The receiver must apply a recovery method parameterized by a stego-key to extract the secret message provided by the sender to recover it. For speech separation, Beza [7], in his paper Secure Mobile Banking Framework by Using Cryptography and Steganography Methods, claims Interactive Voice Response (IVR), SMS, WAP (Wireless Access Protocol), and Standalone Mobile Application Clients are some of the ways that mobile banking is being implemented. Security and cost-effectiveness must be offered on all types of mobile devices. This paper aims to expand a cost-effective and secure Mobile-Banking solution by merging various security algorithmic methodologies. The receiver must apply a recovery method parameterized by a stego-key to extract the secret message delivered by the sender to recover it. Computer science and other related subjects can make extensive use of these methods. They're used to safeguard everything from military communications to emails, credit card numbers, the picture's URL by SMS, the user uses a specific application to download the image. If the password is given accurately, the user will be able to see the information extracted from the image. Because the data is delivered at the user's request, hackers may get access to and leak the user's personal information during transmission. Instead of providing information directly, the author recommends hiding it in a password-protected photo and putting it on a different website. It is then replaced with a picture's URL delivered to the user. The picture's address is received by a unique program already installed on the user's mobile phone. The image holding the secret information is then downloaded from the internet and shown to the user once the private information has been extracted using a password and a steganography method by this program. Customer information is protected by using the same password they used to sign up for an account in the banking system. This same password is also used to encrypt the photo. The following is his method for concealing information in images: To cover data in the pixels' least significant bits (LSB), he employed LSB steganography in this endeavour. Every bit of information is concealed within two pixels when using this technique. A byte is divided into eight bits to hide information, which is done to improve security. Two pixels are picked, and a byte of data is masked using a password.

After that, the customer receives the picture's URL instead of the requested data. This address is received by a unique program on the customer's phone, which we refer to as a "decoder" and this address is used to download the photo. Once the picture has been downloaded, the application automatically disconnects from the internet. The decoder program uses the password the user provided and the previously discussed algorithm to extract the data from the image. Customers have presented the correct information if the password is correctly entered. This project's implementation can be divided into two stages: First, the server should prepare the user's request by executing a coding program, disguising information in a picture, and then delivering the picture's address to the customer's mobile phone so they can view it. Secondly, Sending a request, obtaining a photo's address, downloading the image and extracting its metadata are all part of this area for the user.

In [11], Lu et al. A novel design has been proposed with the goal of simplifying the AES algorithm employed in Cryptography and Network Security 2017 when it is implemented on hardware such as mobile devices and smart cards, both of which are utilized in this context. After taking into account a great number of other factors, he devised a variety of data types and key sizes that might be applied in the field of cryptography.

John Daugman was the first to suggest and implement iris recognition as a biometric feature. Systematic extraction, representation, and comparison of iris surface texture information are the goals of an iris recognition system (IRS). Yang et al. [12] Propose cancelable iris and steganography-based strong authentication solutions. Most existing interruptible iris biometric systems require a user-specific key to direct feature transformation. If this key is obtained, attackers can leak relevant information and represent the actual iris feature data. The study improves system security by using Steganography to hide information. This boosted overall system security by masking the user-specific key. The report suggested utilizing fundamental purpose to mask the secret key. Chai et al. [8]

A. Soria-Lorente and S. Berres generated a new approach to the Entropy Thresholding method [5]. A computer displays a pixelated digital image. This paper proposes a steganography algorithm to hide a message in a 24-bit RGB image. RGB is a byte. 0 is darkest, 255 brightest. The m -byte cover image C is divided into $3mn/64$ 88-byte blocks for DCT. Combining JPEG steganography and Entropy Thresholding. This method hides secret messages in photographs with minimum detectability. JPEG breaks the cover image into 8×8 parts. The discrete cosine transform then hides the message by modifying low-frequency cover image coefficients. The Entropy Thresholding approach incorporates a secret message based on the matrix's entropy. Matlab is used to implement the proposed technique. A quality factor of $qF = 57$ is applied to five photos with a resolution of 784×512 pixels. Randomly selected keys were utilized. Innovative steganographic algorithms that use public and private keys to minimise detection have been proposed in this article. Using PSNR, IQMs, and histograms, it is concluded that the stego image does not include any anomalies that may be detected. Because the proposed approach yielded a relative entropy value of 8, it is safe to use the steganographic system that was generated. Embedding probability also showed that the suggested approach is very resistant to a Chi-square assault, which was previously unknown. The idea of DNA steganography has been proposed to circumvent the constraints of picture steganography [11].

The image has a limited capacity for storing data and cannot conceal a large amount of information. DNA insertion, replacement, and complementary algorithms can all be used to hide sensitive information from prying eyes. In these three methods, the DNA insertion method has the lowest chance of breaking the information contained within the DNA sequence. Papers like this one explain how to tweak DNA insertion algorithms in order to make false DNA sequences less likely to be cracked.

