A Systematic Literature Review of the Elliptic Curve Cryptography (ECC) Algorithm for Encrypting Data Sharing in Cloud Computing

Poovendran Alagarsundaram

Humetis Technologies Inc,

Kingston, NJ, USA

Email: poovendrana@ieee.org

ABSTRACT

Cloud computing has transformed the information technology environment by allowing users to share resources, services, and data across a network. Despite its various benefits, such as scalability, flexibility, and cost-effectiveness, cloud computing raises serious security concerns, notably about data confidentiality and integrity during transmission. Traditional encryption approaches, such as the Advanced Encryption Standard (AES), have helped to secure data, but their limitations in computing efficiency and resource consumption compel the exploration of other ways. This research conducts a comprehensive literature review on Elliptic Curve Cryptography (ECC), an effective encryption technology for safeguarding data sharing in cloud computing. ECC is known for its strong security, reduced key sizes, and faster processing, making it an excellent choice for cloud environments. The work investigates the mathematical foundations of ECC, such as point addition and point doubling, and proposes a systematic approach to ECC-based encryption and decryption. Furthermore, it compares ECC's performance and security to AES, emphasizing ECC's advantages in resource utilization and computational efficiency. The findings indicate that ECC can greatly improve data security in cloud computing while optimizing resource efficiency, resulting in a scalable and cost-effective solution for modern data encryption requirements.

Keywords: Cloud computing, data encryption, elliptic curve cryptography (ECC), computational efficiency, resource utilization, data security.

1. INTRODUCTION

Cloud computing has transformed the information technology (IT) landscape by allowing the sharing of resources, services, and data across a network. This innovative technology provides multiple advantages, including scalability, adaptability, and cost-efficiency, which have fueled its rapid adoption across a variety of industries. Organisations use cloud computing to improve operational capabilities, communication, and process efficiencies. However, the intrinsic nature of cloud computing, which involves storing and delivering data across the internet, presents substantial security challenges. Ensuring the confidentiality, integrity, and availability of data in a cloud environment is critical, especially as cyber threats become more complex and sophisticated.

The swift assimilation of cloud computing in recent times has revolutionised the way enterprises handle and share information. Scalability and efficiency are two benefits of the cloud, but there are also serious security risks associated with it, especially when it comes to data privacy and transmission security. Although scalable and computationally efficient, traditional encryption techniques like AES have proved essential to data security. Elliptic Curve Cryptography (ECC) is investigated in this comprehensive literature study as a suitable substitute encryption technique for safeguarding cloud data sharing. Enhancing data security in cloud computing can be achieved with ECC because of its potential to provide robust protection with reduced key sizes and faster computation. This paper attempts to provide a thorough knowledge of ECC's effectiveness and usefulness in solving cloud data security challenges by synthesising existing research and ideas.

Many resources are available for consumers to access and share over the internet thanks to cloud computing. Its ability to save expenses and offer scalable solutions that satisfy a range of company demands has led to its widespread adoption. The requirement for physical infrastructure is greatly reduced since businesses can now store enormous volumes of data, run apps, and administer services remotely. Despite these benefits, using cloud computing raises important security issues, especially with regard to protecting and privacy of data. Data is vulnerable to a range of cyber dangers during its transfer between clients and cloud servers, such as eavesdropping, illegal access, and data breaches.

Encryption has developed into a vital security feature in cloud computing to reduce these dangers. The Advanced Encryption Standard (AES) has long been a popular choice for data security. Although AES is renowned for its stability and dependability, it has drawbacks in terms of computational effectiveness and resource usage, particularly when data transmission volumes and frequencies increase. These drawbacks make it necessary to investigate alternate encryption techniques that provide higher security at a higher efficiency.

Data security is still a major worry in spite of the advances made in cloud computing. Conventional security procedures sometimes overlook the protection of data while it is being transmitted, instead concentrating mostly on safeguarding data that is at rest within cloud storage systems. Particularly open to interception, eavesdropping, and other malicious activity are data in transit. Although widely used to secure data, AES has limits in terms of computing performance and resource utilization. These restrictions become more noticeable with increasing data quantities and transmission frequency. Due to the considerable computational cost associated with AES encryption and its comparatively large key sizes, encryption and decryption processes take longer and require more resources. Thus, the need for a more effective and safe encryption technique to safeguard data while it is being transmitted in cloud environments is urgent.

