Hybrid Algorithm using Rivest-Shamir-Adleman and Elliptic Curve Cryptography for Secure Email Communication

Kwame Assa-Agyei[1], Kayode Owa[2], Tawfik Al-Hadhrami[3], Funminiyi Olajide[4]
School of Science and Technology, Nottingham Trent University, United Kingdom[1, 2, 3]
School of Computer Science and Engineering, University of Westminster, United Kingdom[4]

***Abstract*—Email serves as the primary communication system in our daily lives, and to bolster its security and efficiency, many email systems employ Public Key Infrastructure (PKI). However, the convenience of email also introduces numerous security vulnerabilities, including unauthorized access, eavesdropping, identity spoofing, interception, and data tampering. This study is primarily focused on examining how two encryption techniques, RSA and ECC, affect the efficiency of secure email systems. Furthermore, the research seeks to introduce a hybrid cryptography algorithm that utilizes both RSA and ECC to ensure security and confidentiality in the context of secure email communication. The research evaluates various performance metrics, including key exchange time, encryption and decryption durations, signature generation, and verification times, to understand how these encryption methods affect the efficiency and efficacy of secure email communication. The experimental findings highlight the advantages of ECC in terms of Key Exchange Time, making it a compelling choice for establishing secure email communication channels. While RSA demonstrates a slight advantage in encryption, decryption, and signature generation for smaller files, ECC's efficiency becomes apparent as file sizes increase, positioning it as a favorable option for handling larger attachments in secure emails. Through the comparison of experiments, it is also concluded that the hybrid encryption algorithm optimizes the key exchange times, encryption efficiency, signature generation and verification times.**

Keywords—RSA; ECC; Advanced Encryption Standard; encryption; decryption; signature generation; verification; key exchange time; hybrid encryption

I. Introduction

In today's interconnected world, email has become an indispensable tool for communication. It facilitates the exchange of information, ideas, and documents across vast distances, enabling individuals and organizations to collaborate and communicate efficiently [1]. However, the convenience of email also brings with it a host of security concerns, as emails can easily fall prey to unauthorized access, eavesdropping, identity spoofing, interception, or data tampering [2] [3]. This is where cryptography plays a pivotal role in ensuring the confidentiality, integrity, and authenticity of email communication. Email encryption is the cornerstone of secure email communication. It employs complex algorithms to transform the content of an email into an unreadable format, known as ciphertext, which can only be deciphered by the intended recipient possessing the decryption key [4]. This cryptographic process ensures the following [5] [6]:

1) *Confidentiality:* Encrypted emails are incomprehensible to anyone without the decryption key, thwarting unauthorized access and eavesdropping.

2) *Integrity:* The recipient can promptly detect any alterations made to the encrypted email, ensuring the message's integrity remains intact during transmission.

3) *Authentication:* Combining encryption with digital signatures verifies the sender's identity, ensuring the legitimacy of the email for the recipient.

Email security protocols such as Secure Sockets Layer (SSL) and Transport Layer Security (TLS) commonly utilize encryption algorithms like RSA, AES, and elliptic curve cryptography, while public-key infrastructure (PKI) systems and digital certificates play vital roles in verifying the authenticity of email senders [7].

This research project's primary aim is to explore the correlation between the performance of secure email communication systems and the encryption methods employed. The study examines how incorporating both RSA and ECC encryption techniques influences the efficiency of secure email systems. The study seeks to discern any connections between the choice of cryptographic algorithms, RSA and ECC, and the overall performance of secure email systems. Furthermore, the study suggests a hybrid cryptography algorithm that utilizes both RSA and ECC to enhance security and preserve confidentiality in the context of secure email communication. The research will assess various performance aspects, including key exchange time, encryption and decryption times, signature generation and verification times, to ascertain how these encryption methods influence the efficiency and efficacy of secure email communication. Through an analysis, the study's aim to identify any potential relationships or dependencies between the selection of encryption methodologies and the outcomes in terms of secure email system performance.

A. Limitation of Existing Models

While existing encryption models, such as those based solely on RSA or ECC, have undoubtedly contributed to the advancement of secure communication protocols, they nonetheless exhibit limitations that may not be well suited to address the multifaceted challenges inherent in secure email communication. A significant drawback concerns the scalability of conventional encryption methods, especially when it comes to email communication. As the size and complexity of email attachments continue to grow, conventional encryption algorithms, like RSA, may struggle to maintain optimal performance, leading to increased computational overhead and potential delays in key exchange, encryption, and decryption processes [8].

B. Rationale for the Proposed Model

The selection of the hybrid cryptography algorithm, combining RSA and ECC, for secure email communication stems from a thorough consideration of various factors. Firstly, while RSA has been a stalwart in encryption for decades, its efficiency can be compromised, particularly when dealing with larger files or computational constraints. RSA's computational complexity grows with the size of the data, affecting encryption and decryption times [8].

On the other hand, ECC has emerged as a promising alternative due to its ability to provide equivalent security with shorter key lengths, thus reducing computational overhead and enhancing performance, especially in resource-constrained environments. Compared to RSA, ECC offers the same level of security with significantly smaller key sizes, making it more efficient for key exchange and digital signatures [9].

However, both RSA and ECC have their respective strengths and weaknesses. RSA excels in signature generation and verification for smaller files, while ECC demonstrates superior efficiency in key exchange, particularly noticeable as file sizes increase [10]. Recognizing these complementary attributes, the hybrid approach seeks to leverage the strengths of each encryption technique to mitigate their individual limitations. Combining RSA and ECC allows for a more balanced security posture by utilizing RSA for digital signatures and ECC for key exchange.

Moreover, by combining RSA and ECC within a hybrid model, we aim to achieve a balance between security and efficiency in secure email communication. This approach allows us to capitalize on RSA's robustness in certain aspects while harnessing ECC's efficiency gains in others, ultimately offering a more versatile and adaptable solution for addressing the diverse security challenges inherent in email communication.

The aim of the study is to enhance the performance of secure email systems by leveraging the distinct advantages of both RSA and ECC, while minimizing their individual limitations. This rationale underscores the relevance and appropriateness of the proposed model in addressing the inherent complexities of secure email communication in today's digital landscape.

Hence, the current study makes the following key contributions.

1) To perform an extensive analysis of the performance of selected algorithms, namely: RSA and ECC for secured email communication.

2) To perform an extensive evaluation of the key exchange time, signature generation and verification times between RSA and ECC techniques.

3) To present a hybrid cryptography algorithm that employs both RSA and ECC to ensure confidentiality and enhance security for secure email communication.

The rest of the paper is organized as follows: Section II presents the related work. The experimental setup is presented in Section III. Sections IV presents the hybrid techniques employing RSA and ECC. Section V and Section VI present the performance analysis of solo ECC, RSA and the proposed hybrid algorithm and discussion of the research. Finally, the conclusion is drawn in Section VII.