This research demonstrates that AITSteg [6], a new text steganography system, allows end-to-end security for text conversations carried by SMS or social media between end-users. Using a credible situation, the effectiveness of AITSteg is evaluated. The efficiency of the proposed method is then determined through an evaluation of embedding capacity, invisibility, robustness, and security. According to the test results, the AITSteg prevents message leakage, man-in-the-middle attacks, and reader manipulation. In addition, we compare our experimental results to those of rival approaches to establish our superiority. It appears that text steganography leveraging symmetric keys via social media is the first technology of its kind to provide end-to-end security for message delivery. This section evaluates the effectiveness of the proposed strategy in light of the evaluation criteria. AITSteg was developed in Java, and Android devices were utilized for the experiments. This work introduces AITSteg, a novel text steganography technique for the secure transmission of text messages over SMS or social media between smartphone users. The embedding trace is completely imperceptible to readers due to the capacity of the proposed technology to conceal enormous quantities of secret information within a small cover message. It is also feasible to employ symmetric key approaches in conjunction with mathematical encoding to generate alternative private bits for the same sensitive information at different periods in time, thereby protecting the data from external intrusion. The examination of the AITSteg technique demonstrates that it can repel a wide variety of attacks. In addition to these benefits, the new approach is also undetectable, more stable than current technologies, and has superior electrical conductivity (EC). If all goes according to plan, this approach provides a certain level of protection for covert talks conducted via standard social media or messaging platforms.

Chapter 3

Methodology

3.1 Working Process

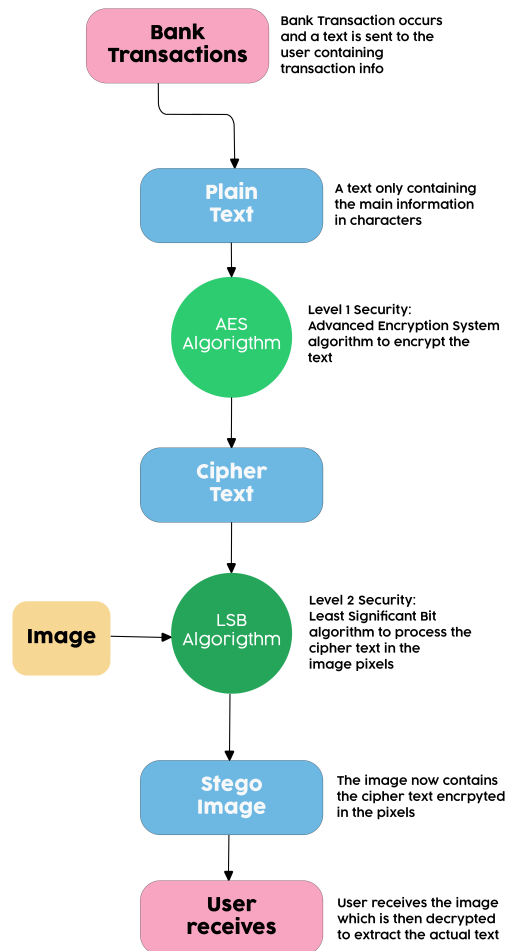


Figure 3.1: Full working process

3.2 Used Architecture

3.2.1 AES

The Advanced Encryption Standard is a highly secure and one-of-a-kind encryption technique that employs many processes for various rounds based on the size of the key. To ensure security, substitution and permutation networks are utilized. It is a set of operations used in block cipher algorithms that are called SPN. When AES is used to encrypt plain text or content that has to be encrypted, it hides the real text or data using four different approaches and procedures. The four different steps of the encryption are:

1) SubBytes: During this stage, each byte will have a different byte replaced with it. A mapping table, which is also known as the Substitution box, is utilized in order to carry out the procedure. Because of the way that this substitution is carried out, a byte will never be replaced by itself, nor will it ever be replaced by another byte that is the complement of the byte that is currently being used. The completion of this phase yields the same 16-byte (4 x 4) matrix that was produced in the previous stage. The S-box, which is responsible for storing the data necessary to carry out the substitution, is discreetly concealed and does not appear anywhere at any point in the process of these procedures. As per read in this article.**14**

```
○○○
s_box = (
0x63, 0x7C, 0x77, 0x7B, 0xF2, 0x6B, 0x6F, 0xC5, 0x30, 0x01, 0x67, 0x2B, 0xFE, 0xD7, 0xAB, 0x76,
0xCA, 0x82, 0xC9, 0x7D, 0xFA, 0x59, 0x47, 0xF0, 0xAD, 0xD4, 0xA2, 0xAF, 0x9C, 0xA4, 0x72, 0xC0,
0xB7, 0xFD, 0x93, 0x26, 0x36, 0x3F, 0xF7, 0xCC, 0x34, 0xA5, 0xE5, 0xF1, 0x71, 0xD8, 0x31, 0x15,
0x04, 0xC7, 0x23, 0xC3, 0x18, 0x96, 0x05, 0x9A, 0x07, 0x12, 0x80, 0xE2, 0xEB, 0x27, 0xB2, 0x75,
0x09, 0x83, 0x2C, 0x1A, 0x1B, 0x6E, 0x5A, 0xA0, 0x52, 0x3B, 0xD6, 0xB3, 0x29, 0xE3, 0x2F, 0x84,
0x53, 0xD1, 0x00, 0xED, 0x20, 0xFC, 0xB1, 0x5B, 0x6A, 0xCB, 0xBE, 0x39, 0x4A, 0x4C, 0x58, 0xCF,
0xD0, 0xEF, 0xAA, 0xFB, 0x43, 0x4D, 0x33, 0x85, 0x45, 0xF9, 0x02, 0x7F, 0x50, 0x3C, 0x9F, 0xA8,
0x51, 0xA3, 0x40, 0x8F, 0x92, 0x9D, 0x38, 0xF5, 0xBC, 0xB6, 0xDA, 0x21, 0x10, 0xFF, 0xF3, 0xD2,
0xCD, 0x0C, 0x13, 0xEC, 0x5F, 0x97, 0x44, 0x17, 0xC4, 0xA7, 0x7E, 0x3D, 0x64, 0x5D, 0x19, 0x73,
0x60, 0x81, 0x4F, 0xDC, 0x22, 0x2A, 0x90, 0x88, 0x46, 0xEE, 0xB8, 0x14, 0xDE, 0x5E, 0x0B, 0xDB,
0xE0, 0x32, 0x3A, 0x0A, 0x49, 0x06, 0x24, 0x5C, 0xC2, 0xD3, 0xAC, 0x62, 0x91, 0x95, 0xE4, 0x79,
0xE7, 0xC8, 0x37, 0x6D, 0x8D, 0xD5, 0x4E, 0xA9, 0x6C, 0x56, 0xF4, 0xEA, 0x65, 0x7A, 0xAE, 0x08,
0xBA, 0x78, 0x25, 0x2E, 0x1C, 0xA6, 0xB4, 0xC6, 0xE8, 0xDD, 0x74, 0x1F, 0x4B, 0xBD, 0x8B, 0x8A,
0x70, 0x3E, 0xB5, 0x66, 0x48, 0x03, 0xF6, 0x0E, 0x61, 0x35, 0x57, 0xB9, 0x86, 0xC1, 0x1D, 0x9E,
0xE1, 0xF8, 0x98, 0x11, 0x69, 0xD9, 0x8E, 0x94, 0x9B, 0x1E, 0x87, 0xE9, 0xCE, 0x55, 0x28, 0xDF,
0x8C, 0xA1, 0x89, 0x0D, 0xBF, 0xE6, 0x42, 0x68, 0x41, 0x99, 0x2D, 0x0F, 0xB0, 0x54, 0xBB, 0x16,
)
```

