Cureus Journal of Computer Science

Open Access Original Article

Hybrid Encryption Models for Optimal Balance of Security, Scalability, and Computational Efficiency in Cloud Computing

Received 04/29/2025
Review began 05/26/2025
Review ended 08/14/2025
Published 08/18/2025

© Copyright 2025
Ahialey et al. This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

DOI:
https://doi.org/10.7759/s44389-025-05146-3

Edem Ahialey [1] , Regina E. Turkson [1] , Abdul-Lateef Yussif [1] , Enoch A. Frimpong [2] , Elliot Attipoe [1] , Emmanuel D. Tetteh [1]

1. Department of Computer Science and Information Technology, University of Cape Coast, Cape Coast, GHA 2. Department of Computer Science and Technology, Cape Coast Technical University, Cape Coast, GHA

Corresponding author: Edem Ahialey, edem.ahialey@stu.ucc.edu.gh

Abstract

As cloud computing continues to evolve and manage increasingly large and diverse datasets, achieving a balance between security, scalability, and computational efficiency remains a critical challenge. This study presents a comparative evaluation of four hybrid encryption models, Rivest-Shamir-Adleman-Advanced Encryption Standard (RSA-AES), Elliptic Curve Cryptography-Advanced Encryption Standard (ECC-AES), RSA-Blowfish, and ECC-Blowfish, designed to integrate the strengths of symmetric and asymmetric cryptography. Each model is implemented using robust configurations, including AES-256-Galois/Counter Mode, Blowfish-EAX, RSA-3072, ECC Curve25519, and SHA3-256 hashing. Experimental analysis was conducted using multimodal datasets comprising text, image, audio, and tabular files with sizes ranging from 10 MB to over 1 GB to reflect real-world cloud storage conditions. Performance metrics such as encryption/decryption time, throughput, and cryptographic randomness were measured. Statistical significance testing and the NIST Statistical Test Suite were applied to evaluate consistency and cipher strength. Results indicate that ECC-AES achieved the highest throughput and most stable randomness characteristics across file types, while RSA-AES showed increased cryptographic robustness with larger data sizes. Although Blowfish-based hybrids ensured strong randomness, their slower performance makes them less suited for high-throughput cloud environments. The findings support ECC-AES as the optimal hybrid encryption model for cloud computing, offering a strong trade-off between security and operational efficiency.

Categories: Performance Tuning and Optimization, Cloud Security, Cryptographic Algorithms
Keywords: cloud computing, hybrid encryption, elliptic curve cryptography, rivest–shamir–adleman, advanced encryption standard, blowfish

Introduction

Today's digital landscape has made cloud computing integral to business operations, offering scalable and flexible IT solutions while reducing the need for traditional infrastructure [1]. By 2025, the cloud is expected to store 200 zettabytes of data, with 60% of corporate data already in the cloud [2-3]. While cloud services provide significant advantages through models such as Infrastructure as a Service, Platform as a Service, and Software as a Service [4-5], they also introduce critical security concerns. In 2022 alone, 45% of reported security breaches occurred in cloud environments [6]. As a result, ensuring data confidentiality, integrity, and availability is vital for maintaining trust and regulatory compliance [7].

Cloud security challenges are multifaceted, involving weak authentication mechanisms, insecure APIs, data loss, denial-of-service attacks, and advanced persistent threats [8]. Misconfigurations and vulnerabilities in shared cloud infrastructures further exacerbate risks. To address these challenges, researchers have emphasized cryptography, particularly encryption as a foundational approach for securing cloud data [9]. Encryption protects sensitive information from unauthorized access, ensuring confidentiality, integrity, authentication, and non-repudiation [10]. Symmetric encryption algorithms like AES and Blowfish are efficient for securing large datasets, though they rely on secure key distribution, which remains a significant challenge [11-12]. Asymmetric encryption schemes, such as Rivest-Shamir-Adleman (RSA) and Elliptic Curve Cryptography (ECC), offer secure key exchange through public and private key mechanisms, albeit with slower performance [13-14]. To leverage the strengths of both paradigms, hybrid encryption techniques have emerged as effective solutions. These techniques combine the efficiency of symmetric algorithms with the secure key management of asymmetric cryptography, making them ideal for cloud environments.

Symmetric encryption, relying on a single shared key, remains efficient for processing large data volumes. Blowfish, supports 64-bit blocks and variable key lengths from 32 to 448 bits [15]. Advanced Encryption Standard (AES), standardized by NIST in 2001, offers 128-bits, 192-bits, and 256-bits key options with 10, 12, and 14 rounds, respectively, and is widely adopted for its security and speed [16]. In contrast,

asymmetric encryption uses a public/private key pair to facilitate secure communication. RSA is based on the computational difficulty of factoring large composite numbers, while ECC achieves strong security with smaller keys. Curve25519, developed by Bernstein, is notable for side-channel resistance and cross-platform optimization [17-18].

