Enhancing security efficiency through the integration of Elliptic Curve Cryptography with Audio Steganography

MSc Research Project
MSc in Cybersecurity

Nishant Bhosale
Student ID: 22227377

School of Computing
National College of Ireland

Supervisor: Khadija Hafeez

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: …… Nishant Sandeep Bhosale. ………………………………………………………

Student ID: …22227377……………………………………………………….……

Programme: … MSc In Cybersecurity ……………………… **Year:** 2023-2024..

Module: …MSc Research Project Practicum 2……………………….………

Supervisor: …… Khadija Hafeez ……………………………………………………….………

Submission Due Date: ……12/8/24…………………………………………………………………….………

Project Title: Enhancing security efficiency through the integration of Elliptic Curve Cryptography with Audio Steganography……….………

Word Count: ……8771……………………….. **Page Count: 22**…………..……..

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.

Signature: N Bhosale

Date: …12/8/24………………………………………………………………………………………

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 efficiency through the integration of Elliptic Curve Cryptography with Audio Steganography

Nishant Bhosale
22227377

Abstract

This research combines Audio Steganography with Elliptic Curve Cryptography (ECC), focusing on the efficiency part of security to handle the performance problems in standard RSA encryption. ECC is a possible substitute because it has smaller key sizes and requires less operations. The study hopes to create a system that uses ECC's benefits in a many-layered security structure through audio steganography.

The objective of this research is to design a security app combining ECC with audio steganography which will offer faster processing and low computational power. The method includes implementing ECC algorithms in Python, using multilayer Least Significant Bit embedding for data covering up in audio files. Expected outcomes will include improved encryption efficiency, better data security measures, for secure data transmission and storage.

1 Introduction

The research question was how the use of ECC along with audio steganography would help to improve the efficiency of current techniques of encryption in real time. Since the beginning, the concept was on the use of asymmetric algorithms, specifically the RSA encryption method to be used as a shield for the data. RSA has always been a very first layer of security that makes use of the public and private keys for encryption and decryption purposes respectively. This technique was very successful in hiding text data. This made sure that in the worst-case scenario of somebody actually discovering the audio file hidden, they would have no idea about the nature of the message in it. This application, along with RSA encryption and audio steganography, added one more step for security so that no other person could easily access the sensitive data.
But since RSA has also been successful in the secure data, it has shown enormous performance problems especially when applied to a two-step encryption and steganography process. This paper, thus, encourages the adoption of Elliptic Curve Cryptography (ECC) as a more efficient and superior replacement for RSA algorithm.

1.1 Inspiration

A method that finds use in designing a multilevel security model whereby the main aim is to ensure no unauthorized access to any classified information is through the combination of audio steganography and ECC. The concept that ECC can be more efficient and implementable than RSA is self-explanatory from the previous study. It is because of this ECC can yield similar security with lower key lengths and processing rates. Also, this shift from RSA to ECC in this study would lead to enhanced data protection from its inconvenience issue with RSA.

Further the research complies with cybersecurity ethics since the personal data of individuals can be saved from destructive attacks. Not only that these ECC and audio steganography-based enhancements for encryption also reduce the risks on steganalysis. Through detailed discussion on ECC working principles, including process of key pair generation, process of encryption and decryption and techniques for embedding audio steganography, the research attempts to provide an efficient answer for modern cybersecurity issues.

1.2 Research Question

How does Elliptic Curve Cryptography with audio steganography offer better security efficiency with secret texts compared to the traditional RSA algorithm with audio steganography?

Further, the paper tries to determine the measurable outcomes in terms of processing speed, computational, and overall security resilience.

1.3 Proposed Solution

To address this research question, the study proposes a systematic approach:

1. **Elliptic Curve Cryptography (ECC) Algorithm Implementation:** One needs to have the basic understanding of the elliptic curve parameters along with the encryption/decryption approach and generation of key pairs in order to start using the Elliptic Curve Cryptography (ECC). Because of the widespread compatibility and efficiency in implementing these approaches, Python is a good option. You can choose ECDH for key exchange that is secure or use ECDSA for authentication, depending on the needs of your system and how to balance security and performance.
2. **Integration of Audio Steganography:** The second part integrates the encrypted data seamlessly into audio files. This will be an area that will cater to more advanced techniques for embedding data into audio while maintaining the quality of the audio itself. The goal is to design algorithms for secure data embedding with no perceivable audible distortion. Strategies to improve the capability of audio files to mask the embedded data should also be studied toward making them more robust to steganalysis methods.
3. **Testing and Evaluation:** A full testing phase is integrated into this third step to assess the performance of the implemented system. In this research, the efficiency of the system will be tested for rapid encryption and decryption of data, data hiding, and prevention against steganalysis attacks. It will also research the influence of different ECC parameters on security system performance and computing requirements in order to achieve the best configuration.

The process of implementing the proposed solution is going to bring out the benefits of integrating ECC with audio steganography compared to conventional RSA-based methods. This project aims at practical insights with maximization of security measures while improving efficiencies and resource consumption right at every stage of data transmission and storage.

2 Related Work

2.1 Rivest, Shamir, Adleman (RSA)

In the paper "Combining RSA and Audio Steganography on Personal Computers for Enhancing Security," a double-layer security method is proposed by embedding RSA-encrypted data in audio files using the Least Significant Bit (LSB) approach and permitting up to three LSBs to enhance capacity while keeping the sound quality pretty good. The paper emphasized the advantages of public-key encryption in RSA compared to AES and further showed that 3-LSBs could be utilized to increase data payload without any noticeable damage to the audio. The research, while making effective use of LSB steganography, does not include detailed evaluation or deep investigation into the interaction between cryptography and steganography as our work does. Our approach allows for a richer performance analysis, illustrating an improved understanding of trade-offs and achieving a higher balance between security and audio characteristics (Al-Juaid and Gutub, 2019).

The paper "Audio Steganography using RSA Algorithm" presents the method of embedding encrypted data inside audio files through RSA cryptography and LSB replacement. Although providing strong security, the method would be less efficient than the implementation proposed because of the use of ECC, which in turn gives better performance compared to the lesser efficiency methods based on RSA, where the key sizes are usually much larger. Besides, this ECC-based RSA replacement inherently has the promise of relatively higher power efficiency, superior performance within limited resources, and a better, state-of-the-art, more effective cryptographic solution to be used in resource-limited contexts (Harshal Chhadwa, Glynes D’souza, Swaradi Godane, Pooja Sharma, 2018).