Figure 3.2: A possible substitution box

2)ShiftRows: This step is exactly what it sounds like it would be. An exact number of times is applied to each row. In the first row, no shift is made; in the second row, one shift is made; in the third row, two shifts are made; in the fourth row, three shifts are made. In most situations, a left circular shift is carried out. [16]

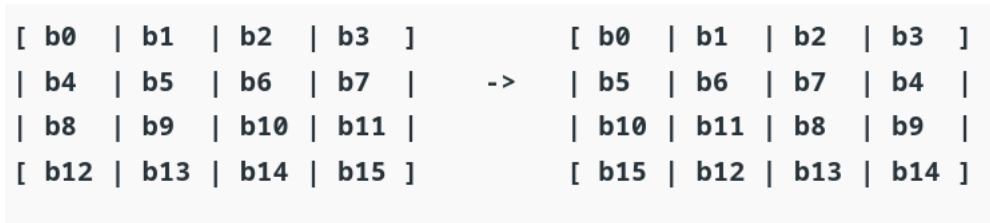


Figure 3.3: Shifting Rows

3) Mix-Columns: This stage consists primarily of multiplying matrices together. Since each column is multiplied by its own unique matrix, the position of each byte within the column is displaced as a direct result of this operation. This shift occurs because each column is multiplied by its own unique matrix. In the last round, this step was skipped.[16] The polynomial equation which is used to derive this multiplication is such:

$$\begin{aligned}
 (1) \quad a(x) \bullet b(x) = c(x) &= (a_3x^3 + a_2x^2 + a_1x + a_0) \bullet (b_3x^3 + b_2x^2 + b_1x + b_0) \\
 (2) \quad &= c_6x^6 + c_5x^5 + c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0
 \end{aligned}$$

Figure 3.4: Equation for multiplication of columns

4) AddRoundKeys: this might be the most interesting aspect of the whole thing. The AES algorithm will repeat this entire procedure multiple times, although the number of iterations will vary depending on the size of the key. To be more specific, a key with 128 bits will do 10 rounds, a key with 192 bits will perform 12 rounds, and a key with 256 bits will utilize 14 rounds. In each cycle, a unique iteration of the initial key is produced as the starting point. The "AES Key Schedule" program is what is used to produce such a key. In accordance with this timeline, a single round key will be expanded into several additional round keys. There is a disparity in the total number of rounds across the three distinct AES implementations. Each version calls for a distinct 128-bit round key for each round, in addition to one additional key. The initial key is used as the basis for the production of all of the necessary round keys by the key schedule. [16]