Elliptic Curve Cryptography (ECC) in cloud computing is the subject of this study, which aims to:

- Mathematical Foundations: Examine elliptic curves, point addition, and point doubling, as well as the concepts and procedures of ECC.
- Development of Methodology: To guarantee data security during transmission, establish a methodical procedure for ECC-based encryption and decryption in cloud settings.

- Comparing the Performance and Security of ECC vs AES: Assess the benefits of ECC for cloud applications on resource usage, security, and computational efficiency.
- Provide usable advice and guidance for implementing ECC to improve data security tactics in practical cloud computing environments.

The lack of study on Elliptic Curve Cryptography (ECC), particularly as it relates to encrypting data while it is being sent in cloud computing environments, represents a research gap. Although ECC is known for its efficiency and security advantages, most of the material that has been written about it has been applied to theoretical cryptography research or to the security of stored data. When it comes to safeguarding data in transit, actual research and useful implementation instructions that show how successful ECC is in real-world cloud environments are conspicuously lacking. Closing this gap is essential since more and more businesses are exchanging data via cloud services, which means that strong encryption techniques that guarantee data integrity and confidentiality during transmission are required.

2. LITERATURE SURVEY

Chang et al. (2022) present a novel asymmetric cryptosystem that uses biometric photos as private keys and combines optical scanning and elliptic curve encryption. A separate biometric private key from the recipient is utilised for decryption, whereas the sender's biometric image is used for encryption. Numerical and experimental results validate the practicality of this technique, which provides higher security than conventional optical scanning cryptography.

In order to solve security flaws in the current protocols, Lu and Zhao (2022) propose a new SIP authenticated key agreement protocol (AKAP) that makes use of elliptic curve cryptography (ECC). The protocol's security is tested with the AVISPA simulation tool, and its correctness is assessed using the Burrows-Abadi-Needham (BAN) method. When compared to earlier protocols, such as Zhang, Tang, and Zhu's (2015) protocol, the new protocol shows enhanced security, exposing the earlier protocol's susceptibility to key-compromise impersonation attacks.

Yadav and Tiwari (2022) suggest utilising asymmetric pairing with shorter ciphertext as an effective and safe way to exchange private medical data. This approach makes use of cloud computing capabilities and is specifically developed for quick learning in the healthcare industry.

By using Type-III pairings, the most secure and efficient pairing type available, it is more effective and efficient than conventional broadcast encryption systems. Under the decisional BDHE complexity assumption, the method's semantic security is demonstrated, which qualifies it for real-time data exchange in cloud contexts.

Wang et al. (2022) suggest utilising asymmetric pairing with shorter ciphertext as an effective and safe way to exchange private medical data. This approach makes use of cloud computing capabilities and is specifically developed for quick learning in the healthcare industry. By using Type-III pairings, the most secure and efficient pairing type available, it is more effective and efficient than conventional broadcast encryption systems. Under the decisional BDHE complexity assumption, the method's semantic security is demonstrated, which qualifies it for real-time data exchange in cloud contexts.

Peng et al. (2022) present a safe technique for homomorphic encryption-based blockchain data transmission that guarantees data dependability and accuracy. This technique successfully prevents data tampering and distortion by achieving a high transmission accuracy of 88% and a brief transmission time. For blockchain data transfer, IoT devices gather asymmetric encrypted public keys, which provide improved transmission accuracy and faster transmission times over conventional techniques.

Nair et al. (2022) investigate the application of blockchain technology to create a safe, decentralised trust framework for cloud computing data transfers. The study focuses on data migration using content-addressing from typical cloud platforms to distributed file systems such as the InterPlanetary File System (IPFS). By doing away with the necessity for a centralised trust mechanism, this decentralised method improves data transaction security and traceability. The study emphasises how blockchain technology may help with trust and security issues that arise in cloud computing environments. These benefits include integrity, non-repudiation, and security.