The paper "Security Enhancement in Audio Steganography by RSA Algorithm" focuses on how to secure text by using the text along with RSA; one way is using RSA for the encryption of text and the LSB method to hide inside audio files. This double-layer security has its objective, such that not only the content is secure but also the presence of it; since SNR is strong, audio distortion is least. The above study proves RSA provides security; it gets compared to ECC, which proposed work has quicker processing and smaller key size with equivalent security, thereby more suitable for environments with limitations in resources. Use of ECC in the proposed method therefore gives a recent and efficient use for high security purposes in audio steganography.(Jawed and Das, 2015)

The work "A Survey of Cryptographic Algorithms for Cloud Computing" appraises the traditional techniques of DES, AES, RSA, and Diffie-Hellman for cloud data security. Although it gives great emphasis to encryption, the double-layered security approach implemented in the proposed work combining ECC and audio steganography is missing. My approach provides key size efficiency and faster processing, hence a more advanced solution to security in the cloud.(Jhuria, Singh and Nigoti, 2013)

The efficiency of using RSA and DES cryptographic algorithms in combination with audio steganography is compared in the research paper titled "Performance Analysis of RSA Algorithm and DES Algorithm with Audio Steganography" by A. Gambhir and R. Arya. This research therefore clearly offers a demonstration of how security in data, mainly for sensitive industries like banking, can be enhanced by combining the many approaches. It has been proved that the two strategies provide multi-layered security, since data is being effectively hidden without any modification on the audio files' properties. This proposed work also uses ECC cryptography for key size compact and fast processing in comparison to RSA and integrates cryptography with steganography. The divergence lies here, where the proposed work goes more sophisticatedly and effectively than the conventional cryptographic techniques on which emphasis is laid throughout the paper. (Gambhir, Ph.D. and Arya, 2019).

The paper titled as “Integrating RSA Cryptography & Audio Steganography” is the integration of RSA encryption and LSB steganography where the messages are encrypted and embedded in the audio files to make the messages both safe and invisible. MATLAB results reveal that the data is very well hidden such that it cannot be seen by anyone, my method recommends adopting techniques like Elliptic Curve Cryptography (ECC) and enhancing LSB steganography to increase security and steganalysis resilience. (Gambhir and Khara, 2016).

2.2 Rivest, Shamir, Adleman (RSA) vs Elliptic Curve Cryptography (ECC)

The paper "A Comparative Analysis" reveals that ECC is more effective than RSA for devices with memory constraints because ECC employs smaller keys. In general RSA can take lesser time to encrypt, but ECC is much faster in decryption and overall computation. The proposed approach should benefit from ECC’s strength in the context of offering security in low-resource environment and also consider the integration of ECC and RSA for secure systems. While the comparison is quite instructive, it would be useful to analyze real-life issues of implementing such approaches and real-world performance indicators. . (Mahto and YADAV, 2017)

Described in the paper of Agilandeswari and Dhanapandian “An Efficient Elliptic Curve Cryptography in Audio Steganography using LSB and MSB” is the method of using LSB and MSB techniques with the integration of ECC for the safe steganography of the audio information. According to the authors, it is recommended to use ECC so that the level of protection against threats can be made extremely high, for this, samples of audio are selected randomly, and secret messages are inserted into both LSB and MSB. New codes offer better technique that involves applying double layer encryption mechanisms using both cryptographic and steganographic methods, and making it harder for the attacker as well as ensuring the authenticity of the data, even though their selection of ECC for the random selection of points enhance the security. The paper titled, ‘A Novel Chaos-Based Cryptography Algorithm and Its Performance Analysis’ will describe another chaotic oscillator-based picture encryption using PRNGs and S-boxes produced from chaotic dynamics. It has the following desirable features of security such as resistance towards several cryptanalytic attacks and high key sensitiveness. The proposed algorithm is much more secure than this work because of the integration of the steganography techniques and elliptic curve cryptography algorithms that would enhance the invisibility and the reliability. Moreover, while analyzing the flow, it is necessary to underline that, in spite of the fact that the work is focused on the picture encryption, the approach described in the paper can be more flexible and potentially may be used for the audio steganography as well. [Author’s name], El-Latif, Tanwar, S. , Tyagi, V. , Kumar, M. , & Privac, Y. (2022).

Such algorithms when applied to open hardware and for-profit platforms for IoT devices are examined in the paper with the title “A Performance Analysis of Lightweight Cryptography Algorithm for Data Privacy in IoT Devices”. With the continuously growing area of IoT, the importance of safe connections and data protection is described in the study, which points at the need for using lightweight cryptography. Hence, proposed technique enhances security and invisibility to justify the unwanted change and tampering of speech content and image information as different layers; one being elliptic curve cryptography and the other being steganography. (Kim and Kim, 2018) .

A comparison of the efficiencies and securities of RSA and ECC is as shown in the study titled: ‘Performance Analysis of RSA and Elliptic Curve Cryptography’. From the study, it is revealed that ECC gives comparable security like RSA but with lesser keys, making it suitable for areas of constrained resources, such as Internet of Things gadgets. From the above study, it is clearly seen that, ECC is faster than RSA most of the time, especially when it comes to the time taken to encrypt as well as decrypt. It also performs key generation for today’s modern digital signatures more efficiently. Thus, the proposed work incorporates the integration between ECC and steganography to improve security and privacy for protecting data especially in cases of multimedia. According to Dindayal Mahto and Dilip Kumar Yadav, (2017).

The overview of ECC is described in Moncef G. Amara & Amar Siad’s work: “Elliptic Curve Cryptography and Its Applications” but focus on the issues of security that it overcame common approaches such as RSA more efficiently. It actually defines what ECC is, what scalar multiplication is, and a bit regarding how both are used in key agreement (ECDH) and signatures (ECDSA). ECC has several advantages which are covered in the given study: it is able to provide outstanding data security being based on decreased lengths of keys, which is useful in contexts where certain limitations are present, such as smart cards or mobile phones. Secondly, it contrasts ECC and RSA; and, thirdly, it shows the advantages of ECC in terms of security and computational load of a concrete key’s size. (Amara and Siad, 2011).

2.3 Steganography (Steganography Techniques, Audio, LSB Methods

The paper "Double Layer Security Using Crypto-Stegno Techniques: There are quite a number of studies done, Some of the aspects that are covered under “A Comprehensive Review” include the integration of image steganography and Cryptography with various objectives such as payload and imperceptibility being measured. Although, it points to the possibilities of “crypto-stego” and calls for more development on color image steganography, it does not show actual application. However, the innovative approach is more efficient one which has gone through numerous tests for various image formats and different levels of encryption, with focus on effectiveness and security of the certain data exchange. . Altmetric revealed that the article was cited in 9 other sources, within a week of its publication where the sources are (Jan et al. , 2021).