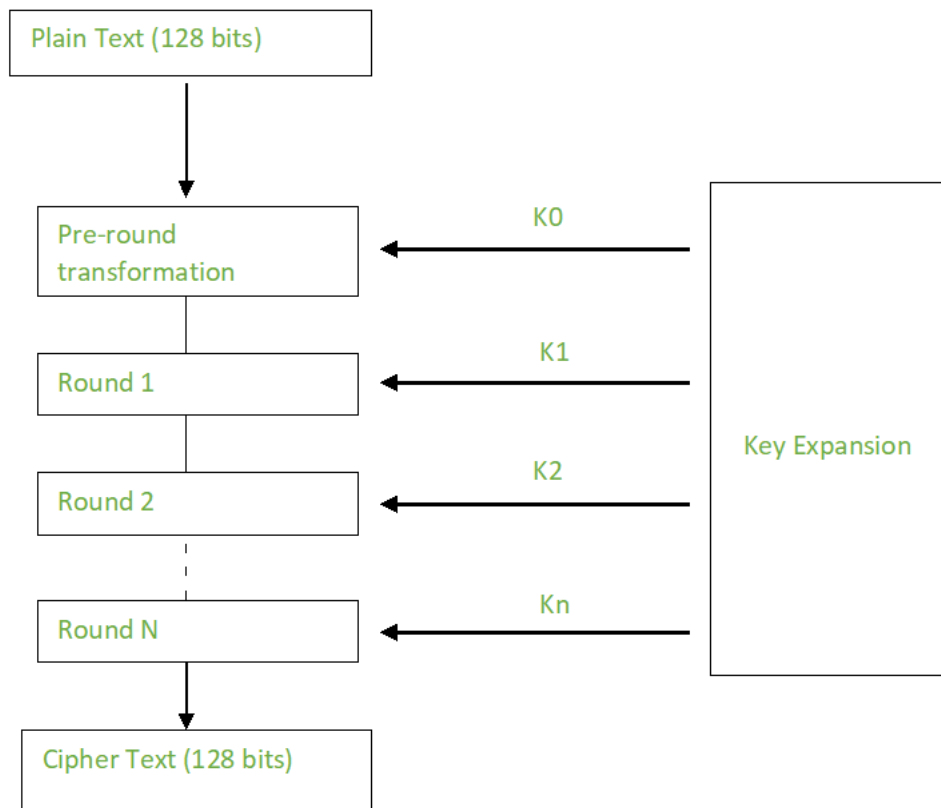


Figure 3.5: Variation of Round Keys

These are the four stages executed 10/12/14 times (depending on the size of the key) to encrypt a plaintext. To decrypt, i.e., to convert the ciphertext back to the original plaintext, these stages are repeated the same number of times as in the encryption process, but in the reverse sequence, as this is how decryption should function. In the below figures the whole process is depicted:

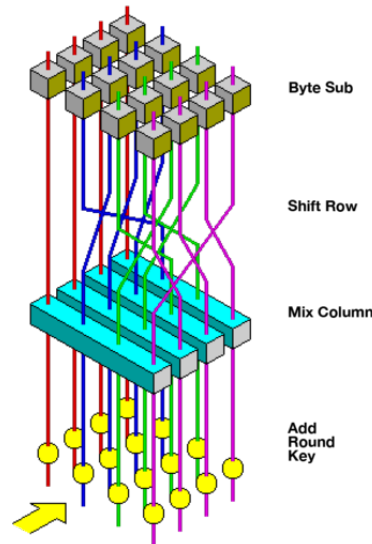


Figure 3.6: AES Algorithm

3.2.2 LSB

LSB is a fundamental way for embedding data in a cover file. Steganography uses LSB substitution. Each image includes three parts (RGB). The information concerning this pixel is stored within a single byte. By modifying the first bits of data for every pixel, it is possible to add concealed text. Before proceeding, there must be a smaller or equal size difference between the text being stored and that which will be covered by the image. Pixels are known to be stored as separate bits. Grayscale images store each pixel's intensity value using eight bits (1 byte). Also, each pixel needs 24 bits to produce a color image (RGB). LSB image steganography entails the following procedures: The steps for concealing the image of the message are as follows:

- 1)The book's cover art should be studied. As a result of including the message image, it is simpler to conceal the modifications that have been made.
- 2)The image that will be used to convey the information should be scrutinized attentively.

For each image, generate independent bit plane layers. It's ideal to use the picture's upper four bitplanes to replace up to four of the cover image's least relevant bitplanes.[3] Using fewer bitplanes of the message image might distort and lose data from the returned image. Cover image bitplanes must match the four most significant bitplanes of the message image. Recombining bitplanes creates the steganographic image. Each grayscale pixel has eight bits. "Least Significant Bit" is the

final pixel bit that doesn't affect the pixel's value. This feature hides image data. Using Least Significant Bits, only "3" changes would be implemented. There's more storage. Steganography involves replacing an image's least significant bit with data (LSB). To prevent steganalysis, we encrypt raw data before embedding it. Even though encryption takes longer, security improves. It's really easy! Secret bits are replaced for least significant bytes in an image. LSB embedding is used to hide messages in multimedia. LSB embedding can encrypt communications in specific data domains, like RGB bitmap data or JPEG frequency coefficients. LSB embedding supports several data kinds and formats. LSB embedding is a popular but an old model steganographic approach. [3]

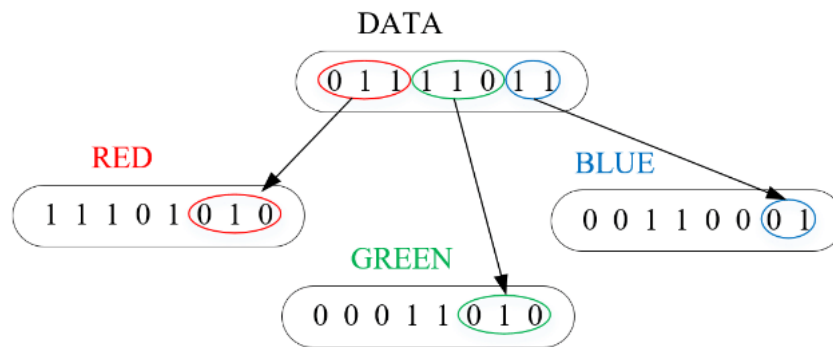


Figure 3.7: LSB Technique

3.2.3 Steganography

Steganography involves concealing a secret message such that it appears to be part of the surrounding environment or even superimposed on top of it. It is fine to use whatever you want if you want to make a substitution for something else. The insertion of a hidden text message into an image is utilized in a significant number of steganographic techniques used today. The use of steganography can be thought of as serving both the objective of hiding and that of deception. It is a form of covert communication in which messages are transmitted while remaining hidden through the use of any medium. The fact that the data is not encrypted and there is no use of a key is what distinguishes this from cryptography. Instead, it is a strategy for the stealthy concealing of data that is easy to put into action. Steganography, in contrast to cryptography, is a procedure that makes it possible to conceal information while also lying about it. It is necessary to make use of a particular key, technique, or table in order to encode the data into a given medium.