II. Related Work

In this study, a secure system for key agreement and session authentication for Internet of Things (IoT) devices was conceptualized, developed, and subjected to testing. The simulation results showed that the method was resilient to different assaults. The results of the performance evaluation also showed that, when compared to DSA and RSA, the time complexity was lowest for the ECC used in this case. The developed protocol also had the lowest computational overhead, the quickest turnaround times, and the greatest stability with the least amount of communication overhead [11]. In study [12], researchers conducted an analysis of distinct cryptographic algorithms, evaluating aspects like key size, message size, and execution time. With the proliferation of diverse encryption techniques, facilitating swift and dependable communication among IoT devices has become a complex task, one that must be accomplished without causing interruptions. Determining the most suitable, compatible, and advantageous encryption method for communication has proven to be quite intricate. Through their examination, the authors reached the consensus that among various options, Schnorr, RSA, Elliptic Curve Cryptography, and ElGamal emerge as the superior choices. Kaur and Aggarwal [13] undertook an extensive examination of cryptographic methods including RSA, Blowfish, Diffie-Hellman, ECC, and others. Among these techniques, ECC has demonstrated itself as the encryption method that excels in both security and efficiency. In reference to [14], the researchers delved into an analysis of diverse encryption methods encompassing both symmetric and asymmetric cryptographic techniques. These methods included Rivest Shamir and Adleman, Diffie-Hellman, Digital Signature Algorithm, as well as Elliptic Curve Cryptography (ECC). The study aimed to explore their practical implementation and ascertain the most effective cryptographic techniques capable of ensuring comprehensive data confidentiality during transmission. Each cryptographic algorithm i found to possess distinct strengths, features, advantages, complexities, efficiency levels, and limitations. Among these factors, the examination revealed that digital signatures offer robust confidentiality and non-repudiation, thereby serving as a means to safeguard data integrity, availability, and confidentiality. A thorough examination is carried out to investigate the ECC, RSA, and Diffie-Hellman Algorithms within the realm of Network Security. The study addresses the challenges associated with sharing and transferring private keys among systems. When assessing the efficacy of ECC, RSA, and Diffie-Hellman algorithms, ECC stands out as the favoured option due to its capacity to deliver almost comparable security levels while employing fewer bits than both RSA and Diffie-Hellman. The researchers also delved into the realm of elliptic curve cryptography, exploring its significant applications within the market. This exploration involved examining its prevalence in technologies like Bitcoin, Secure Shell, and Transport Layer Security. It is evident from the literature that elliptic curve cryptography stands out as a promising approach, capable of offering exceptional security advantages over comparable algorithms. Moreover, its cost-effectiveness positions it as a valuable contender for cryptographic applications [15]. In study [16], the researchers conducted an analysis of the performance characteristics of conventional public-key cryptographic systems, namely RSA, DSA, and DH, in comparison to ECC. The investigations highlighted that the traditional public-key methods encounter performance-related challenges. The study proposed that general-purpose CPUs could effectively incorporate hardware acceleration to enhance public-key algorithm processing. The performance assessment indicated that ECC exhibited superior performance compared to RSA. Specifically, for ECC with GF(p) and GF(2m), the researchers noted a speedup of 2.4 times and 4.9 times, respectively, relative to RSA at the current security levels. Moreover, for subsequent security levels, the corresponding speedups were even more substantial—7.8 times and 15.0 times, respectively. Ponomarev et al., (2010) [17] conducted an investigation into the computational demands imposed by handling the control plane of the Host Identity Protocol (HIP) using Rivest-Shamir-Adleman (RSA) encryption in comparison to Elliptic Curve Cryptography (ECC) techniques. This study focused on measuring the processing resources consumed by the cryptographic procedures of the Host Identity Protocol Base Exchange. The findings highlighted that the cryptographic operations involved in the Host Identity Protocol Base Exchange consumed substantial processing resources, and this aspect is quantified in the study. In terms of specific results, the study indicated that, by employing ECC for the Diffie-Hellman exchange, a server could manage new connections ranging from two to three times more efficiently. Moreover, the study highlighted that employing ECC in cryptographic operations significantly enhanced HIP performance for lightweight mobile clients like the Nokia N810 Internet Tablet. This improvement was manifested in a 75-85% reduction in total Base Exchange (BEX) time, emphasizing the faster cryptographic operations enabled by ECC. In this investigation [18], a comparison was conducted between the elliptic curve cryptography (ECC) algorithm utilizing a 160-bit key size and the Rivest-Shamir-Adleman (RSA) technique employing a 1024-bit key size. The results demonstrated that ECC can offer comparable security levels with smaller key sizes when contrasted with more traditional cryptographic systems like RSA. Consequently, the adoption of ECC is strongly recommended to enhance security and efficiency without a proportional increase in computational demands. The research indicated that ECC maintains a lower cost ratio. Moreover, continuous enhancements are necessary for ECC itself to optimize the performance of newly developed chips. In a study similar to this, the authors referenced in study [5] investigated the encryption and decryption times of various approaches using data packets of different sizes. The comparisons indicated that ECC leads to a significant reduction in transmission expenses. The results underscored that ECC outperforms other asymmetric algorithms in terms of efficiency. This study evaluated the impacts of different ECC curves and RSA key sizes using IoT nodes with limited resources, and it compared the performance of ECDSA and RSA TLS cipher suites. The results indicated that, although ECDSA consistently outperformed RSA in all test runs, practical scenario testing is necessary to determine the suitable security configuration for a given hardware platform. Situations may arise where more secure options, due to software implementations and optimizations, exhibit superior energy efficiency and data throughput, surpassing theoretically lighter and simpler alternatives. The results, influenced by enhancements in the libraries handling ECC operations, specifically showcased that the secp256r1 curve exhibited superior performance compared to the secp224r1 curve, while maintaining a higher level of security [19]. In the study conducted by Kardi et al. in 2018, a performance evaluation is carried out to compare RSA and Elliptic Curve Cryptography in the context of wireless sensor networks. The findings of the research indicated that the decryption time of RSA becomes impractical with large key sizes. Conversely, even when employing very large key sizes, the encryption and decryption processes of ECC algorithms remain manageable. Additionally, ECC signature signing is generally faster than verification, whereas RSA signature signing is more time-consuming. As a result, the study recommended a transition from RSA to elliptic curve cryptography [20]. In a comparable investigation, the researchers delved into the foundational aspects of elliptic curves, their associated arithmetic operations, and the advantages of adopting elliptic curve cryptography over RSA within public cryptosystems. The outcomes of this research highlighted that ECC signature signing processes are usually swifter than verification procedures, while RSA signature signing tends to be more time-consuming. Moreover, the generation of public keys demands significantly more time with the RSA technique compared to ECCs. These findings in the implementation phase provided a compelling rationale for the researchers to advocate for a transition from RSA to elliptic curve cryptography [21]. This study conducted an empirical performance assessment aimed at comparing and quantifying the performance of two encryption schemes: (1) RSA-based BROSMAP and (2) ECC-based BROSMAP, both on the client side (Android) and server-side (XAMPP). In terms of execution time, ECC outperforms RSA significantly, with ECC being nearly twice as fast as RSA 2048 and four times faster than RSA 3072. The primary factors contributing to ECC's superior performance in BROSMAP are its utilization of smaller key sizes and its exclusive reliance on symmetric cryptography for both encryption and decryption processes. Furthermore, the investigation revealed that RSA-based BROSMAP incurs higher computational costs compared