In the publication dedicated to a novel steganography approach for audio files by Abdulrazzaq et al, the LSB methodology has undergone some modification to embed encrypted and compressed images into audio files. Before the actual embedding is initiated, the GMPR algorithm pre-processes the images they plan to store, by compressing them in a way that will allow for more capacity and better audio quality. Despite the fact that the DCT technique has been used effectively and adequately preserves picture quality based on PSNR and SNR scores, the paper is mainly devoted to the picture-to-audio steganography. This is done by combining steganography with cryptography, which gives a better security when compared to this method, use of hybrid methods, which offer more of steganalysis resistance and versatility in the sense that they offer a wider appeilation. Whereas the sources which are relatively recent include: Abdulrazzaq Siddeq and Rodrigues, 2020

The paper titled "A Survey on Digital Audio Steganography Techniques" describes digital audio steganography practices to integrate data into audio, on robustness, carrying capacity and invisibility: LSB coding, echo hiding, and transform domain. Although, it offers recommendations in the area of security and capacity, it concerns mostly low security steganography without applying the encryption layer that may pose a challenge to security. However, what makes the method offered in this paper different from just using the ECC is that steganography is embodied with the ECC, thus proving a more holistic security solution. According to Mishra et al , (2018)

The paper Audio Steganography by Rohit Tanwar and Monika Bisla discusses several methods of how secret messages can be hidden in audio signals and at the same time be robust, efficient, and completely imperceivable by the human ear. Some of the techniques include; phase coding which involves coding the signal at a faster rate than the current transmission speed, spread spectrum that has potential to increase the data speed by achieving a higher level of frequency bandwidth spreading, echo concealing which conceals the signal after repeating it in order to make it difficult for other parties to access it, least significant bit (LSB) that hide messages within the lower bits of data transmission as well as tone insertion that Each strategy, of course, has its strength and weakness. For instance, while echo hiding provides confidentiality at the expense of low secrecy, LSB has very high capacity but poor resistance. The authors aptly point out that audio steganography is much more challenging than image steganography due to the HAS' heightened sensitivity (Tanwar and Bisla, 2014).

2.4 Performance metrics for measuring the efficiency of algorithms

The study aimed on analyzing the capability of AES-192 and RSA-512 algorithms that preside in digital signatures is entitled "Comparative performance of digital signature security using cryptography AES 192 BIT and RSA 512 BIT Algorithm Model". The analysis looks at the steadiness, security in general, as well as speed of codes and decryption of various algorithms. AES 192 is slower than RSA 512; RSA 512 is faster but less secure for huge data but this cipher has better security due to some numbers of rounds. (Dhiyaulhaq and Usman, 2021)

In the publication "A Comparative Study on the Performance and the Security of RSA and ECC Algorithm", RSA is again pointed out that, although secure, has lower speed as compared to ECC due to more resource requirements. However, ECC offers the same degree of protection with shorter keys, which means that it requires less time to process and consumes fewer resources than RSA; thus, it is suitable for mobile and constrained devices. New code extends from this by combining cryptography with steganography in that it provides an extra layer of security and data hide that cannot be viewed by anyone other than the sender and the intended recipient.

3 Research Methodology

3.1 Graphical User Interface Development

Tkinter: The project employs the graphical user interface, also known as GUI, through the utilization of Tkinter.

For example, it is crucial for this interface to exist in order to enable the users to easily interact with the said program. Originally, it has components for selection of files, for sending of messages, as well as for performing cryptographic processes. As an extensive set of controls and indicators, the GUI simplifies the work by presenting the necessities of the user in the form of comprehensible buttons, text inputs, and output messages. This makes it friendlier and enables the users to handle it better than the traditional command line interface even if they have never used the interface.

3.2 File and Process Management

Subprocess Module: This module makes the program this program can now run external scripts and commands. It is useful in compiling different features - key generation, encryption, and decryption, into a single application. Modularization of this kind enforces the ability to run a number of scripts separately and the capability of capturing their outputs and dealing with abnormalities. Thus, for enhancing the general efficiency of the program and making the processes coordinated and less complex, it is necessary to tackle the complicated workflows.

PSUtil Library: PSUtil is employed to monitor system resources primarily the RAM and the CPU. Thanks to this monitoring, availability of quantitative indicators characterizing the fulfillment of the tasks by the system during various operations in real time is ensured. This is particularly relevant when measuring the efficiency and resource usage of the cryptographic methods employed on the application.

3.3 Key Management and Cryptography

As a result, ECC is selected as over RSA because it is more efficient and offers greater security features. Specialists state that ECC is more appropriate for environments that are not rich in resources, such as embedded systems and various mobile devices because ECC offers the comparable level of security with much smaller keys compared to RSA. A smaller key size means that the overhead via computations is relatively small, meaning that the parties can encrypt and decrypt at a faster pace. In strategic applications where fast and prompt response to different queries is useful, this aspect is very helpful. That is why the objective of the project to increase ECC for improving the strength of encryption and efficiency against steganalysis matches the general focus of the project.

Key Derivation Function (KDF): The project employs SHA-256 and HKDF (HMAC-based Extract-and-Expand Key Derivation Function) to produce exclusively shared Sym Key from a shared secret. As for the data integrity and secrecy which have to be kept encrypted, this method must be employed. The KDF further enhances the scenario by deriving a key from the shared secret hence no impact on key compromise if the shared secret is breached. This method ensures the security of the encrypted data structure since integrity of the encryption key is ensured.

Galois/Counter Mode, or AES-GCM: AES-GCM also known as the Galois Counter Mode of AES is an Authenticated Encryption with no padding which provides secrecy as well as integrity of data. This method makes sure that the encrypted data can also be regulated and if it is interfered with, then it would easily be detected thus letting it to be an asset to the project security measures. Combination of ECC with Audio Steganography: To create the two-tier security system, the use of ECC is combined with audio steganography.

A large number of advantages over classical RSA with steganography can be concluded: less computational complexity, higher security per one bit, and less size of keys. It is also important to point out that insertion of steganography and ECC helps to avoid the signal quality decrease and constitutes a safe method of data exchange.

3.4 Data Serialization

PEM (Privacy-Enhanced Mail) Format: The project manages and stores all the cryptographic keys in the PEM format. BASE64 is another standard widespread for encoding keys and certificates and it makes management of the cryptographic content more secure. This format makes it easier to safely swap the keys since it is accurate that it is compatible with almost all the systems and cryptographic libraries.

3.5 Steganography

Least Significant Bit (LSB) Encoding: LSB encoding method helps to insert the secret material into the least significant bits of the samples of an audio file. This technique is used in the project to establish a steganography by converting messages into audio data and then hiding them into the audio track. Due to this, LSB encoding is most preferred because it is simple to implement and impacts the perceived audio quality in a trivial manner.

3.6 Performance Metrics

Performance Monitoring: During the performance of operations related to generating asymmetric keys, encrypting data, embedding/extracting keys, as well as decrypting the encrypted material, the application registers and records diverse numerical parameters, such as time taken, CPU load, and memory usage. These metrics should be constantly monitored in order to determine the efficiency of the steganography and cryptography methods used. Also, analysing the performance data, the project can find the optimization opportunities, compare the costs and benefits of security measures to the impact on the application, and ensure that the required performance indicators are met.