Additionally, it is necessary to make use of these same features in order to decode the primary data. [15]

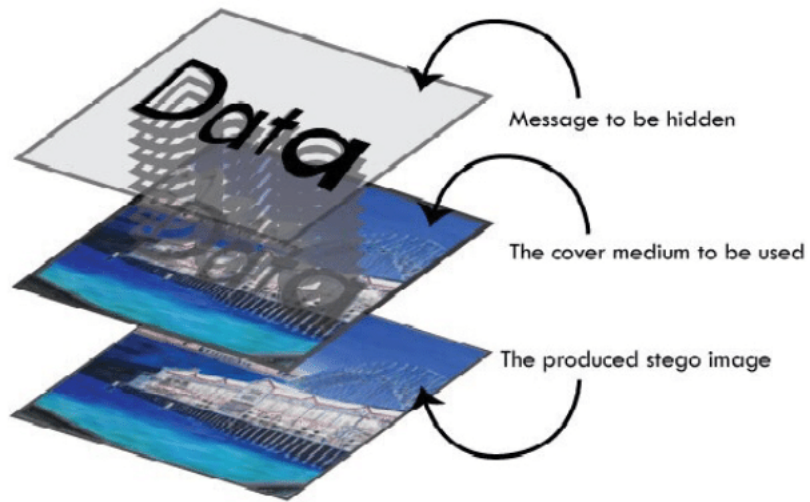


Figure 3.8: Image Steganography

Chapter 4

Research Methodology

4.1 Data Acquisition

In this section, we propose acquiring the user's information. We have chosen the IRIS scanner data that will be taken from a particular user; this data is extremely sensitive and must be protected with great care. The acquisition can be expensive, but we will only need it once when a consumer establishes a bank account. We will then need to protect it in a highly secure manner so that it is not stolen. We chose this data because it is extremely unique biometrics data of a human and varies from person to person, thus the likelihood of it matching with another one is almost zero.

Iris recognition is a type of automated biometric authentication that involves complex pattern-recognition methods on video evidence of one or both of a person's irises, which have unique, stable, and easy-to-see complicated shapes. As it is a multi-step procedure, data collection can be expensive and time-consuming. The bank must have an IRIS scanner to capture a high-contrast image of the individual's eye. Several processes are then taken to convert the image to a machine-readable binary code, which is then securely stored in the database for use in future transactions between the bank and the user.[10] The procedures are outlined below:

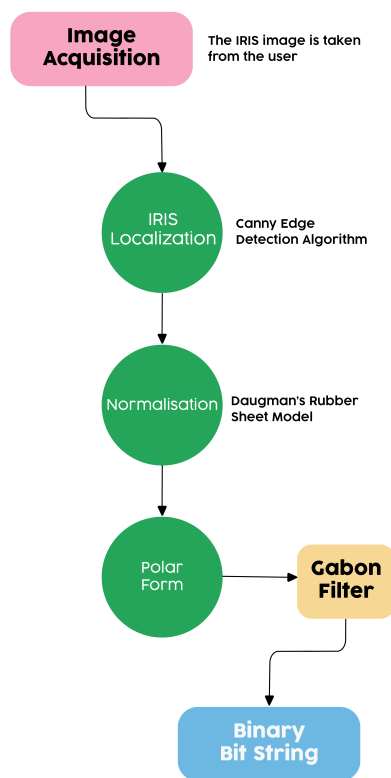


Figure 4.1: Acquiring the IrisCode

As we can see in the above figure the whole procedure, initially we need a very high contrast photo of the user’s eye, to be more precise the scanner takes 240 samples to be more and more accurate.[18] After that, an algorithm is used which is “Canny Edge Detection” in order to locate only the iris from the other parts of the eye such as eyelashes, pupils, etc.

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i - (k+1))^2 + (j - (k+1))^2}{2\sigma^2}\right); 1 \leq i, j \leq (2k+1)$$

Figure 4.2: Equation of Gaussian Filter

To avoid false detections caused by noise in the image, edge detection results must be filtered to remove them from the image. The image is convolved with a Gaussian filter kernel to smooth it out.[18] Image noise on the edge detector is reduced slightly by doing this step, which smooths out the image. The above equation is used to filter out the iris as perfectly as it can.

After this process, Daugman’s Rubber Sheet model is used to normalize the data. In order to determine the phase structure of the iris, the technique developed by Daugman applies a 2D Gabor wavelet transform.[11] IrisCode is used to compress this information into a relatively condensed bitstream.

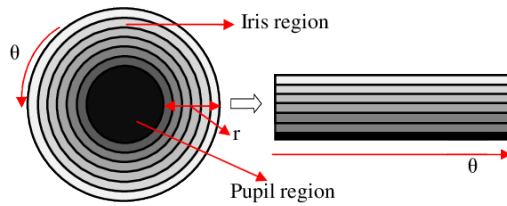


Figure 4.3: Polar form after normalization

The data is now ready to be converted into a binary bitstream after normalization. From the polar form of the normalization, a technique called “Gabor Filter” is used to get the IrisCode. Which is then saved in a secure database for later comparison to determine the user’s identity. The bank will gather the IRIS biometric data as part of the aforesaid procedure. The AES algorithm’s key will be generated from this data, according to our proposal.