to ECC-based BROSMAP. Specifically, ECC demonstrates a computational efficiency that is 561 times greater. As a result of these findings, the researchers strongly recommend the adoption of ECC-based BROSMAP over RSA-based BROSMAP, especially in systems with limited resources such as IoT devices and agent-based systems that prioritize security. In summary, ECC-based BROSMAP meets all the security requirements of RSA-based BROSMAP while offering greater efficiency and lightweight operation, attributed to its absence of asymmetric encryption, use of reduced key sizes, and utilization of ECC keys in conjunction with symmetric encryption [22]. In their study referenced as [23], the researchers conducted an analysis of the security capabilities of ECC and RSA encryption techniques using three sets of sample input data consisting of 8 bits, 64 bits, and 256 bits, each employing randomly generated keys in accordance with NIST recommendations. Their findings illustrate that ECC surpasses RSA in both operational efficiency and security. Furthermore, their work implies that ECC might be the preferred choice, particularly for devices with limited memory resources such as smartphones and palmtop PCs. In this paper, authors [5] conducted a comprehensive review of key cryptographic algorithms, including ECC, El-Gamal, and RSA, with the goal of facilitating a comparative assessment. Our comparisons clearly indicate a significant reduction in transmission costs when employing ECC. These outcomes underscore the practical advantages of ECC's performance. The survey is undertaken to assess the security aspects of these algorithms, considering their widespread utilization. This research conducted an examination of two frequently employed encryption methods, namely Elliptic Curve Cryptography (ECC) and Rivest-Shamir-Adleman (RSA), with a particular emphasis on their applicability in the context of cloud and fog computing. The investigation involved a comparison of the key size and security capabilities of ECC and RSA algorithms, assessing their suitability for deployment in resource-limited fog computing environments. The findings suggest that ECC is a preferable choice for enhanced security and faster performance, all without imposing undue strain on computing resources. In contrast, RSA, with its established track record of security, remains widely accepted [24] . This paper introduced a novel approach to file encryption, employing a hybrid encryption algorithm that combines AES and RSA. It provides a foundational understanding of the AES and RSA algorithms while conducting a thorough examination of their pros and cons. The encryption techniques of these two algorithms have garnered substantial attention within the scholarly community. Through experimental comparisons, the study concludes that the hybrid encryption algorithm enhances encryption efficiency, bolsters key management, and fortifies data security in the context of file protection [25]. The authors in [26] presented a novel technique that combines features from two different algorithms. The primary goal of this approach is to tackle two significant challenges: managing encryption keys in symmetric encryption algorithms and reducing the substantial power consumption associated with asymmetric encryption algorithms. The research delves into cryptographic algorithms using data gathered from related academic journals and conference papers. The study's outcomes demonstrate that the proposed system has successfully elevated the maximum accuracy requirement, mainly due to its enhanced security level achieved by employing multiple keys for encryption and decryption. In this paper, a novel and secure data sharing scheme is presented, with a key emphasis on upholding data security and integrity within cloud environments. The proposed system primarily relies on the fusion of Elliptic Curve Cryptography (ECC) and the Advanced Encryption Standard (AES) method to ensure robust authentication and data integrity. Experimental results highlight the efficiency of this approach, demonstrating superior performance when compared to existing methods. In this paper, a novel and secure data sharing scheme is presented, with a key emphasis on upholding data security and integrity within cloud environments. The proposed system primarily relies on the fusion of Elliptic Curve Cryptography (ECC) and the Advanced Encryption Standard (AES) method to ensure robust authentication and data integrity. Experimental results highlight the efficiency of this approach, demonstrating superior performance when compared to existing methods [27]. This paper delves into the capabilities of cryptography for ensuring security in distributed storage. This exploration involves a thorough examination of standard cryptography techniques such as AES, ECC, and RSA. However, due to variations in the performance of these methods, the study addresses the challenge of identifying an encryption technique that strikes a balance between efficiency and security. Some encryption methods can deliver high security but are time-consuming for both encryption and decryption. Conversely, other approaches may offer efficient encryption but suffer from vulnerabilities in terms of security [28]. In reference [29], a two-tier cryptographic approach and a model are introduced to enhance data security in cloud computing. This model leverages both symmetric and asymmetric encryption algorithms, specifically AES and ECC, to bolster data security against unauthorized access, thus promoting privacy, data integrity, and expediting cryptographic operations. This advancement serves to enhance user trust in cloud computing while also accelerating the utilization of smaller ECC keys in the encryption process. In their research presented in reference [30], the authors conducted an examination of Elliptic Curve Cryptography (ECC) as a means to enhance data security within cloud environments, drawing a comparison with the Advanced Encryption Standard (AES) with a specific focus on time efficiency. The review encompasses an evaluation of encryption and decryption durations for data stored in cloud-based systems. This study explores the improvement of data protection with an emphasis on time efficiency through the application of ECC and AES. The analysis was based on a sample size of N=6 for both ECC and AES, with a significance level of 80% (determined using the G-power value). It is found that the mean time required for ECC encryption was 0.1683, whereas for AES, it was 0.7517. The calculated significance value for the proposed system was 0.643 ($p>0.05$). The results within the confines of this study clearly indicate that ECC surpasses AES in terms of expeditious data encryption with reduced time consumption. In 2023, Rao and Sujatha introduced a security technique for public cloud systems using Hybrid Elliptic Curve Cryptography (HECC). Their proposed method involves the creation of keys utilizing a lightweight Edwards curve, followed by the utilization of Identity Based Encryption to modify the generated private keys. Additionally, the study implements a key reduction technique to further shorten the keys, thereby accelerating the Advanced Encryption Standard (AES) encryption process. The exchange of public keys is facilitated through the Diffie-Hellman key exchange. To evaluate the effectiveness of their proposed model, the authors employ metrics such as throughput and the time taken for key generation, encryption, and decryption. The results indicate that their model outperforms existing ones in various aspects. The key creation process in their method requires just 0.000025 seconds, with encryption taking 0.00349 seconds. Furthermore, the achieved throughput reaches 693.10 kB/s [31]. In this paper, the authors have introduced a robust and efficient protocol that incorporates security measures using both a blind factor and the Elliptic Curve Cryptography (ECC) scheme. ECC is preferred over RSA due to its ability to provide superior security with smaller key sizes, resulting in reduced computational overhead. This enhanced security per unit of data offers various benefits, including faster processing, lower power consumption, reduced bandwidth usage, improved storage efficiency, and more compact certificates. These advantages prove especially advantageous in situations with limitations in terms of bandwidth, processing capacity, power availability, or storage space. In order to improve computational efficiency and minimize memory storage requirements, the authors have developed a novel Hybrid Public Key Cryptographic algorithm. The results indicate that the incorporation of Dual-RSA and ECC has significantly improved the algorithm's performance, both in terms of computational cost and memory storage demands [32]. This paper introduced a hybrid cryptography algorithm aimed at ensuring confidentiality and enhancing security for internet communications. The research places particular emphasis on minimizing the time required for encryption and decryption to avoid excessive CPU utilization. Experimental findings demonstrate that the proposed solution offers a more efficient means of encrypting messages, with only a marginal difference in the algorithm's runtime. This approach effectively enhances security in the open internet environment [33].