4 Design Specification

The layout of the architectures of the system of audio steganography, RSA, ECC is a sum total of several vital modules, where every part of it is charge towards certain operations that when they are amalgamated offer secure as well as effective data management. Interoperability is not an issue because modularity is highlighted and every part can work independently with other parts to give a unified but safe communication system. The Key Generation module, Encryption module, Steganography Embedding module, Steganography

Extraction, and Decryption module are some of the typical components. A thorough explanation of the design architecture is provided below:A thorough explanation of the design architecture is provided below:

4.1 Key Generation Module

For both ECC and RSA key generation, the cryptographic keys are required in which this module is greatly needed. Accordingly, to the specified security criteria, it controls the process of the generation of both public and private keys. This component is known as the ECC Key Generator that selects the elliptic curve and calculates the required key pairs and the other part known as the RSA Key Generator produces the modulus and key pairs. This eliminates the chances of having insecure system since it produces the public and private keys needed in the encryption and decryption of other mathematical operations.

4.2 Encryption Module

These are the keys which are created and used by the Encryption Module in the process of protecting textual messages. RSA and ECC are its two major components: the encryption unit. Security for data can be achieved through encryption wherein each component encrypts the data with utmost use of the correct cryptographic methods. They ensure that the information is converted to encrypted format and thereby applying the first level of security to the information.

4.3 Steganography Embedding Module

This module stegano-audio codec manages to combine the encrypted message with an audio file using the Least Significant Bit (LSB) method. LSB Embedder, Audio File Handler, and Message Converter are the components of this(System). In order to prepare the encrypted message for the embedding the Message Converter transforms the message into the binary form. The LSB Embedder adds the binary message to the produced file in LSB, and the Audio File Handler manages the process of reading and modifying the audio files. Thus, an unauthorized channel for the exchange of information securely is formed, which is an audio file chock full of secret messages without a significant decrease in sound quality.

4.4 Steganography Extraction Module

This message is expected to be extracted by the organism called the Steganography Extraction Module from the audio file. This module has LSB Extractor, Message Converter and Audio File Handler. Before passing it through the Message Handler, the Audio File Handler indexes the audio file that has the hidden message. Thus, the proposed LSB Extractor is employed to extract the binary message from the LSBs of the audio frames. Finally, through bit processing, the Message Converter reconstructs original input of the sending party from the binary data to go through decryption process.

4.5 Decryption Module

The encrypted communication must be decrypted followed by converting it back to plaintext by the help of the Decryption Module. It consist of ECC and RSA decryption parts. Such parts provide that only intended recipients of the original message can read the message as they use the associate private keys to decode the message. This module concludes secure communication by returning the received message to a readable format while at the same time ensuring the genuineness and confidentiality of the received data.

4.6 Integration and Data Flow

The data flow and system integration are made to ensure that the components of the system would mesh. Firstly, the process involves creation of the key, then, following that is encryption and next is the embedding after which the extraction finally followed by decryption. But at the same time each of them possesses an autonomous mode of functioning and all of them are interconnected in a logical way to create a system of quite safe communication.

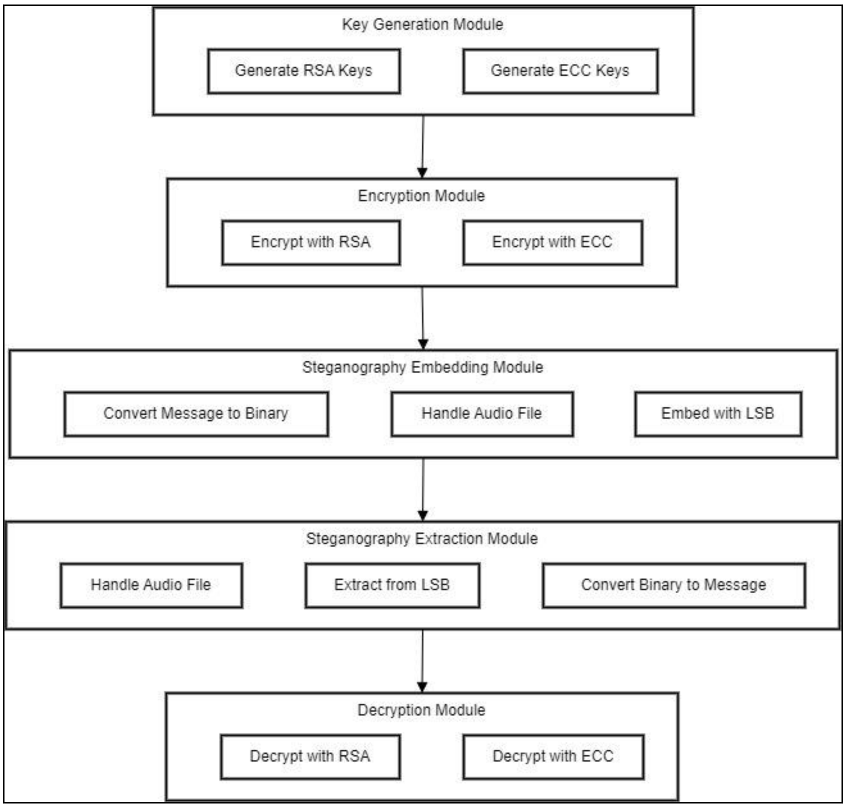


Fig1 : Architecture of the System within App.

5 Implementation

This methodology focuses on how to use ECC or RSA with audio steganography about how to use superior security and efficiency from ECC. The goal of our work is to develop a proprietary and encrypted means of communication in which there are no traces of an unauthorized third party with the help of several encryption algorithms and applying the characteristics of the audio steganography. The technique also underlines that the use of ECC instead of RSA is more effective from perspectives of security and computational load, special in the cases of restricted resources. It entails key creation, message encryption and fusion with audio signals, extraction of the message, decryption among others.

5.1 Key Generation

5.1.1 RSA Key Generation

Another building block of RSA is based on the issue of the factoring of enormous integers computational complexity. The process starts with the generation of two very large primaries, which are usually over 2048 bits to ensure security. This is found by multiplying such primes p and q, which are identified as the two primes used for encryption and decryption of the messages, a modulus p×q is taken as the part of both the public and private keys. The security level is directly proportional to the measure of ***D***n, meaning that large values provide enhanced protection.

In other words, Euler’s totient function expressed as $\phi(n) = (p - 1)(q - 1)$ is crucial in identifying the public exponent e and the private exponent d in cryptography. The value of e is usually selected as 65537, which is the most widely used value since it gives the highest performance and satisfactory protection. The Extended Euclidean Algorithm is employed in the computation of the private exponent ddd so that condition e·d=1(modϕ(n))e \cdot d = 1 \pmod{\phi(n)}e·d=1(modϕ(n)) can be met. While RSA does provide reliable security answer, its need for more computational capacity to accommodate larger keys draws the attention of ECC which provides equally reasonable security answer despite having much shorter keys.