4.2 Workflow

During the phase devoted to the collection of data, we reviewed and illustrated the process by which we will extract the information from an iris and convert it into a bitstream of binary code. This binary code will serve as the key for the AES algorithm that we are using. But first, we have to convert that binary code into text format. In order to do this, we need to be aware of how many binary bits we are receiving; as was mentioned earlier, this could be a multiple of 16 bits. Since we are aware that 8 bits are sufficient to create any ASCII value character.

if the IrisCode is 16 bits ($n = 1, 2, 3, 4, \dots$), then the number of bits required to convert that into text format will be $16n * 2$ bits, which equals $32n$ bits To convert the binary bits to the decimal format we can use the below code:

```
def BinaryToDecimal(binary):
    binary1 = binary
    decimal, i, n = 0, 0, 0
    while(binary != 0):
        dec = binary % 10
        decimal = decimal + dec * pow(2, i)
        binary = binary//10
        i += 1
    return (decimal)

bin_data = '10001111100101110010111010111110011'
```

Figure 4.4: Binary conversion

Then we can change the decimal value to its ASCII value to get the string that will be used as a key for the AES algorithm. We've previously learned how the AES algorithm works, but now we must determine where to apply our IrisCode. The algorithm will use the plain text string obtained from the binary bits. The "AddRoundKey," or the fourth phase in the AES method, will eventually be altered by adding salt to each round. The "PBKDF2," a password-based key derivation function, alters the original key data with each round by using a new round key in contrast to the original. Encryption and decryption are demonstrated using the AES algorithm, which is seen in the following code snippets:

```
def encrypt(plain_text, key):
    private_key = hashlib.sha256(key.encode("utf-8")).digest()
    plain_text = pad(plain_text)
    print("After padding:", plain_text)
    iv = Random.new().read(AES.block_size)
    cipher = AES.new(private_key, AES.MODE_CBC, iv = iv)
    return base64.b64encode(iv + cipher.encrypt(plain_text))
```

Figure 4.5: AES encryption function

```

def decrypt(cipher_text, key):
    private_key = hashlib.sha256(key.encode("utf-8")).digest()
    cipher_text = base64.b64decode(cipher_text)
    iv = cipher_text[:16]
    cipher = AES.new(private_key, AES.MODE_CBC, iv)
    return unpad(cipher.decrypt(cipher_text[16:]))

```

Figure 4.6: AES decryption function

These two functions take plain text and convert it into cipher text by using the key that we proposed for the IrisCode encryption algorithm. The string that we get after our binary to decimal function's output will be used as the key. The below snippets show how the inputs will be required and what output will be given or stored as necessary:

```

message=input("Enter message to encrypt: ");
key = input("Enter encryption key: ")
encrypted_msg = encrypt(message, key)
print("Encrypted Message:", encrypted_msg)
decrypted_msg = decrypt(encrypted_msg, key)
print("Decrypted Message:", bytes.decode(decrypted_msg))

```

Figure 4.7: inputs

Now that we have the message that has been encrypted and can only be decrypted with the encryption key, we are able to proceed to the next step of our process, which is to conceal this ciphertext within an image by making use of steganography. Once this step is complete, we will have successfully completed our process. The encoding and decoding functions that hide data within a picture are demonstrated in the following code snippets. These functions can later be used to extract data from an image that has already been encoded:

We will be able to see the identical image after the encoding process, but only after the decoding process will we be able to see the concealed image. And the bank will only send the image that has been decoded, which the user will then get, and after they have provided the key, they will be able to view the transaction's initial text. Tkinter is a Python package(a Graphics User Interface tool) that was used[36] to illustrate the scenario that will play out behind the scenes of the entire operation.

```
○ ○ ○  
  
/bin/python /home/syedmh/Documents/Thesis/ThesisPro/aes2  
  
Enter message to encrypt: thisIsTheString  
Enter encryption key: 123987  
After padding: thisIsTheString  
  
Encrypted Message: b'r3V0A0Ssjw/4ZOKL42/hWSQOLKy7lt9b0Vt7D75RA3E='  
Decrypted Message: thisIsTheString
```

Figure 4.8: Outputs of the ciphered text

```
○ ○ ○  
  
def encode():  
    img = input("Enter image name(with extension) : ")  
    image = Image.open(img, 'r')  
  
    data = input("Enter data to be encoded : ")  
    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()))
```

Figure 4.9: Encoding function for Image steg

```
○ ○ ○  
  
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__()[0:3] +  
                  imgdata.__next__()[0:3] +  
                  imgdata.__next__()[0:3]]  
  
        binstr = ''  
  
        for i in pixels[:8]:  
            if (i % 2 == 0):  
                binstr += '0'  
            else:  
                binstr += '1'  
  
        data += chr(int(binstr, 2))  
        if (pixels[-1] % 2 != 0):  
            return data
```