Fazal et al. (2022) stress the vital importance of data privacy while putting forth a novel method for protecting Covid-19 patient data in decision support systems. Their strategy aims to prevent unauthorised access by securing patient names and sensitive attributes through the use of pseudonymization and Blowfish encryption. The study highlights the potential for unauthorised individuals to get harmful access to patient data and assesses how well the suggested methodology works to reduce this risk and guarantee data security.

A novel secure data security architecture for cloud computing is put forth by Sauber et al. (2021) in an effort to address problems like data breaches and fictitious user identities. For increased protection against intrusions and unauthorised access, the model makes use of encryption and one-time passwords. Enhanced security for end users and data owners is demonstrated by the model, which was tested and implemented using the Next Generation Secure Cloud Server (NG-Cloud) simulation.

Aldabbagh et al. (2021) present a safe cloud-based mobile learning system that encrypts data using the hybrid optimum elliptic curve cryptography (HOECC) algorithm. High degrees of authentication, data integrity, and confidentiality are ensured by this method, which uses an adaptable tunicate slime-mold (ATS) algorithm to create ideal key values. 50 students participated in a survey as part of the study to evaluate the system's effectiveness, and the results showed quicker encryption and decryption speeds than with previous methods. The efficacy of merging HOECC and ATS algorithms to improve security and productivity in cloud-based mobile learning environments is demonstrated by this study.

A thorough analysis of the literature on cloud computing security is conducted by Alouffi et al. (2021), with an emphasis on dangers and difficulties. Seven significant security risks, including data leakage and tampering, are identified by the research as being common in cloud computing systems. It proposes using blockchain technology as a possible remedy to allay these worries. The study emphasises how important it is for future research to focus on data availability, confidentiality, and integrity in cloud computing in order to improve overall security measures. There are still many studies and applications in the field of cloud computing, and there are continuous attempts to successfully handle the security issues that come with it.

Two scalable and secure methods, P2GT and P2GT+, are proposed by Huang et al. (2021) to manage access to genetic data and enable personalised medicine tests in cloud computing. In order to easily conduct encrypted single nucleotide polymorphism (SNP)-based personalised medicine testing, P2GT integrates an equality test algorithm into its key-policy attribute-based encryption (KP-ABE) system for access management. By adding identity-based encryption (IBE) to this architecture, P2GT+ improves it and allows for flexible joint authorization for genetic compatibility, disease susceptibility, and privacy-preserving paternity tests. Extensive studies on the 1,000 Genomes dataset verify both techniques, confirming their scalability and feasibility for authorised and safe genetic testing in cloud contexts.

3. METHODOLOGY FOR DATA ENCRYPTION USING ELLIPTIC CURVE CRYPTOGRAPHY (ECC)

Cloud computing has revolutionised the landscape of information technology (IT) by allowing users to share resources, services, and data across a network. However, this common environment leaves data vulnerable to unauthorised access. Traditionally, data security in cloud computing has been largely concerned with securing stored data through internet security measures, with less emphasis placed on protecting data during transit. Recognising the crucial relevance of data security, this methodology recommends using encryption to assure data security during transmission. Instead of using the Advanced Encryption Standard (AES), this method leverages Elliptic Curve Cryptography (ECC) to improve security and operational efficiency.