5.1.2 ECC Key Generation

RSA is sometimes considered inferior to Elliptic Curve Cryptography (ECC) because it is more secure with less numbers of key digits attributed to the fact that the solution of ECDLP is also very difficult. ECC is based on the elliptic curve that is defined by the equation y2 = x3 + ax + b mod p where p is a prime number, which makes the equation-increased secured. In this approach, which is the well known secp256k1 curve that is well known to be highly secure. The parameters (a,b,p)(a, b, p)(a,b,p) must satisfy the following equation in order to avoid singularities: This is equivalent to saying that the ivk is nonzero: 4a3 + 27b2 ≠ 0 for some integers a and b where p = q + 4k . 4a3+27b2≠0(modp). For producing secure public keys, a high order nnn base point GGG is required.

5.1.3 Key Pair Generation

In the case ECC, a random number ddd is chosen as the private key from the field 1<d<n1 < d < n1<d\n. Due to the nature of ECC, the scalar multiplication is used to compute the public key Q=dGQ = dGQ=dG which is efficient. While the former is made openly available to the public, the latter has to be kept confidential and cannot be shared at all. The key size of ECC is considerably less than that of RSA this leads to quick execution of the encryption, decryption and generation of keys. In situations, where computational resources are scarce the above approach is important.def generate_ecc_keys():

5.2 Message Encryption

5.2.1 RSA Encryption

In RSA encryption, the public key is displayed as (n,e)(n,e)(n;e). Thus, plaintext messages are secured during transmission. Following understanding of the plaintext, Optimal Asymmetric Encryption Padding (OAEP) which uses SHA-256 is applied. To counter chosen-plaintext attack, OAEP guarantees that even if different plaintext has been encrypted it will always produce a different ciphertext. The membership value given to the message must be securely stored in a binary file. Indeed, it must be noted that while RSA has good encryption features, it is not as fast as ECC: as key sizes increase, their processing time is comparatively longer where data is encrypted frequently.

5.2.2 ECC Encryption

The electronic commerce communication uses symmetric/asymmetric cryptography Integrated Encryption Scheme called ECC to ensure high level security. In this procedure, an ephemeral key pair (k, kG) (k,kG) is derived from a random scalar k k, chosen by the sender. Next, the recipient's public key, Q Q, is used to calculate the shared secret value $S = kQ$ S=kQ, which then is an input-passed to Key Derivation Function (Finally, AES-GCM encryption is employed to efficiently secure the message making sure the integrity and confidentiality of the conveyed data.

Thus, the output is the ciphertext, IV, and the ephemeral public key while guaranteeing that the transmission is secure. ECC is considered preferred to RSA especially when performance is the issue due to the former's comparatively faster encryption rates and shorter keys. Through the used methodology the practical advantages of using ECC with regard to the rescue of the resources and faster processing are shown, which in the contemporary applications required reliable yet efficient encryption.

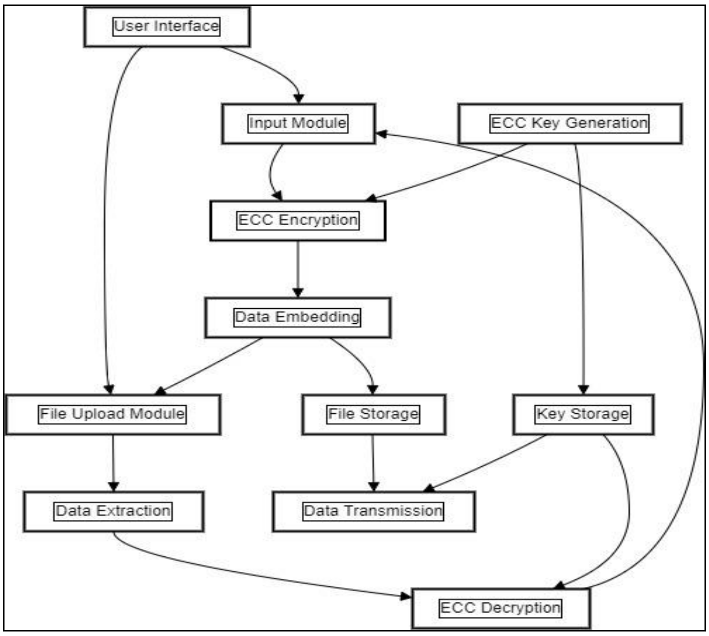


Fig 2: ECC Process Architecture.

5.2.2.1 Steganography of Secret-messages with Audios

1. Message Preparation: A delimiter (b'###') is appended at the end once the message has been read from a file. This delimiter plays the role of indicating to the extractor where the hidden message is basically cut off and ends.

2. Binary Conversion: There are two types of formats, binary formats are produced through the using of the message and the delimiter. As for each byte of the message, an 8-bit binary number is used. The List comprehension and bin() method are used in combination during the conversion to ensure that each byte is represented as an 8:1b sequence.

3. Audio File Handling: A bytearray is made and the frames or the audio samples are sliced out after the audio file is opened in read mode. The amplitude for every audio sample is represented by a byte and as such can easily be manipulated with this array.

4. LSB Embedding: The last few bits of the byte count for the audio frame's representation – these are the bits of the LSBs of the bytes of the audio frame where the binary message bits can be found in. This is done by ORing the message bit (| bit) with the LSB after and-ing the audio byte (frame_bytes[i] &254) with zero. The operation used in the code, (frame_bytes[i] &254) | bit keeps the rest of the bits in particular frames free from interference and maintains the quality of sound.

5. Output Audio File: Next, a new audio file is generated with audio frames which contain the embedded message after the process of editing has been done. As with the previous file, the quality is the same as the original file, only this one contains the hidden message.

5.2.2.2 Extracting Embedded Messages from Audio Files

1. Audio File Reading: According to the audio frames the audio file containing the secret message is opened and the frames are read into a bytearray.

2. LSB Extraction: To reconstruct the hidden message the LSBs of all the bytes in the bytearray are obtained. To do this a bitwise AND of the frame_bytes [i] is performed to obtain the LSB of each byte i. e frame_bytes[i] & 1.

3. Binary to Byte Conversion: The extracted bits are then converted into integer form starting from the bits' binary forms by grouping these into bytes, that is eight bits per byte. It further forms the retrieved message from these integers by converting them into bytes.

4. Message Retrieval: The message content is separated from any supplementary data by dividing the reconstructed message bytes using the delimiter (b'###'). After that, the message is written into a file for further use of the application that implemented the messaging functionality.

5.3 Message Decryption

5.3.1 RSA Decryption

Here, the recipient’s private key will be ddd to assist in RSA decryption and allow the plaintext message to be obtained. The original data is obtained when the encrypted message is decrypted using the private key and OAEP padding removed from it. This technique is very helpful, for the message that is being sent is encrypted and the only person that can read it is the intended receiver with the other end of the private key. While in RSA decryption security is quite strong and reliable, ECC has much less computational complexity than RSA, which makes ECC a better choice when multiple and fast-hand decryption is needed. This approach points out how useful ECC becomes in the modern usage of cryptography since speed is vital in these applications.