Figure 4.10: decoding function



Figure 4.11: Graphical image of encoding

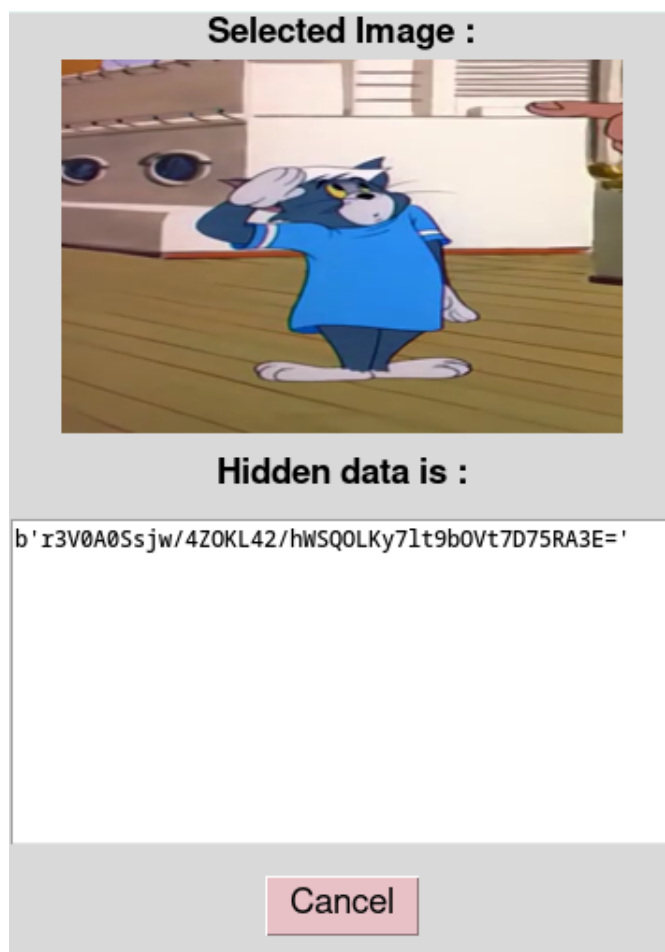


Figure 4.12: Graphical image of decoding

Chapter 5

Result Analysis

5.1 AES Analysis

It's going to take a long time to break the AES encryption algorithm. It's because brute force attacks are the only means for hackers to break into a system, and brute force is the only technique for a hacker to attack. For a better idea of how long it would take a hacker to get access via brute force, see the table below:

Computing power	Average time to crack using exhaustive search
High-end PC	27,337,893,038,406,611,194,430,009,974,922,940,323,611,067,429,756,962,487,493,203 years. 27 trillion trillion trillion trillion trillion years
Fastest supercomputer	27,337,893,038,406,611,194,430,009,974,922,940,323,611,067,429,756,962,487 years. 27,337,893 trillion trillion trillion trillion years
2 billion high-end PCs	13,668,946,519,203,305,597,215,004,987,461,470,161,805,533,714,878,481 years 13,689 trillion trillion trillion trillion years
Age of the universe	15,000,000,000 years 15 billion years

Figure 5.1: Chart for Cracking AES

The bottom line is that if there was any way that AES could be broken, everything in the world would come to a stop. It is generally agreed that there is not much of a Differences between the AES-128 and the AES-256 methods in terms of method breaking. If an innovation can crack 128 bits, it is likely to be able to crack 256.[17]

The Advanced Encryption Standard (AES) has never been cracked, and it is resistant to any type of brute-force assault, despite the widespread misconception that it has been, as well as multiple persuasive pieces of evidence to the contrary. Despite the fact that the processing speeds of current computers have risen over the years, the numbers of the key that is used for encryption should always be sufficiently large enough so that it cannot be broken by modern computers. In our particular scenario, the key of the AES is comprised of the binary bit string obtained from the IRIS scan.

After conducting in-depth research, we have come to the conclusion that by including the IRIS data into the process, we will be able to make it far more secure than if we had only relied on AES encryption. The length of the key used in an AES determines the level of security provided by the algorithm. A brute-force attack requires 2^{128} attempts to crack an encryption with a key having 128 bits, hence the number of possible combinations is limited. If the length of the key is n bits, then the security level of AES is also n bits, and a brute-force attack will take 2 to the power of n tries.

5.2 IRIS Security Analysis

However, the problem lies in the fact that this AES algorithm is not personalized. It is the same for everyone. But adding the IRIS binary data as the key personalizes the whole security structure. It is almost impossible to replicate one's IRIS data with another one. In the course of a person's life, IRIS remains the same in appearance. Thus, IRIS scan is different and distinctive from all other biometric technologies. The probability of two people having identical IRIS patterns is $1/10^{78}$, which is very minimal. Thus we can say that the IRIS data is personalized and more secure than other forms of biometric methods.

5.3 AES versus Other Encryption Models

Now we will dive deep into how AES compares against other notable Symmetric Encryption Models.

5.3.1 By the factor of Decryption

	AES	Serpent	Twofish	64-bit	DES
Key Length	128/192/256	128/192/256	128/256	64	56
Possible Combinations	$3.4 \times 10^{38} / 6.2 \times 10^{57} / 1.1 \times 10^{77}$			1.8×10^{19}	7.6×10^6
Time to crack	Up to 3.31×10^{56} years			65 minutes	399 seconds

Figure 5.2: Chart for comparison of AES against other Encryption methods

We can clearly analyze that using DES or 64-bit encryption key would make our data vulnerable to hackers, as they can easily crack it.[19]

5.3.2 By the factor of Time

This is the chart for AES rounds against Serpent and Twofish

	AES	Serpent	Twofish
Key Length	128/192/256	128/192/256	128/256
Encryption Steps Traversed	4 distinct steps for each 10/12/14 rounds	2 distincts steps for each 32 rounds	same like 'Serpent' with an extra "round function"

Figure 5.3: Chart for comparison of AES rounds against Serpent and Twofish

From the above table, we can note that Serpent and Twofish both traverse 32 rounds of 2 steps: SubBytes and Key Mixing X-OR. On the other hand, AES traverses 10/12/14 rounds, which is significantly lower than that of Serpent and Twofish. But AES confirms another extra 2 steps which are MixColumns, ShiftRows (non-linear mixing) in addition to the 2 steps done by Serpent and Twofish.[19] Due to the amount of rounds being considerably lower, AES requires less time for hardware execution than Serpent and Twofish.