III. Experimental Setting

A local area network (LAN) consisting of a dedicated network server and two client machines were used to carry-out the simulation. Fig. 1 shows the diagram of the simulation setup.

The testing environment, which included standard email clients and server, featured laptops with Intel i7 processors, 16GB RAM, and solid-state drives. The server was equipped with the following specifications - OS: Ubuntu, CPU: Intel Xeon, Cores: 4 core, 2.4 GHz, Architecture: 64-bit, and RAM: 8 GB DDR4 - represent typical end-user systems in corporate or personal contexts. To commence, email server Exim and clients Mozilla Thunderbird are configured to support RSA, ECC and hybrid encryption. Public and private keys are generated for RSA, typically 2048 bits, and ECC, using the widely adopted SECP224R1 curve for each user. Key exchange occurred during the initial email contact, and keys were securely stored. The experiment is conducted five times and the mean values for each metric were recorded. A series of practical tests are conducted, involving the sending of emails of varying sizes, from small text-based messages to larger attachments. Throughout the tests, measurements are taken for encryption and decryption times, key exchange time, signature generation, and verification times.

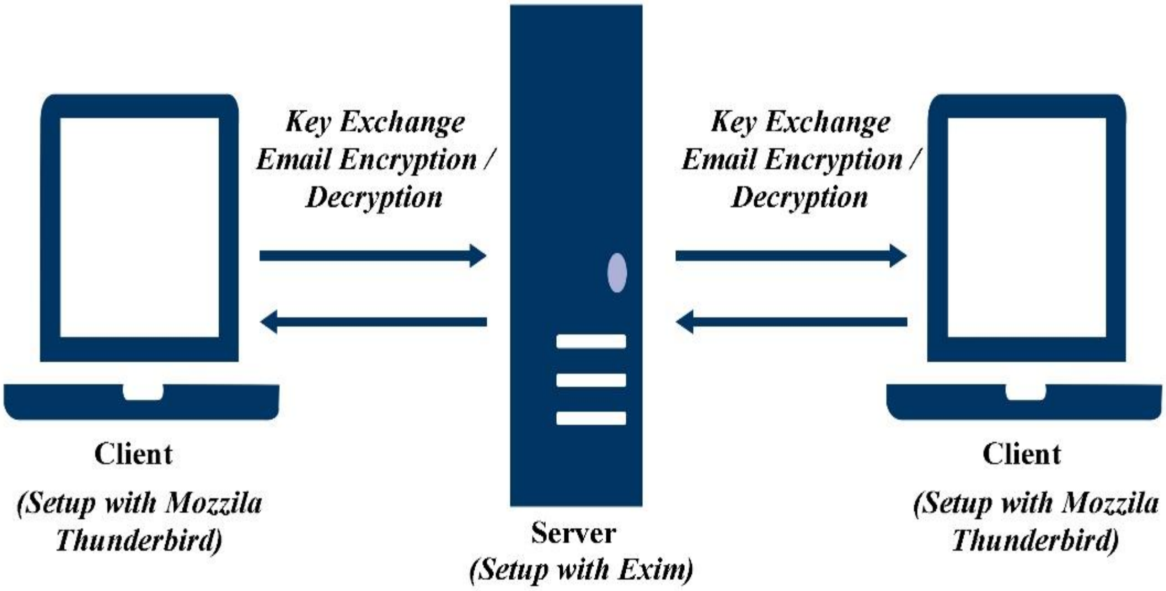
Key Exchange
Email Encryption / Decryption
Key Exchange
Email Encryption / Decryption
Client
(Setup with Mozzila Thunderbird)
Server
(Setup with Exim)
Client
(Setup with Mozzila Thunderbird)

Fig. 1. Simulation setup.

IV. RSA-ECC Hybrid Technique

The proposed hybrid technique merges the robust aspects of both RSA and ECC, creating a cooperative strategy that not only strengthens the security of email communication but also streamlines its efficiency. RSA and ECC stand as widely embraced encryption methods for safeguarding email exchanges. RSA, a traditional approach, provides formidable security but can be computationally demanding, particularly when managing large files. Conversely, ECC excels in terms of encryption and decryption speed, making it an ideal choice for resource-constrained environments. The suggested hybrid method presents an inventive solution to the enduring challenge of striking a balance between security and performance in secure email communication. By harnessing the strengths of RSA and ECC, this hybrid technique provides an adaptable solution that can be tailored to meet specific email communication needs. Fig. 2 provides a visual representation of the proposed fusion of RSA and ECC algorithms. This visual representation provides a clear and intuitive understanding of the sequential data transformations performed within the algorithm. For a more in-depth exploration of the inner workings of this proposed algorithm, Algorithms 1 and 2 offer a comprehensive breakdown of its structure. These algorithmic descriptions present a step-by-step elucidation of the specific operations and procedures involved at each stage of the algorithms.

Algorithm 1: Sender-side Operations

```
Input: ECC key pair, RSA public key, plain text email
Output: Encrypted email, Digital Signature
Step1: Generate sender's ECC Private Key as $S^a$ and Public Key as $S^b$
Step 2: Request recipient's RSA public key from the server as $R^b$
Step 3: Derive shared secret using sender's ECC private key and recipient's RSA public key. $S = D(S^a, R^b)$
Step 4: Generate a symmetric key ($A^s$) for encrypting the email message.
Step 5: Compose the email message.
Step 6: Encrypt the email message using the symmetric key.
    Ciphertext, $C^t = E(PT, A^s)$
Step 7: Encrypt the symmetric key using recipient's RSA public key.
    $C^{As} = E(A^s, R^b)$
Step 8: Generate a digital signature for the email message.
    $Ds = Gs(S^a, PT)$
Step 9: Send the secure email to the recipient
```

In Algorithm 1, the sender initiates secure email transmission by generating ECC Private and Public Keys (S^a and S^b). The recipient's RSA public key (R^b) is acquired, and a shared secret is derived through S^a and R^b. A symmetric key (A^s) is created for encrypting the email message using AES, ensuring confidentiality. The composed email message is encrypted with A^s, yielding ciphertext (C^t). For added security, A^s is encrypted with R^b, resulting in the encrypted symmetric key (C^{As}), safeguarding its transmission. To ensure data integrity and authentication, a digital signature (Ds) is generated using S^a and the plain text email (PT). The secure email, comprising C^t, C^{As}, and Ds, is then transmitted to the recipient. This algorithm thus combines ECC and RSA functionalities to achieve a comprehensive security framework, encompassing symmetric and asymmetric encryption, as well as digital signatures for secure email communication.

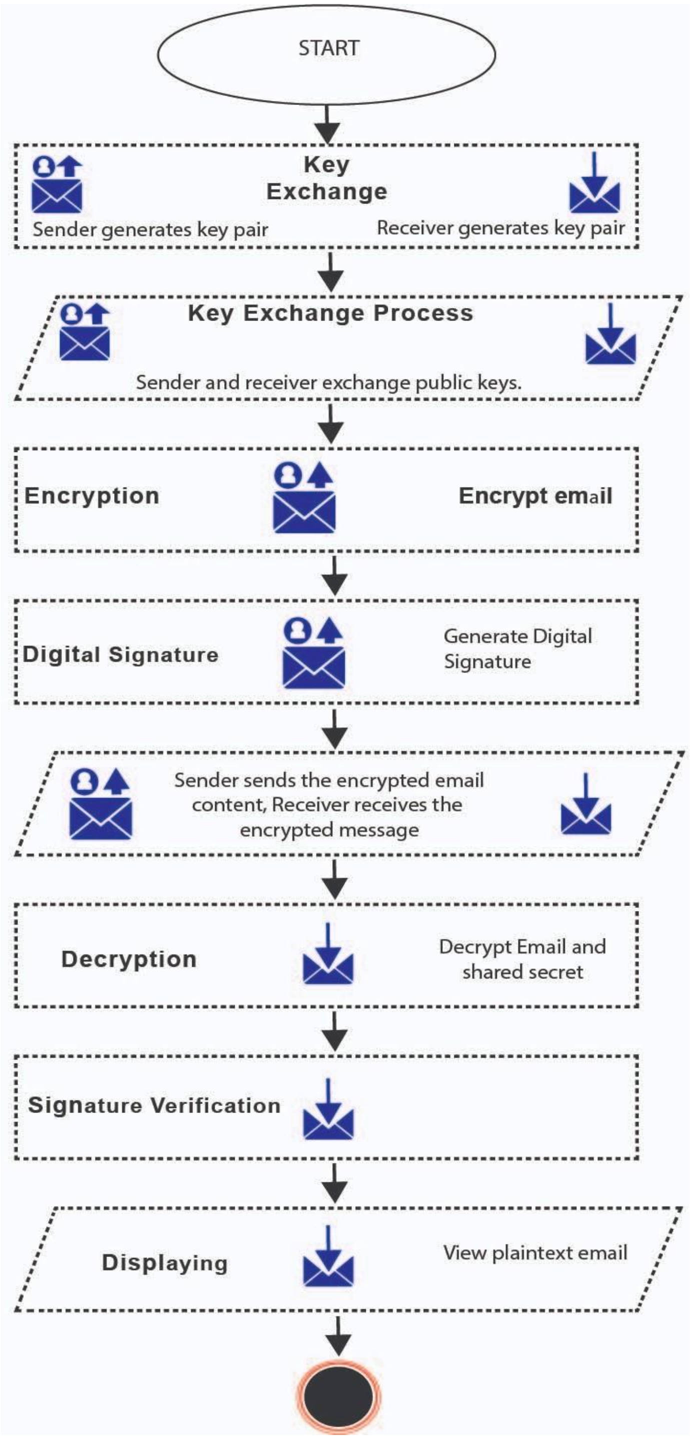
START
Key Exchange
Sender generates key pair
Receiver generates key pair
Key Exchange Process
Sender and receiver exchange public keys.
Encryption
Encrypt email
Digital Signature
Generate Digital Signature
Sender sends the encrypted email content, Receiver receives the encrypted message
Decryption
Decrypt Email and shared secret
Signature Verification
Displaying
View plaintext email