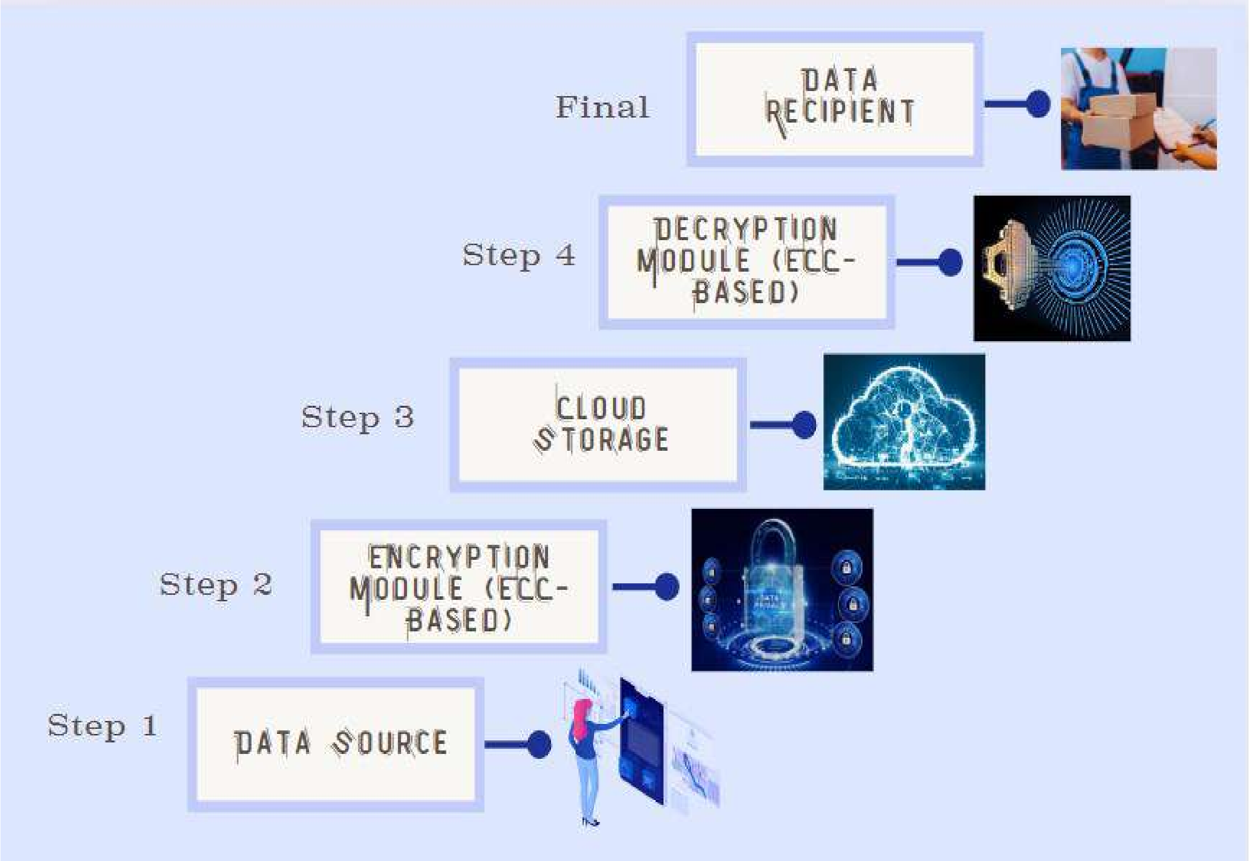

Figure 1: Architecture of ECC-Based Encryption System in Cloud Computing.

The elements and operation of an ECC-based encryption system in a cloud computing environment are shown in figure 1. The data source, encryption module, cloud storage, decryption module, and data recipient are some of its components.

Feature	AES	ECC
Key Size	128-bit, 192-bit, 256-bit	160-bit, 224-bit, 256-bit, 384-bit
Security Level	Moderate to High	High (with smaller key sizes)
Computational Efficiency	Higher computational overhead	Lower computational overhead
Encryption/Decryption Speed	Slower (due to larger keys)	Faster (due to smaller keys)

Resource Usage	Higher (more memory and power)	Lower (less memory and power)
Suitable for Cloud Computing	Yes, but less efficient	Yes, more efficient and secure
Common Use Cases	General data encryption	Secure communications, digital signatures

Table 1: Comparison of AES and ECC for Data Encryption in Cloud Computing.