5.3.2 ECC Decryption

The recipient's private key (ddd) must be used to extract the shared secret from the ephemeral public key in order to decrypt a message using ECC. A symmetric key is generated that enables AES-GCM to decrypt the ciphertext using a KDF and this shared secret. By verifying that the message originates from a reliable source, decryption ensures that it is real and hasn't been tampered with. ECC decryption is significantly more efficient than RSA decryption, requiring fewer processing resources and producing faster results. This efficiency is especially useful for resource-constrained devices like cell phones and embedded systems that need to decrypt data fast and securely.

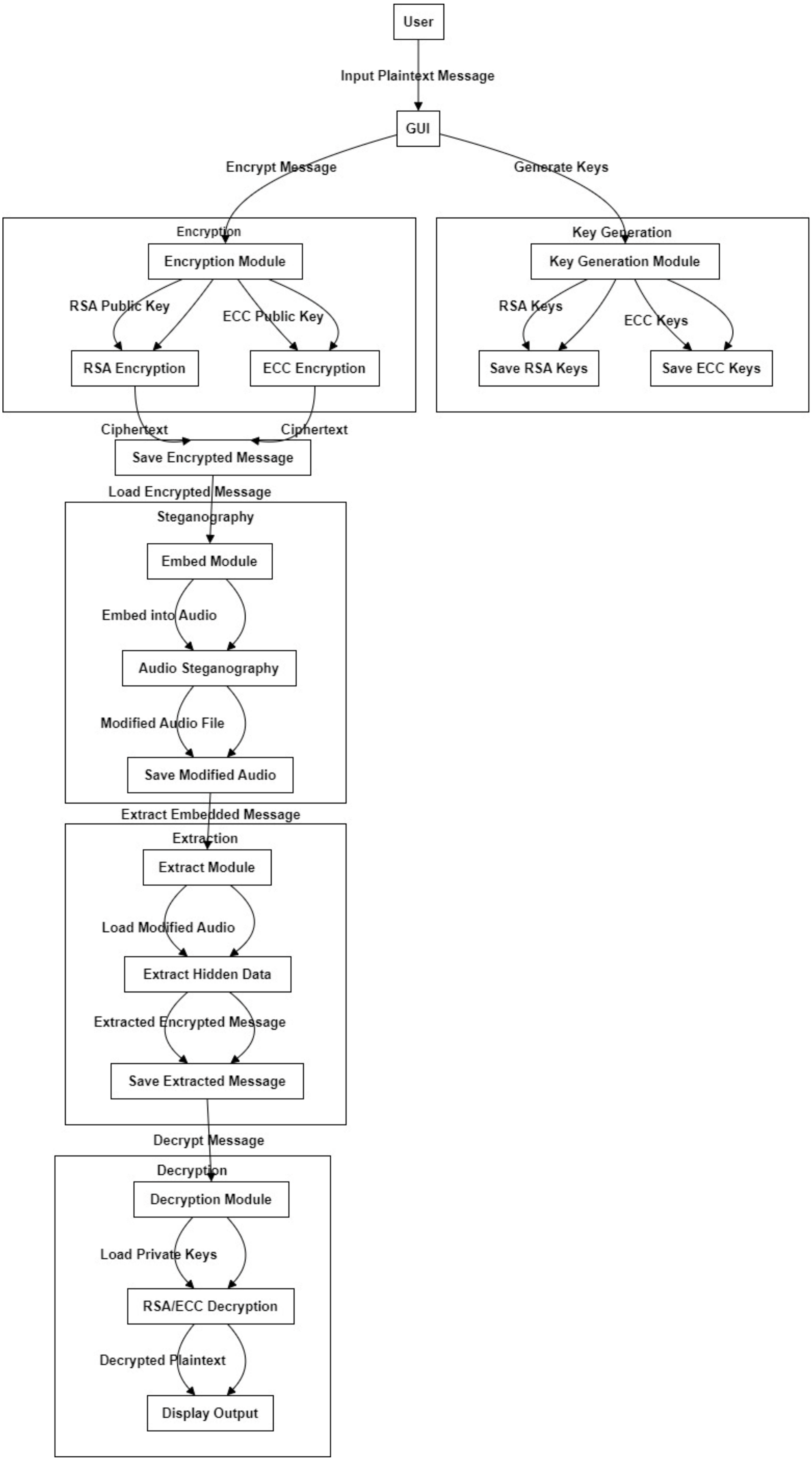


Fig 3:Flowchart Diagram of App.

6 Evaluation

6.1 Performance Metrics Analysis

Since the RSA and ECC, combined with audio steganography, were tested in terms of their performance parameters, the experimentation was carried out consecutively to achieve a comprehensive analysis of the results. The tests revealed that while RSA encryption time remained comparable, varying from 0 Thus, it was possible to observe the direct proportionality between the number of keys and both RSA and AES encryption times. 10 to 0. ;504, ECC was not significantly different from the other models, and the overall processing time was a bit more efficient. Crucially the keys are much smaller with ECC providing the same level of security as much larger RSA keys; and this significantly aided its performance in terms of speed and use of resources. SSD's and RSA's CPU usage, But, fluctuated greatly, indicating responsiveness to CPU concurrency while ECC had a more modest and reliable CPU usage. Concerning memory use, it was noted that RSA fluctuations are significantly larger and sometimes have short bursts, mainly during message embedding compared to ECC memory usage.

6.2 Execution Time Analysis

Comparing the patterns of multiple runs of RSA and ECC encryption operations regarding their execution time provided information. This is the time taken in seconds to carry out RSA encryption for the various sizes and it recorded an average of between 0.10 to 0.14 seconds and this is an ideal timing with little fluctuations ranging from 0.35 seconds. It also emphasizes that RSA algorithm has very stable execution time and does not show significant differences in consecutive runs. On the other hand, ECC encryption times were in average about 0.11 averages at 0.14 seconds, as its performance is stable like RSA. More notably, ECC was generally slightly quicker than RSA except in cases where the divergence was negligible.

Table 1. Execution time (min and max)

Operation	RSA Min Time (s)	RSA Max Time (s)	ECC Min Time (s)	ECC Max Time (s)
Encrypt Message	0.10	0.14	0.10	0.15
Extract Message	1.37	1.59	1.38	1.45
Decrypt Message	0.17	0.18	0.14	0.15
Embed Message	0.10	0.10	0.10	0.11

Table 2. CPU usage

Operation	RSA Min CPU (ms)	RSA Max CPU (ms)	ECC Min CPU (ms)	ECC Max CPU (ms)
Encrypt Message	93.75	218.75	125.00	171.88
Extract Message	1515.62	1812.50	1537.50	1750.00
Decrypt Message	156.25	218.75	125.00	171.88
Embed Message	93.75	109.38	78.12	140.62

Specific to the calculation of CPU utilization in milliseconds, one has to get the total magnitude of the user and system time that the CPU spends on a particular task. By tracking the beginning and the end of a task's execution, the total CPU time is determined and converted into milliseconds to allow better dissection and comprehension of the results. When calculating the CPU usage, the RSA encryption had a lot of fluctuations; its CPU usage values ranged from 93. 75ms - 218. 75ms.