Fig. 2. Proposed model flow graph with hybrid ECC, RSA and AES.

Algorithm 2: Recipient-side Operations

Input: RSA key pair, ECC Key pair, Encrypted Email
Output: RSA Public Key, Decrypted email

```
Step1: Generate recipient's RSA and ECC key pairs.
RSA Private, Public ($R^a$, $R^b$) & ECC Private, Public ($S^a$, $S^b$)
Step 2: Export recipient's RSA public key for the sender
Step 3: Receive the encrypted symmetric key from the sender.
Step 4: Decrypt the symmetric key using recipient's RSA private key.
$A^s = D (C^{As}, R^a)$
Step 5: Receive the encrypted email message.
Step 6: Verify the sender's ECC public key.
Step 7: Receive and verify the digital signature.
Step 8: Decrypt the email message using the symmetric key.
        $PT = D (C^t, A^s)$
Step 9: View the decrypted email
```

In Algorithm 2, the recipient begins by generating RSA and ECC key pairs (R^a, R^b) and (S^a, S^b) respectively. The recipient's RSA public key (R^b) is exported for the sender's use. Upon receiving the sender's secure email, the recipient obtains the encrypted symmetric key (C^{As}) and decrypts it using their RSA private key, resulting in the symmetric key (A^s). The recipient then receives the encrypted email message (C^t) and proceeds to verify the sender's ECC public key. Subsequently, the digital signature is received and verified for authenticity and data integrity. Using the decrypted symmetric key (A^s), the email message is decrypted, yielding the plaintext email (PT).

V. Performance Evaluation

The performance analysis is divided into two distinct approaches: analyzing the individual performance of RSA and ECC for secure email communication and assessing the performance of the hybrid approach for secure email communication.

A. Approach 1

1) Key Exchange Times measured in seconds: The key exchange time for RSA and ECC encryption methods in the context of secure email communication were assessed within the established testing environment. To measure key exchange times accurately, secure email communications were initiated, capturing the duration it took for public keys to be exchanged between sender and recipient during the initial email contact. This process was repeated 5 times for each test and the mean values were recorded. The recorded data for Key Exchange Time (KET) of RSA and ECC from the Secure Email Communication Test is as shown in the Table I. Fig. 3 illustrates the analysis of key exchange times for ECC and RSA encryption methods, represented in seconds.

2) Encryption and decryption times in seconds: Email encryption and decryption took place within the established test environment, consistent with the previously outlined specifications. Standardized email clients and servers were configured to support both RSA and ECC encryption methods. To assess these operations, a range of email messages, encompassing diverse sizes from text-based content to substantial attachments are employed as test data. The system was configured to autonomously record the encryption and decryption times for each email, ensuring an impartial and objective measurement of efficiency and practicality. This methodology facilitated a thorough analysis of email encryption and decryption processes using RSA and ECC techniques within the secure email communication system. This study further used the hybrid encryption setup; the asymmetric encryption algorithms (RSA or ECC) facilitate secure key exchange, while the symmetric encryption algorithm (AES) ensures efficient and high-speed encryption and decryption of the email content. This combination strikes a balance between security and performance, making it a practical choice for secure email communication.

TABLE I. Key Exchange Times of RSA and ECC

TEST	KET RSA (seconds)	KET ECC (seconds)
1	0.172268	0.101002
2	0.164218	0.102000
3	0.179985	0.091002
4	0.177888	0.102220
5	0.164782	0.091001

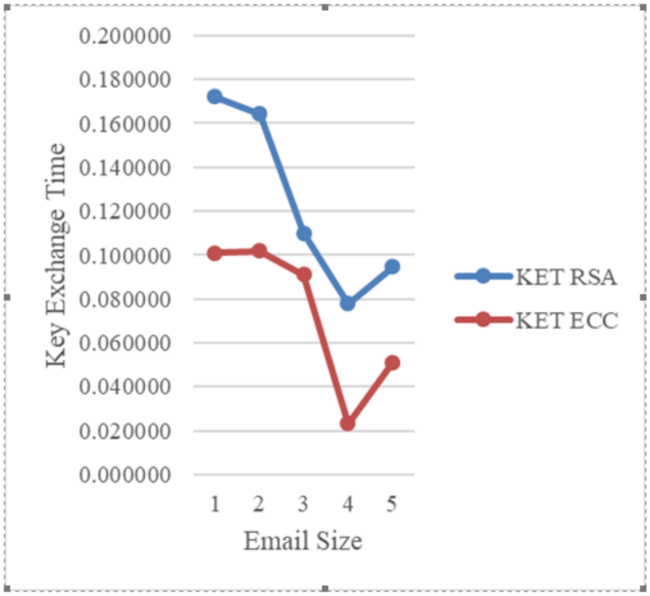

Fig. 3. Key exchange analysis of ECC and RSA (in seconds).

TABLE II. Encryption Time

Sizes (MB)	Encryption Time RSA (seconds)	Encryption Time ECC (seconds)
10	0.024308	0.028219
50	0.106565	0.094268
100	0.235220	0.173252
200	0.481745	0.375365
500	0.938921	0.866455

TABLE III. DECRYPTION TIME

Sizes (MB)	Decryption Time RSA (seconds)	Decryption Time ECC (seconds)
10	0.018158	0.016994
50	0.092173	0.068401
100	0.146187	0.153054
200	0.311700	0.308217
500	0.707802	0.753799

Fig. 4 and Fig. 5 depict the encryption and decryption times for RSA and ECC encryption methods, respectively, measured in seconds.