3.1. Mathematical Foundations of ECC

3.1.1. Elliptic Curves:

An elliptic curve over a finite field F_p (where p is a prime number) is described by the equation:

$$y^2 = x^3 + ax + b \; mod \; p \tag{1}$$

Here, a and b are coefficients that must satisfy the condition:

$$4a^3 + 27b^2 + 0 \; mod \; p \tag{2}$$

The set of solutions (x, y) to this equation, along with a special point at infinity, form the elliptic curve. The elliptic curve is made up of a unique point at infinity and the set of solutions $(x,)$ to this equation. For the group that is defined by the points on the curve, the point at infinity acts as the identity element.

3.1.2. Point Addition and Doubling:

ECC relies on two main operations: Arithmetic on the points of the elliptic curve is done using point addition and point doubling. Cryptographic protocols based on elliptic curves are built on these processes.

1. *Point Addition:* Given two distinct points $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ on the elliptic curve, their sum $R = P + Q$ is determined by:

$$\lambda - \frac{y_2 - y_1}{x_2 - x_1} \; mod \; p \tag{3}$$

$$x_3 - \lambda^2 - x_1 - x_2 \; mod \; p \tag{4}$$

$$y_3 - \lambda(x_1 - x_3) - y_1 \; mod \; p \tag{5}$$

If $x_1 = x_2$ and $y_1 = -y_2, R$ is the point at infinity.

2. *Point Doubling:* When $P = Q$, the point doubling formula is used:

$$\lambda - \frac{3x_1^2+a}{2y_1} \, mod \, p \tag{6}$$

$$x_3 - \lambda^2 - 2x_1 \, mod \, p \tag{7}$$

$$y_3 - \lambda(x_1 - x_3) - y_1 \, mod \, p \tag{8}$$

Complex cryptographic protocols can be built thanks to these processes.

Algorithm: Point Addition

Input: Points P, Q

Output: Resulting point R

```
Begin
    If P is (0, 0)
        Return Q
    If Q is (0, 0)
        Return P
    Extract coordinates x1, y1 from P
    Extract coordinates x2, y2 from Q
    If x1 equals x2 and y1 equals -y2
        Return (0, 0)
    If P equals Q
        Compute λ using point doubling formula: $λ = (3x1^2 + a) / (2y1) mod p$
    Else
        Compute λ using point addition formula: $λ = (y2 - y1) / (x2 - x1) mod p$
    Compute x3: $x3 = λ^2 - x1 - x2 mod p$
    Compute y3: $y3 = λ(x1 - x3) - y1 mod p$
    Return resulting point (x3, y3)
End
```

3.2. Data Encryption and Decryption Using the ECC Algorithm

The three primary steps of the ECC algorithm are key generation, encryption, and decryption. The aforementioned mathematical operations on elliptic curves are used in each of these procedures.

Key Generation:

In elliptic curve cryptography (ECC), key generation entails choosing particular parameters and creating a pair of keys (public and private). Initially choose the elliptic curve parameters a, b, and p. These parameters define the elliptic curve equation $y^2 = x^3 + ax + b \ mod \ p$. A base point, or $G = \left(G_x, G_y\right)$ on the curve, should also be selected. This point acts as the beginning point for all operations. Next, create a private key called d, which is a random integer between $[1, n - 1]$, where n is the order of the base point G. The base point G is multiplied by the private key d to obtain the public key Q, which is then calculated using scalar multiplication, which results in $Q = dG$. An elliptic curve point represents this public key. A public key called Q and a private key called D are the results of this process.

- ✓ Select Elliptic Curve Parameters: Choose a, b_r and p.
- ✓ Select a Base Point : Choose a point $G = \left(G_x, G_y\right)$ on the elliptic curve.
- ✓ Generate Private Key : Select a random integer d in the range $[1, n - 1]$, where n is the order of G.
- ✓ Generate Public Key : Compute $Q = dG$ using scalar multiplication.

```
Algorithm: Key Generation
Input: Curve parameters a, b, p, base point G, order n
Output: Private key d, Public key Q

Begin
    Select curve parameters a, b, p, and base point G
    Generate a random integer d in the range [1, n-1]
    Compute the public key Q by performing scalar multiplication: Q = dG
    Return private key d and public key Q
End
```