However, what this implies is that RSA encryption is likely to be more sensitive to variations in the system load and processes that run simultaneously with it. On the other hand, and more importantly, ECC encryption showed a much less variable CPU usage ranging between 125 and 132 milliseconds. 00ms to 171. 88ms on average and therefore has a performance that is less variable and therefore more reliable compared to the RSA requirement.

When synthesizing the extent of memory usage for RSA encryption, it was observed that the memory usage was all over the place; ranging from 8192 bytes to 77824 bytes. This wide range indicates that RSA encryption's memory consumption strongly depends on certain run conditions and input data. While as for the ECC encryption the memory distribution was somewhat more distinct, it varied from 4096 to 12288 bytes, however it did not change rapidly as it did in case of RSA.

Table 3. Memory Usage Analysis

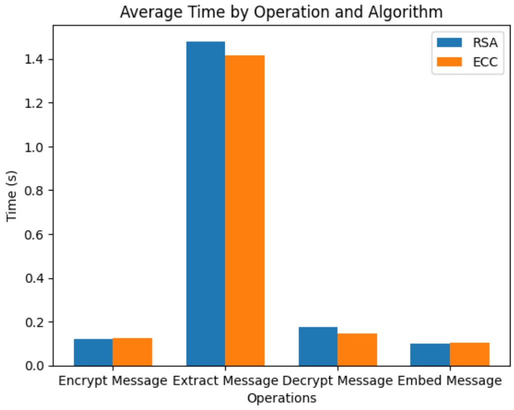
Operation	RSA Min Memory (bytes)	RSA Max Memory (bytes)	ECC Min Memory (bytes)	ECC Max Memory (bytes)
Encrypt Message	16384	77824	4096	12288
Extract Message	4096	8192	8192	12288
Decrypt Message	0	8192	4096	4096
Embed Message	4096	135168	4096	65536

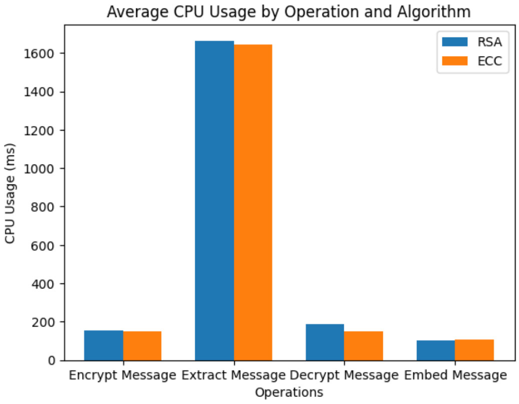
6.3 Statistical Analysis

Execution Time: By using the above factors, it can be deduced that ECC exhibited lesser fluctuation in the time taken to perform the different operations while the mean time taken to perform each operation was slightly lower in the case of ECC than in RSA.

CPU Usage: Theory 3 test also showed that ECC had a less variability in the mean CPU usage with comparison to RCA, therefore it is more efficient and more suitable in utilizing the CPU.

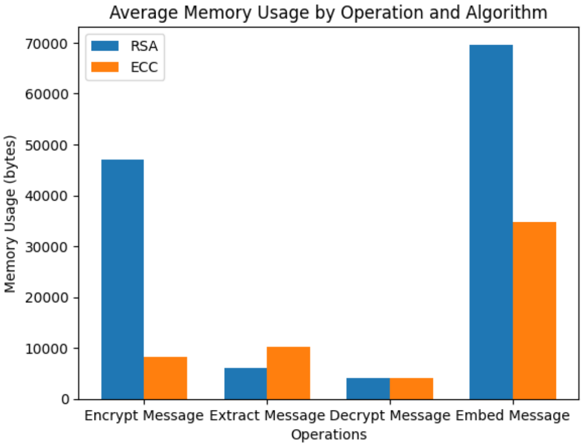
Memory use: In terms of Memory usage ECC was memory efficient and less volatile and therefore could be preferred in setting that have limited memory. Similar to the standard deviation, the mean value of memory usage for ECC was clearly lower compared to that of RSA. To aid in the interpretation of these findings, graphs and charts were created to visually compare the performance of RSA and ECC across metrics:To aid in the interpretation of these findings, graphs and charts were created to visually compare the performance of RSA and ECC across metrics:





Execution Time Comparison Graph

CPU Usage Comparison Graph



Memory Usage Comparison Graph

Academic Perspective: The study is highly beneficial in terms of understanding the performance of RSA and ECC algorithms in terms of practice to meet the clients' needs and gain knowledge of the employed cryptographic methods. The real-world findings of this study provide concrete evidence to back the theoretical benefits of ECC against RSA, availability, expansion, and sleekness, making it possible to link theory with more actual research studies.

Practitioner Perspective: The findings are imperative to practitioners to enhance the selection of the appropriate cryptographic algorithm by giving crucial information to the practitioners who apply the technique. Due to the low and steadily increasing resource demands of ECC compared to the other systems, it may be utilized in environments where resource effectiveness and performance stability are paramount. Hence, by implementing ECC, the practitioners would be able to record improved performance in systems that require the least CPU and memory consumption.

6.4 Discussion

In an attempt of providing further improvement in the cryptographic system than the current widely used RSA techniques, this work comprehensively studies ECC encoder coupled with the Steg analogy of audio signals. The investigations focused on the application of multiple layers LSB and to approve the idea of concealing data in audio data and secondly on ECC algorithms in Python. Measurable factors observed were the time taken to encrypt and decrypt the data, computational and storage behaviours. The implementation procedure was systematic, which involved key generation using both the curves and the primes, encrypting normal text data, hiding the encrypted data into the audio wave, and then retrieving and decrypting the normal data. The decision to use ECC was more informed by the small key sizes it boasts and optimally fewer computational steps than its counterparts that it was believed would help improve overall system efficiency and or security.

These results proved to be useful, as they supported the hypothesis that ECC achieved a higher performance than RSA in several aspects, including the consumption of resources. Hence, we can state that ECC demonstrated more deterministic patterns of CPU and memory performance in contrast to HEC and is capable of providing higher resource predictability required in the context of the corresponding domains such as mobile devices and IoT. However, the net increases in speed as a result of all of those enhancements in the IT/IS were relatively small. Although ECC’s small key sizes often meant longer decryption intervals, which were slightly faster than the cited amounts due to differences in decryption algorithms, the results showed that the performance gain might not be enough to warrant replacing RSA in all cases. An analysis of the experiment setting reveals that though the overall approach of the work demonstrated benefits in adopting ECC over RSA, certain aspects of the methods are susceptible to enhancement. At the same time, the number of test runs was rather small, which could have resulted in the failure to cover the most significant variance in performance. Often a bigger sample size could provide more information on the algorithms effectiveness. However, some other parameters can also be incorporated in the further studies like energy efficiency and extent to which the given schemes are scalable for mobile and IoT devices.