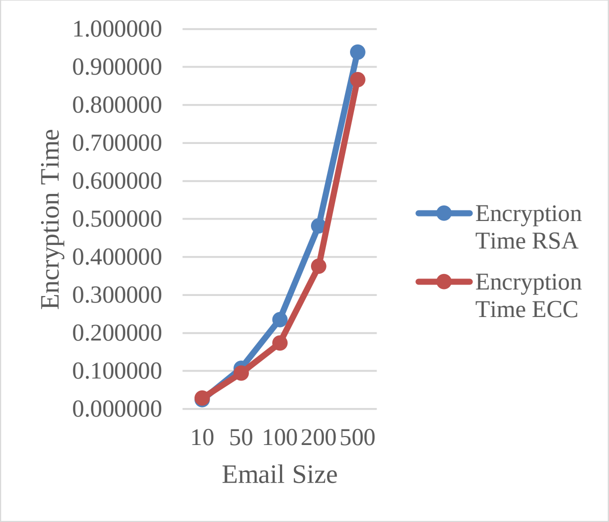

Fig. 4. RSA and ECC encryption time (in seconds).

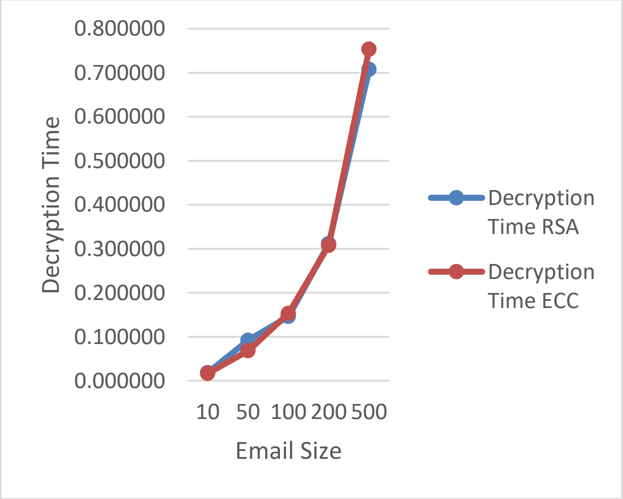

Fig. 5. RSA and ECC decryption time (in seconds).

3) Signature generation and verification: Signature generation and verification were integral aspects of the evaluation within the designated test environment, adhering to the established system specifications. By configuring standard email clients and servers to support both RSA and ECC encryption methods, the framework facilitated the generation of digital signatures for email messages. These signatures were generated autonomously during the test, and the system reported the time taken in seconds for each signature. Subsequently, the verification of these digital signatures occurred seamlessly within the same environment. A comprehensive analysis of signature generation and verification processes using RSA and ECC encryption methods was thus conducted objectively, with the system providing precise timing data for each operation.

Fig. 6 and Fig. 7 illustrate the signature generation and verification times for ECC and RSA encryption methods, respectively, measured in seconds.

TABLE IV. SIGNATURE GENERATION TIME

Sizes (MB)	SGT RSA (seconds)	SGT ECC (seconds)
10	0.030329	0.064428
50	0.127694	0.173007
100	0.278920	0.242878
200	0.451833	0.436492
500	1.239319	1.093490

TABLE V. SIGNATURE VERIFICATION TIME

Sizes (MB)	SVT RSA (seconds)	SVT ECC (seconds)
10	0.028464	0.030986
50	0.146251	0.144860
100	0.274981	0.236938
200	0.535547	0.544687
500	1.236818	1.253438

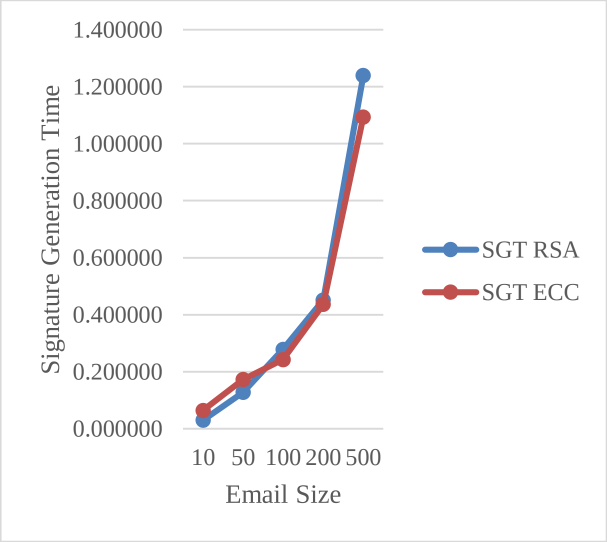

Fig. 6. Signature generation of ECC and RSA (in seconds).

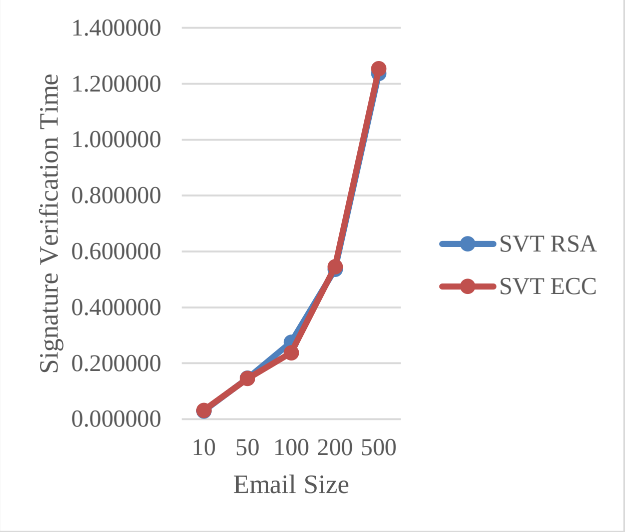

Fig. 7. Signature verification of ECC and RSA (in seconds).

B. Approach 2

1) Key exchange times measured in seconds: The key exchange time for the hybrid of RSA and ECC encryption methods were assessed within the established testing environment. To measure key exchange times accurately, secure email communications were initiated, capturing the duration it took the keys to be exchanged between sender and recipient during the initial email contact. This process was repeated five times for each test, and the mean values are recorded in Table VI.

TABLE VI. KEY EXCHANGE TIMES FOR HYBRID TECHNIQUE

TEST	KET (seconds)
1	0.064191
2	0.112602
3	0.070835
4	0.068739
5	0.067263

2) Encryption, decryption, signature generation and verification times in seconds: Table VII displays the encryption, decryption, signature generation, and verification performance metrics for the hybrid technique. To evaluate these operations, a variety of email messages with different sizes, ranging from text-based content to sizable attachments, were utilized as test data. The system was set up to automatically capture the times taken for encryption, decryption, signature generation, and verification for each email. For each experiment, this procedure was carried out five times, and the mean values were obtained.

TABLE VII. HYBRID TECHNIQUE (RSA AND ECC)

Sizes (MB)	EncryptionTime	Decryption Time	Signature Generation Time	Signature Verification Time
10	0.020769	0.014005	0.026106	0.026000
50	0.091153	0.062974	0.130278	0.120312
100	0.156722	0.140040	0.238006	0.262419
200	0.327086	0.307289	0.430858	0.423738
500	0.832917	0.636395	1.073605	1.160965

VI. DISCUSSION OF RESULTS

Table I presents the collected data concerning the Key Exchange Time (KET) for RSA and ECC in the context of secure email communication. The results consistently indicate that ECC outperforms RSA in terms of key exchange efficiency across all the tested scenarios. In practical terms, it implies that the process of establishing secure communication channels through key exchange is notably swifter and more efficient when utilizing ECC as the cryptographic algorithm, as opposed to RSA.

Table II reports the Encryption Time (in seconds) for both RSA and ECC in the same context. It illustrates the time taken to encrypt emails with various file sizes, ranging from 10 MB to 500 MB. The results demonstrate interesting trends in the performance of these cryptographic algorithms during the encryption process.