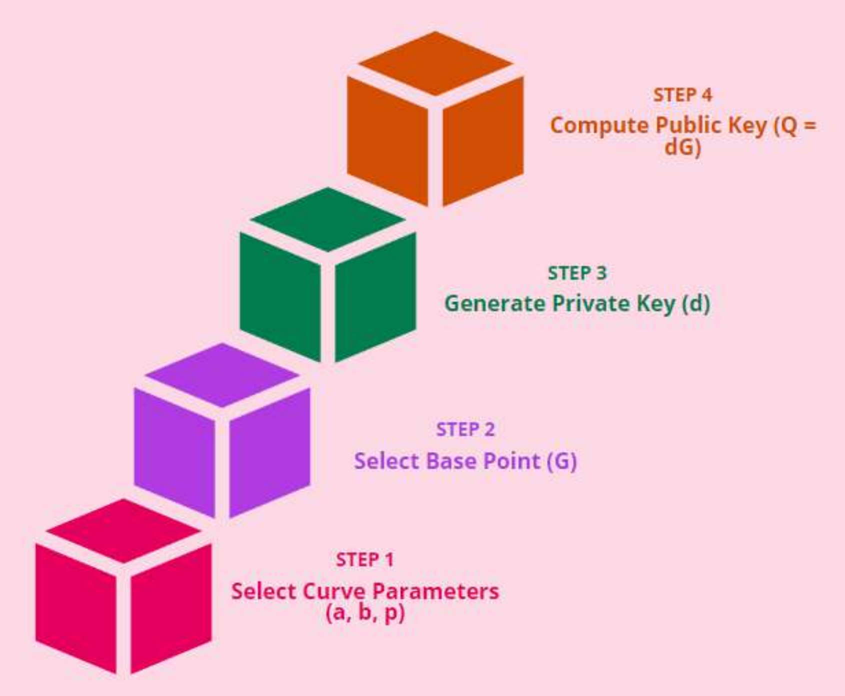

Figure 2: Elliptic Curve Cryptography (ECC) Key Generation.

The process of key generation in ECC is shown in figure 2. The first step is to choose a base point and the elliptic curve's parameters. The private key is selected at random from a range of integers. The base point is then multiplied by a scalar factor using the private key to determine the public key.

Encryption:

ECC encryption turns a plaintext message into ciphertext using the recipient's public key. To begin, map the plaintext message M to the point P on the elliptic curve. Choose a random number k from the range [1, n -1]. Two cypher points are computed: $C_1 = kG$ and $C_2 = P_M + kQ$, where G G is the base point and Q Q is the recipient's public key. Here, $k\ G$ kG ensures the unpredictability of the encryption process, while kQ is utilised to conceal the message point P_M. The ciphertext consists of the pair of points (C_1, C_2) and is transmitted to the recipient.

- ✓ Represent Message as a Point : Map the plaintext message to a point M on the elliptic curve.
- ✓ Select a Random Integer : Choose k in the range $[1, n - 1]$.
- ✓ Compute Cipher Points C_1 and C_2 :

$$C_1 = kG \tag{9}$$

$$C_2 = P_M + kQ \tag{10}$$

✓ Transmit (C_1, C_2) : Send the pair of points as the ciphertext.

```
Algorithm: Encryption
Input: Message M (mapped to point P_M), Public key Q, Base point G, Order n
Output: Ciphertext (C_1, C_2)

Begin
    Map the plaintext message M to a point P_M on the elliptic curve
    Select a random integer k in the range [1, n-1]
    Compute the cipher points:
        C_1 = kG
        S = kQ
        C_2 = P_M + S
    Return ciphertext (C_1, C_2)
End
```

Decryption:

Decryption in ECC entails reversing the encryption process to recover the original plaintext message from the ciphertext. After receiving the ciphertext points (C_1, C_2), the recipient uses their private key (d) to compute the shared secret $S = dC_1$. This shared secret is critical since it helps isolate the original message. To recover the original message point P_M, remove the shared secret from the second cypher point: $P_M = C_2 - S$. The decrypted message is the point $P_M$on the elliptic curve, corresponding to the original plaintext message M. This technique assures that only the recipient, who has the private key, may decrypt and recover the original message.