Escalating the system to the actual working environment might yield more information pertaining to the actual efficacy and ease of the software tool. It could also expose new difficulties or factors for improvement that were not evident during the systematic performance of all checks. In light of the literature reviewed, these findings are in par with the existing theoretical and empirical literature that supports the idea that ECC offers better efficiency and stability compared to RSA especially in the developing countries. However, because of the minor increase in speed for both ECC and RSA, it can be recommended to select the cryptography type based on specific characteristics of the application, including resource availability, security, as well as operational conditions.

7 Conclusion and Future Work

When ECC is combined with audio steganography, as applied in this research study, it is apparent that the innovation is much more efficient in cryptography as compared to a commonplace RSA algorithm. The findings indicate that_rsa encrypting times were varying from 0. 10 to 0. 14 seconds, ECC seemed to be slightly ahead with times ranging from 0. 10 to 0. 15 seconds, specifically in decrypting problems because it takes considerably shorter time than RSA, mostly because its key sizes and computational steps hence the NPR are relatively smaller. One of the biggest differences between the algorithms is the resource consumption. RSA had a highly variable CPU utilization compared to the three other clients – all four's minimum CPU usage did not exceed 93. 75 ms to 218. The average response time was 75 ms, which creates the suggestion of inefficiency under a heavy system load. ECC kept the total CPU usage constant and very little fluctuation was observed between the usage 125. 00 ms and 171. 98 ms, which attributes the smaller speed to its greater utilization of system resources and, therefore, the solution's applicability to mobile devices and IoT. Also, ECC's memory usage was less volatile and evenly distributed between 4096 bytes and 12288 bytes, while RSA's usage experienced frequent and possibly random increases. Thus, these results demonstrate the efficiency and stability of ECC algorithm that allows selecting it as the most suitable for modern cryptographic applications especially when one deals with strict limitations of computational and memory capabilities.

As for further research in line with this study, it can be continued to define the integration of the most recent quantum-resistant cryptographic methods into the given platform that comprises the ECC and audio steganography. However, with the development of quantum computers, cryptography like ECC poses to become insecure; thus, this research will shift its focus to make the system post-quantum secure. The project could potentially strengthen a more future-proof security system by implementing and especially testing quantum-resistant algorithms to work out such a system's feasibility.

References

Abdulrazzaq, S.T., Siddeq, M.M. and Rodrigues, M.A. (2020) 'A Novel Steganography Approach for Audio Files', *SN Computer Science*, 1(2), pp. 1–13. Available at: https://doi.org/10.1007/s42979-020-0080-2.

Agilandeswari, V. (no date) 'An Efficient Elliptic Curve Cryptography in Audio Steganography using LSB and MSB', 3(01).

Al-Juaid, N. and Gutub, A. (2019) 'Combining RSA and audio steganography on personal computers for enhancing security', *SN Applied Sciences*, 1(8), p. 830. Available at: https://doi.org/10.1007/s42452-019-0875-8.

Amara, M. and Siad, A. (2011) 'Elliptic Curve Cryptography and its applications', in *International Workshop on Systems, Signal Processing and their Applications, WOSSPA. International Workshop on Systems, Signal Processing and their Applications, WOSSPA*, pp. 247–250. Available at: https://doi.org/10.1109/WOSSPA.2011.5931464.

Dhiyaulhaq, D.H. and Usman, S.A. (2021) ‘Comparative Performance of Digital Signature Security Using Cryptography AES 192 BIT and RSA 512 BIT Algorithm Model’, *Journal of Advances in Information Systems and Technology*, 2(2), pp. 63–72. Available at: https://doi.org/10.15294/jaist.v2i2.44312.

Dindayal Mahto and Dilip Kumar Yadav (2017) ‘Performance Analysis of RSA and Elliptic Curve Cryptography’.

El-Latif, A.A.A. *et al.* (2022) ‘A Novel Chaos-Based Cryptography Algorithm and Its Performance Analysis’, *Mathematics*, 10(14), p. 2434. Available at: https://doi.org/10.3390/math10142434.

Gambhir, A. and Khara, S. (2016) ‘Integrating RSA cryptography & audio steganography’, in *2016 International Conference on Computing, Communication and Automation (ICCCA). 2016 International Conference on Computing, Communication and Automation (ICCCA)*, pp. 481–484. Available at: https://doi.org/10.1109/CCAA.2016.7813767.

Gambhir, A., Ph.D., K. and Arya, R. (2019) ‘Performance Analysis and Implementation of DES Algorithm and RSA Algorithm with Image and Audio Steganography Techniques: Proceedings of ICCASP 2018’, in *Advances in Intelligent Systems and Computing*, pp. 1021–1028. Available at: https://doi.org/10.1007/978-981-13-1513-8_103.

Harshal Chhadwa, Glynes D’souza, Swaradi Godane, Pooja Sharma (2018) ‘Audio Steganography using RSA Algorithm’, *International Journal of Soft Computing and Engineering (IJSCE)* [Preprint].

Jan, A. *et al.* (2021) ‘Double layer security using crypto-stego techniques: a comprehensive review’, *Health and Technology*, 12(1), pp. 9–31. Available at: https://doi.org/10.1007/s12553-021-00602-1.

Jawed, A. and Das (2015) ‘Security Enhancement in Audio Steganography by RSA Algorithm’, 6, pp. 139–142.

Jhuria, M., Singh, S. and Nigoti, R. (2013) ‘A Survey of Cryptographic Algorithms for Cloud Computing’, *International Journal of Emerging Technologies in Computational and Applied Sciences* [Preprint].

Kim, Y.-S. and Kim, G. (2018) ‘A Performance Analysis of Lightweight Cryptography Algorithm for Data Privacy in IoT Devices’, in *2018 International Conference on Information and Communication Technology Convergence (ICTC). 2018 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 936–938. Available at: https://doi.org/10.1109/ICTC.2018.8539592.

Mahto, D. and YADAV, D. (2017) ‘RSA and ECC: A comparative analysis’, *International Journal of Applied Engineering Research*, 12, pp. 9053–9061.

Mishra, S. *et al.* (2018) ‘Audio Steganography Techniques: A Survey’, in, pp. 581–589. Available at: https://doi.org/10.1007/978-981-10-3773-3_56.

(PDF) A Comparative Study on the Performance and the Security of RSA and ECC Algorithm (no date). Available at:

https://www.researchgate.net/publication/344788441_A_Comparative_Study_on_the_Performance_and_the_Security_of_RSA_and_ECC_Algorithm (Accessed: 9 August 2024).

Tanwar, R. and Bisla, M. (2014) ‘Audio steganography’, in. *ICROIT 2014 - Proceedings of the 2014 International Conference on Reliability, Optimization and Information Technology*, pp. 322–325. Available at: https://doi.org/10.1109/ICROIT.2014.6798347.