For smaller file sizes (10 MB and 50 MB), RSA exhibits slightly shorter encryption times compared to ECC. However, as the file sizes increase to 100 MB, 200 MB, and 500 MB, ECC consistently demonstrates superior efficiency in encryption. ECC's encryption times remain notably lower than RSA's for all these larger file sizes, suggesting that ECC is particularly well-suited for securing and transmitting larger email attachments. This outcome highlights ECC's efficiency in handling data encryption tasks for secure email communication, particularly when dealing with substantial file sizes. While RSA performs reasonably well for smaller files, ECC emerges as a more efficient choice as the data to be encrypted grows in size.

Just as Encryption Time from Table II, Decryption Time (in seconds) for both RSA and ECC in Table III provides insights into the time required to decrypt emails with various file sizes, ranging from 10 MB to 500 MB. The results reveal several noteworthy observations. For smaller file sizes (10 MB and 50 MB), ECC demonstrates slightly shorter decryption times compared to RSA, indicating its efficiency in handling smaller data. However, as the file sizes increase to 100 MB, 200 MB, and 500 MB, RSA exhibits competitive or slightly shorter decryption times than ECC. This suggests that RSA can be advantageous for decrypting larger email attachments efficiently.

Data presented in Table IV shows results for the Signature Generation Time (in seconds) for both RSA and ECC. The results reveal that for smaller file sizes (10 MB and 50 MB), RSA demonstrates notably shorter signature generation times compared to ECC, showcasing its efficiency in handling small data for signature creation. However, as file sizes increase to 100 MB, 200 MB, and 500 MB, ECC gradually catches up and, in some cases, surpasses RSA in terms of signature generation efficiency. This suggests that ECC is better suited for efficiently generating signatures for larger email attachments.

Table V provides insights into the time required to verify digital signatures for emails with same attached files as stated earlier. The results demonstrate interesting patterns in signature verification efficiency. For smaller file sizes (10 MB and 50 MB), ECC exhibits slightly longer verification times compared to RSA. However, as file sizes increase to 100 MB, 200 MB, and 500 MB, ECC's verification time becomes comparable to or slightly shorter than RSA's. This indicates that ECC is competitive with RSA in terms of signature verification efficiency, particularly for larger email attachments. The signature verification time findings suggest that ECC is a viable choice for verifying digital signatures, especially for larger data sizes. While RSA may have a slight advantage for smaller files, the efficiency of ECC becomes evident as the data size increases. The selection between RSA and ECC for signature verification should consider the typical email attachment sizes used in practice to optimize performance and efficiency.

Comparing Table I to Table VI, the hybrid technique demonstrates better key exchange time (KET) compared to solo RSA and ECC implementation. This indicates that the process of establishing secure communication channels through key exchange is notably swifter and more efficient when utilizing the proposed hybrid algorithm as opposed to RSA and ECC. Finally, Table VII presents the time (in seconds) recorded for encryption, decryption, signature generation, and signature verification in the context of secure email communication using the hybrid algorithm. When conducting a comparison with Tables II to V, it becomes evident that, on average, the hybrid encryption algorithm enhances the efficiency of encryption and decryption times, as well as signature generation and verification times. In certain instances, the individual ECC times displayed slightly better performance compared to the hybrid algorithm, indicating a close correlation between ECC and the hybrid approach. In summary, the proposed Hybrid technique excels in providing a versatile and efficient encryption solution for secure email communication across a wide range of email message sizes.

VII. Conclusion

Information technology services, like email systems, offer efficient solutions that are accessible to users regardless of their technical proficiency. These systems enable data storage, management, and local or internet-based access. However, the convenience of email usage also brings about a range of security vulnerabilities, encompassing unauthorized access, eavesdropping, identity impersonation, interception, and data tampering. This paper provides an analysis of key cryptographic algorithms, namely RSA, ECC and the hybrid algorithm in the context of securing email communications. The study reveals that ECC excels in terms of key exchange efficiency and effectively manages larger email attachments, making it an attractive choice for enhancing the security of modern email systems. While RSA performs adequately for smaller data sizes, ECC consistently outperforms it as data sizes increase, positioning it as a more efficient cryptographic algorithm for securing email communication. The experimental outcomes indicate that the suggested hybrid solution offers a more efficient method for encrypting email messages, with only a minimal disparity in runtime when compared to the ECC algorithm. Furthermore, this solution ensures a high level of security for secure email communication.

These findings offer valuable insights for practical optimization of email security. The hybrid algorithm introduced in this paper shows promise for being applied in system design, software development, and various other domains, offering an effective means of protecting data. In the future, this research can be further developed by enhancing the security of the hybrid approach. The integration of multiple security layers offers the potential to improve the system's productivity and efficiency.

Acknowledgment

Our gratitude to all the people and groups who helped to complete this study piece. Our gratitude is also expressed to the research team and our colleagues, who helped us out by sharing their skills, information, and encouragement. They played a critical role in carrying out the studies and collecting the results.

References

[1] Z. Kasiran, A. Dalil, and M. Z. Ghazali, “Analysis on Computational Time of Hybrid Cryptography in Email System,” J. Posit. Sch. Psychol., vol. 2022, no. 3, pp. 8415–8422, 2022, [Online]. Available: http://journalppw.com.

[2] Z. Alkhalil, C. Hewage, L. Nawaf, and I. Khan, “Phishing Attacks: A Recent Comprehensive Study and a New Anatomy,” Front. Comput. Sci., vol. 3, no. March, pp. 1–23, 2021, doi: 10.3389/fcomp.2021.563060.

[3] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, “A Comprehensive Review of Cyber Security Vulnerabilities, Threats, Attacks, and Solutions,” Electron., vol. 12, no. 6, 2023, doi: 10.3390/electronics12061333.

[4] G. B. Thompson, “Journal of information Science,” J. Inf. Sci., vol. 9, no. 2, p. 74, 1984, doi: 10.1177/016555158400900204.

[5] F. Mallouli, A. Hellal, N. Sharief Saeed, and F. Abdulraheem Alzahrani, “A Survey on Cryptography: Comparative Study between RSA vs ECC Algorithms, and RSA vs El-Gamal Algorithms,” Proc. - 6th IEEE Int. Conf. Cyber Secur. Cloud Comput. CSCloud 2019 5th IEEE Int. Conf. Edge Comput. Scalable Cloud, EdgeCom 2019, pp. 173–176, 2019, doi: 10.1109/CSCloud/EdgeCom.2019.00022.

[6] A. Karki, “A Comparative Analysis of Public Key Cryptography,” Int. J. Mod. Comput. Sci., vol. 4, no. 6, pp. 2320–7868, 2016, [Online]. Available: http://www.iusikkim.edu.in/IJMCS161213.pdf.

[7] R. M. Abobeah, M. M. Ezz, and H. M. Harb, “Public-Key Cryptography Techniques Evaluation,” Int. J. Comput. Networks Appl., vol. 2, no. 2, pp. 64–75, 2015.

[8] M. Boussif, “Scalable Implementation of Array of 8-bit-Based RSA With Large Key Size,” Proc. 2022 5th Int. Conf. Adv. Syst. Emergent

Technol. IC_ASET 2022, pp. 375–380, 2022, doi: 10.1109/IC_ASET53395.2022.9765873.

[9] D. Mahto and D. K. Yadav, "RSA and ECC,A Comparative Analysis.pdf," Int. J. Appl. Eng. Res., vol. 12, no. 19, pp. 9053–9061, 2017.