✓ Receive Cipher Points (C_1, C_2) : Obtain the transmitted ciphertext.

✓ Compute Shared Secret S:

$$S = dC_1 \tag{11}$$

In this case, d stands for the recipient's private key, and $C_1 = kG$. As $C_1 = d(kG) = k(dG) = kQ$, then can compute the shared secret S using the private key and the first cypher point since $C_1 = d(kG) = k(dG) = kQ$.

✓ Recover Original Message M :

$$M = C_2 - S \tag{12}$$

Given $C_2 = P_M + kQ$ from the encryption step, and $S = kQ$, subtracting S from C_2 yields:

$$M = P_M + kQ - kQ = P_M \tag{13}$$

This correctly recovers the original message point P_M.

Correctness Verification

Shared Secret Calculation:

$$S = dC_1 = d(kG) = k(dG) = kQ \tag{14}$$

Message Recovery:

$$M = C_2 - S = (P_M + kQ) - kQ = P_M \tag{15}$$

Thus, the decryption equations $S = dC_1$ and $M = C_2 - S$ are indeed correct. The decryption process accurately retrieves the original message point P_M from the ciphertext.

```
Algorithm: Decryption
Input: Ciphertext (C_1, C_2), Private key d
Output: Plaintext message M (point P_M)

Begin
    Receive the ciphertext points (C_1, C_2)
    Compute the shared secret S using scalar multiplication: S = dC_1
    Recover the original message point P_M:
        P_M = C_2 - S
    Return plaintext message M (point P_M)
End
```

ECC offers strong security with much reduced key sizes, making it equivalent to more conventional encryption techniques like RSA and AES. For example, the security of a 256-bit ECC key is equal to that of a 3072-bit RSA key. Because smaller keys require fewer resources and faster computations, ECC is especially well-suited for settings where resource limits and computational efficiency are crucial. The reduced key sizes of ECC are largely responsible for its efficiency. Smaller keys enable faster encryption and decryption procedures by lessening the computing load on devices. This is especially helpful in cloud environments, where managing resources effectively is necessary to serve several users and applications.

Strong and effective encryption that works well in cloud computing settings is elliptic curve cryptography (ECC). Today's data encryption requirements can be perfectly met by using ECC instead of AES because it offers high security levels with less computational complexity. Providing a thorough manual for safeguarding data transfer in cloud computing, this methodology describes the mathematical underpinnings, intricate algorithms, and real-world application of ECC.

4. RESULT AND DISCUSSION

A major emphasis on the Advanced Encryption Standard (AES) for data encryption in cloud computing is highlighted by the systematic literature review. Since AES is generally recognized for its consistency and dependability, many cloud environments use it as the foundation for data security. However, the limitations of AES's computational efficiency and resource utilization become more evident when cloud computing grows in size and data transmission volume rises. Studies reveal that AES's significant computational overhead, especially when dealing with bigger key sizes, might result in increased resource usage, which can affect the overall efficiency and performance of the system. This makes it necessary to investigate other encryption techniques that can offer strong security without sacrificing effectiveness.

A viable substitute for AES is Elliptic Curve Cryptography (ECC), especially when considering cloud computing. Because ECC uses smaller keys, it can achieve significant efficiency gains in terms of faster computations and less resource consumption. ECC keeps security levels high even with these reduced key sizes. A 256-bit ECC key, for instance, offers security on par with a 3072-bit RSA key. Because of its effectiveness, ECC is especially well-suited for cloud environments, where supporting a large number of users and applications requires efficient resource management. Because less power and memory are needed for encryption and decryption procedures, the decreased computational burden linked with ECC improves performance while also helping cloud operations save money.