Chapter 6

Future Work and Conclusion

6.1 Future Work

We want to implement the following changes to our initial proposal in our future work: When a user creates a bank account and the bank authority scans their IRIS data, an app is downloaded to their device that contains the information needed to successfully decrypt the stego picture provided for each transaction. But, in order to secure it and ensure that it does not fall into the hands of anybody other than the user, we propose that a private key be generated at the time of app installation and kept secure only by the user's access within the app.

End-to-end encryption will be implemented for each transaction, or for each random stego picture sent from the bank authority to the user, to prevent the inner data from being stolen. Only the user and bank connection will know the secret key, which will be untraceable even by the mobile carrier.

6.2 Conclusion

To conclude, online banking, mobile banking, and other non-traditional banking methods have become equally important, if not more so, than traditional banking systems. It is vital to preserve the safety and security of the transactions that take place throughout the day by a big number of people since these transactions take place multiple times at different times. There are already potent ideas that have been established or even employed to secure it, and these ideas have been thoroughly verified. We intend to make it even better by adding the functionality that was mentioned in greater detail up above. This is our current plan. We are keeping our fingers crossed that the IRIS scan, a method that requires the extremely unique characteristics of a single individual amid a great number of others, will be effective. It has also been demonstrated through medical research that our retinas and irises are unique to each of us. Our objective is to make it more secure by making the algorithm more robust by testing an increasing number of datasets. This is done so that it can be demonstrated and tested that this proposed method of ours can be applied by all banks, thereby increasing the security of online banking transactions.

Bibliography

- [1] T. Morkel, J. H. Eloff, and M. S. Olivier, "An overview of image steganography.," in *ISSA*, vol. 1, 2005, pp. 1–11.
- [2] U. Rizwan and H. F. Ahmed, "A new approach in steganography using different algorithms and applying randomization concept," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 1, no. 9, 2012.
- [3] S. Bhallamudi, *Image steganography*, Dec. 2015. DOI: 10.13140/RG.2.2.21323.18727.
- [4] A. Abdullah, "Advanced encryption standard (aes) algorithm to encrypt and decrypt data," Jun. 2017.
- [5] A. Soria-Lorente and S. Berres, "A secure steganographic algorithm based on frequency domain for the transmission of hidden information," *Security and Communication Networks*, vol. 2017, 2017.
- [6] M. T. Ahvanooy, Q. Li, J. Hou, H. D. Mazraeh, and J. Zhang, "Aitsteg: An innovative text steganography technique for hidden transmission of text message via social media," *IEEE Access*, vol. 6, pp. 65 981–65 995, 2018.
- [7] T. Beza, "Secure mobile banking frame work by using cryptography and steganography methods," *Global Scietific*, 2018.
- [8] T.-Y. Chai, B.-M. Goi, Y.-H. Tay, and Z. Jin, "A new design for alignment-free chaffed cancelable iris key binding scheme," *Symmetry*, vol. 11, no. 2, p. 164, 2019.
- [9] "Electronic frontier foundation." (Oct. 2019), [Online]. Available: <https://www.eff.org/pages/iris-recognition>.
- [10] "Iris recognition, electronic frontier foundation." (Oct. 2019), [Online]. Available: <https://www.eff.org/pages/iris-recognition>.
- [11] H. Rana, M. Azam, M. R. Akhtar, J. Quinn, and M. A. Moni, "A fast iris recognition system through optimum feature extraction," *PeerJ Computer Science*, vol. 5, e184, Apr. 2019. DOI: 10.7717/peerj-cs.184.
- [12] W. Yang, S. Wang, J. Hu, *et al.*, "A cancelable iris-and steganography-based user authentication system for the internet of things," *Sensors*, vol. 19, no. 13, p. 2985, 2019.
- [13] "Banking security: A crucial aspect for the finance industry." (Apr. 2020), [Online]. Available: <https://www.analyticsinsight.net/banking-security-a-crucial-aspect-for-the-finance-industry/>.

- [14] “What is encryption and how does it work?” (Apr. 2020), [Online]. Available: <https://www.techtarget.com/searchsecurity/definition/encryption>.
- [15] “What is steganography? - definition from searchsecurity.” (Jul. 2021), [Online]. Available: <https://www.techtarget.com/searchsecurity/definition/steganography>.
- [16] “Advanced encryption standard (aes).” (Feb. 2022), [Online]. Available: <https://www.geeksforgeeks.org/advanced-encryption-standard-aes/>.
- [17] “How long would it take to brute force aes-256?, scrambox.” (Jan. 2022), [Online]. Available: <https://scrambox.com/article/brute-force-aes/?fbclid=IwAR0Jv683vW4m3OxelhrUUy8QlSTlxbkecai7aSRod-NlkBGmI7U7wpufWko>.
- [18] “Iris scanner: How it works: Eye scan technology overview.” (Jan. 2022), [Online]. Available: <https://refaces.com/articles/iris-scanner>.
- [19] “Aes: Who won?” (), [Online]. Available: <https://www.infoworld.com/article/2076215/aes--who-won-.html>.