[10] M. Bansal, S. Gupta, and S. Mathur, "Comparison of ECC and RSA Algorithm with DNA Encoding for IoT Security," Proc. 6th Int. Conf. Inven. Comput. Technol. ICICT 2021, pp. 1340–1343, 2021, doi: 10.1109/ICICT50816.2021.9358591.

[11] V. O. Nyangaresi, A. J. Rodrigues, and S. O. Abeka, "Secure Algorithm for IoT Devices Authentication," EAI/Springer Innov. Commun. Comput., no. January, pp. 1–22, 2023, doi: 10.1007/978-3-030-92968-8_1.

[12] S. Ahmed and T. Ahmed, "Comparative Analysis of Cryptographic Algorithms in Context of Communication: A Systematic Review," Int. J. Sci. Res. Publ., vol. 12, no. 7, pp. 161–173, 2022, doi: 10.29322/ijsrp.12.07.2022.p12720.

[13] P. Kaur and S. Aggarwal, "Cryptographic algorithms in IoT - a detailed analysis," Proc. - 2021 2nd Int. Conf. Comput. Methods Sci. Technol. ICCMST 2021, pp. 45–50, 2021, doi: 10.1109/ICCMST54943.2021.00021.

[14] S. Al Busafi and B. Kumar, "Review and analysis of cryptography techniques," Proc. 2020 9th Int. Conf. Syst. Model. Adv. Res. Trends, SMART 2020, pp. 323–327, 2020, doi: 10.1109/SMART50582.2020.9336792.

[15] C. Varma, "A Study of the ECC, RSA and the Diffie-Hellman Algorithms in Network Security," Proc. 2018 Int. Conf. Curr. Trends Towar. Converging Technol. ICCTCT 2018, pp. 18–21, 2018, doi: 10.1109/ICCTCT.2018.8551161.

[16] H. Eberle, N. Gura, S. C. Shantz, V. Gupta, L. Rarick, and S. Sundaram, "A public-key cryptographic processor for RSA and ECC," Proc. Int. Conf. Appl. Syst. Archit. Process., pp. 98–110, 2004, doi: 10.1109/ASAP.2004.1342462.

[17] O. Ponomarev, A. Khurri, and A. Gurtov, "Elliptic Curve Cryptography (ECC) for Host Identity Protocol (HIP)," 9th Int. Conf. Networks, ICN 2010, pp. 215–219, 2010, doi: 10.1109/ICN.2010.68.

[18] M. Bafandehkar, S. M. Yasin, R. Mahmod, and Z. M. Hanapi, "Comparison of ECC and RSA algorithm in resource constrained devices," 2013 Int. Conf. IT Converg. Secur. ICITCS 2013, pp. 9–11, 2013, doi: 10.1109/ICITCS.2013.6717816.

[19] M. Suarez-Albela, T. M. Fernandez-Carames, P. Fraga-Lamas, and L. Castedo, "A practical performance comparison of ECC and RSA for resource-constrained IoT devices," 2018 Glob. Internet Things Summit, GIoTS 2018, pp. 0–5, 2018, doi: 10.1109/GIOTS.2018.8534575.

[20] A. Kardi, R. Zagrouba, and M. Alqahtani, "Performance Evaluation of RSA and Elliptic Curve Cryptography in Wireless Sensor Networks," 21st Saudi Comput. Soc. Natl. Comput. Conf. NCC 2018, vol. 65537, pp. 302–306, 2018, doi: 10.1109/NCG.2018.8592963.

[21] S. R. Singh, A. K. Khan, and T. S. Singh, "A critical review on Elliptic Curve Cryptography," Int. Conf. Autom. Control Dyn. Optim. Tech. ICACDOT 2016, vol. 3, no. 7, pp. 13–18, 2017, doi: 10.1109/ICACDOT.2016.7877543.

[22] H. Hasan et al., "Secure lightweight ECC-based protocol for multi-agent IoT systems," Int. Conf. Wirel. Mob. Comput. Netw. Commun., vol. 2017-Octob, 2017, doi: 10.1109/WiMOB.2017.8115788.

[23] D. Mahto, D. A. Khan, and D. K. Yadav, "Security analysis of elliptic Curve cryptography and RSA," Lect. Notes Eng. Comput. Sci., vol. 2223, pp. 419–422, 2016.

[24] D. Patel, B. Patel, J. Vasa, and M. Patel, A Comparison of the Key Size and Security Level of the ECC and RSA Algorithms with a Focus on Cloud / Fog. Springer Nature Singapore, 2023. doi: 10.1007/978-981-99-3758-5.

[25] L. Zou, M. Ni, Y. Huang, W. Shi, and X. Li, Hybrid Encryption Algorithm Based on AES and RSA in File Encryption, vol. 551 LNEE. Springer Singapore, 2020. doi: 10.1007/978-981-15-3250-4_68.

[26] S. Sa'idu, P. Taneja, and K. Shreya, "A Comparative Analysis of Cryptographic Algorithms : AES & RSA and Hybrid Algorithm for Encryption and Decryption," Int. J. Innov. Sci. Res. Technol., vol. 7, no. 8, pp. 1725–1732, 2022.

[27] S. Rehman, N. Talat Bajwa, M. A. Shah, A. O. Aseeri, and A. Anjum, "Hybrid aes-ecc model for the security of data over cloud storage," Electron., vol. 10, no. 21, pp. 1–20, 2021, doi: 10.3390/electronics10212673.

[28] Y. Al-Dhuraibi, F. Paraiso, N. Djarallah, and P. Merle, "Elasticity in Cloud Computing: State of the Art and Research Challenges," IEEE Trans. Serv. Comput., vol. 11, no. 2, pp. 430–447, 2018, doi: 10.1109/TSC.2017.2711009.

[29] D. Kodzo, M. Hodowu, D. R. Korda, and E. Danso Ansong, "An Enhancement of Data Security in Cloud Computing with an Implementation of a Two-Level Cryptographic Technique, using AES and ECC Algorithm," Int. J. Eng. Res. Technol., vol. 9, no. March 2021, pp. 2278–0181, 2020, [Online]. Available: http://www.ijert.org

[30] M. Sivajyothi and T. Devi, "Analysis of Elliptic Curve Cryptography with AES for Protecting Data in Cloud with improved Time efficiency," Proc. 2nd Int. Conf. Innov. Pract. Technol. Manag. ICIPTM 2022, no. Bhosle 2013, pp. 573–577, 2022, doi: 10.1109/ICIPTM54933.2022.9753926.

[31] B. Ranganatha Rao and B. Sujatha, "A hybrid elliptic curve cryptography (HECC) technique for fast encryption of data for public cloud security," Meas. Sensors, vol. 29, no. June, p. 100870, 2023, doi: 10.1016/j.measen.2023.100870.

[32] M. J. Dubai, T. R. Mahesh, and P. A. Ghosh, "Design of new security algorithm: Using hybrid Cryptography architecture," ICECT 2011 - 2011 3rd Int. Conf. Electron. Comput. Technol., vol. 5, pp. 99–101, 2011, doi: 10.1109/ICECTECH.2011.5941965.

[33] S. K. Ghosh, S. Rana, A. Pansari, J. Hazra, and S. Biswas, "Hybrid Cryptography Algorithm for Secure and Low Cost Communication," 2020 Int. Conf. Comput. Sci. Eng. Appl. ICCSEA 2020, pp. 4–8, 2020, doi: 10.1109/ICCSEA49143.2020.9132862.