It is clear from the comparison of AES and ECC that using ECC for data encryption in cloud computing may have advantages. Although AES has long been the industry standard for data security, its inefficiencies in managing large-scale data transmission call for the development of more potent remedies. Because ECC may provide robust protection with less computational overhead, it is a good choice for improving cloud data security. Although updating encryption techniques and key management procedures is necessary, the switch to ECC offers long-term advantages in terms of enhanced performance and cost effectiveness. In order to handle the constantly changing security landscape and guarantee strong data security while maximizing resource usage, this paper promotes the use of ECC in cloud systems. Organizations can strike a balance between security and efficiency by utilizing ECC's advantages, opening the door for more scalable and affordable cloud computing solutions.

5. CONCLUSION

The systematic literature review highlights the potential of Elliptic Curve Cryptography (ECC) as a better alternative to the Advanced Encryption Standard (AES) for encrypting data in cloud computing. ECC's ability to provide high levels of security while using lower key sizes and faster computations makes it ideal for cloud contexts. The comparative analysis shows that ECC can overcome the inefficiencies associated with AES, especially in large-scale data transmission scenarios. By implementing ECC, enterprises can increase data security, performance, and cost savings, opening the door for more efficient and secure cloud computing solutions. Future research should focus on defining feasible ECC implementation guidelines for a variety of cloud computing environments, as well as investigating its integration with new technologies such as blockchain and homomorphic encryption.

REFERENCE

1. Lu, Y., & Zhao, D. (2022). An anonymous SIP authenticated key agreement protocol based on elliptic curve cryptography. management, 12, 14.
2. Chang, X., Li, W., Yan, A., Tsang, P. W. M., & Poon, T. C. (2022). Asymmetric cryptosystem based on optical scanning cryptography and elliptic curve algorithm. Scientific Reports, 12(1), 7722.
3. Chang, X., Li, W., Yan, A., Tsang, P. W. M., & Poon, T. C. (2022). Asymmetric cryptosystem based on optical scanning cryptography and elliptic curve algorithm. Scientific Reports, 12(1), 7722.
4. Yadav, S., & Tiwari, N. (2022). An efficient and secure data sharing method using asymmetric pairing with shorter ciphertext to enable rapid learning in healthcare. Computational Intelligence and Neuroscience, 2022(1), 4788031.
5. Wang, Y., Blobel, B., & Yang, B. (2022). Reinforcing health data sharing through data democratization. Journal of Personalized Medicine, 12(9), 1380.
6. Peng, S., Cai, Z., Liu, W., Wang, W., Li, G., Sun, Y., & Zhu, L. (2022). Blockchain data secure transmission method based on homomorphic encryption. Computational intelligence and neuroscience, 2022(1), 3406228.
7. Nair, R., Zafrullah, S. N., Vinayasree, P., Singh, P., Zahra, M. M. A., Sharma, T., & Ahmadi, F. (2022). Blockchain-Based Decentralized Cloud Solutions for Data Transfer. Computational Intelligence and Neuroscience, 2022(1), 8209854.
8. Fazal, R., Shah, M. A., Khattak, H. A., Rauf, H. T., & Al-Turjman, F. (2022). Achieving data privacy for decision support systems in times of massive data sharing. Cluster Computing, 25(5), 3037-3049.
9. Sauber, A. M., El-Kafrawy, P. M., Shawish, A. F., Amin, M. A., & Hagag, I. M. (2021). A new secure model for data protection over cloud computing. Computational Intelligence and Neuroscience, 2021(1), 8113253.
10. Aldabbagh, G., Alghazzawi, D. M., Hasan, S. H., Alhaddad, M., Malibari, A., & Cheng, L. (2021). Secure Data Exchange in M-Learning Platform using Adaptive Tunicate Slime-Mold-Based Hybrid Optimal Elliptic Curve Cryptography. Applied Sciences, 11(12), 5316.
11. Alouffi, B., Hasnain, M., Alharbi, A., Alosaimi, W., Alyami, H., & Ayaz, M. (2021). A systematic literature review on cloud computing security: threats and mitigation strategies. Ieee Access, 9, 57792-57807.
12. Huang, Q., Yue, W., Yang, Y., & Chen, L. (2021). P2GT: Fine-grained genomic data access control with privacy-preserving testing in cloud computing. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 19(4), 2385-2398.