This study employs SHA3-256 for all models, drawing from the Keccak-based SHA-3 family, known for its flexibility and hardware efficiency [19]. Key performance indicators such as encryption/decryption time and throughput are critical in real-time, IoT, and cloud-based systems, influencing efficiency, cost, and compliance [20-21].

To evaluate cryptographic robustness, this research applies the NIST Statistical Test Suite, which consists of 15 tests for randomness and unpredictability [22]. It has been used in diverse applications, including dynamic key generation in 5G [23], lightweight cipher evaluation for Internet of Healthcare Things [24], and pseudo-random number analysis [22].

Numerous studies have explored hybrid cryptographic schemes for enhancing cloud security. Akter et al. [25] proposed a hybrid RSA-AES model that integrates AES-128 for fast data encryption and RSA for secure key distribution, supplemented with Hash-Based Message Authentication Code for data integrity. Rehman et al. [26] demonstrated the efficiency of an ECC-AES hybrid model in secure data sharing. Muthu Meenakshi and Sabeen [27] analyzed the AES, Blowfish, and Twofish algorithms, focusing on their strengths, weaknesses, and applications in cloud environments. Muthu Meenakshi and Sabeen [27] emphasized the importance of hybrid approaches in addressing modern data security challenges. Iavich et al. [28] compared the Blowfish, Twofish, and RSA cryptosystems to evaluate their performance and security features. They implemented a hybrid cryptographic model combining these algorithms to enhance data protection. The study analyzed the strengths and weaknesses of each algorithm, focusing on encryption speed, key length, and resistance to attacks. Their findings highlighted the advantages of hybrid cryptosystems in achieving a balance between efficiency and security. With the rise of quantum computing, however, traditional cryptographic methods face potential vulnerabilities. In response, Baseri et al. [14] explored post-quantum cryptography and hybrid algorithms that integrate symmetric and asymmetric methods to mitigate these emerging threats.

Building upon previous research, this study explores and evaluates four hybrid encryption models designed to enhance both security and computational efficiency in cloud environments. By systematically assessing their performance in terms of encryption and decryption speed, throughput, and cipher strength, the research aims to demonstrate how hybrid cryptographic approaches can effectively address modern cloud security challenges while maintaining practical efficiency.

Materials And Methods

This paper discusses the development and evaluation of multiple hybrid encryption models to assess their performance across different use cases. The primary objective is to integrate asymmetric and symmetric encryption techniques to improve both security and computational efficiency. The study focuses on four widely used encryption algorithms in industry: ECC, RSA, AES, and Blowfish. By combining these algorithms, four hybrid encryption schemes are formed: RSA-AES, ECC-AES, RSA-Blowfish, and ECC-Blowfish. Each hybrid model utilizes a symmetric algorithm to encrypt the data, while an asymmetric algorithm is employed for secure key exchange over potentially unsafe networks. This study evaluates the four hybrid encryption models, Hybrid AES-ECC, Hybrid Blowfish-ECC, RSA-AES Hybrid, and Hybrid RSA-Blowfish, with the goal of optimizing both security and computational performance in cloud environments.

Using robust configurations such as AES-256 in Galois/Counter Mode (GCM), Blowfish in EAX mode, RSA with 3,072-bit keys, ECC Curve25519, and SHA3-256 hashing, the evaluation focuses on encryption and decryption times, throughput, and cipher randomness, the latter assessed using the NIST statistical test suite. The dataset used for this study consisted of multimodal folders of sizes 251.29MB, 503.61MB, 751.04MB, and 1,003.99MB, downloaded from Online Test Case Generator [29]. Each folder contained WAV audio files, CSV files, text files, and JPEG images distributed equally. These file types reflect common cloud workloads such as data analytics (CSV), document storage (TXT), media content (WAV, JPEG), and web data. These folders were used to assess encryption time, decryption time, and throughput metrics. The 251MB folder was specifically selected to perform the NIST Statistical Test to evaluate randomness and security robustness.

Figure *1* illustrates the design of the hybrid model that incorporates the ECC Curve25519 curve, while Figure *2* outlines the design of the hybrid model utilizing RSA with 3,072-bit keys for the key exchange process; both diagrams were originally created by the authors to represent the proposed architectures and workflows. In this approach, the sender generates a SHA3-256 hash of the input data and sends it to the recipient. Once the recipient receives the message, they decrypt it and calculate a hash of the decrypted message. The recipient then verifies the message's integrity by comparing their computed hash with the one provided by the sender.

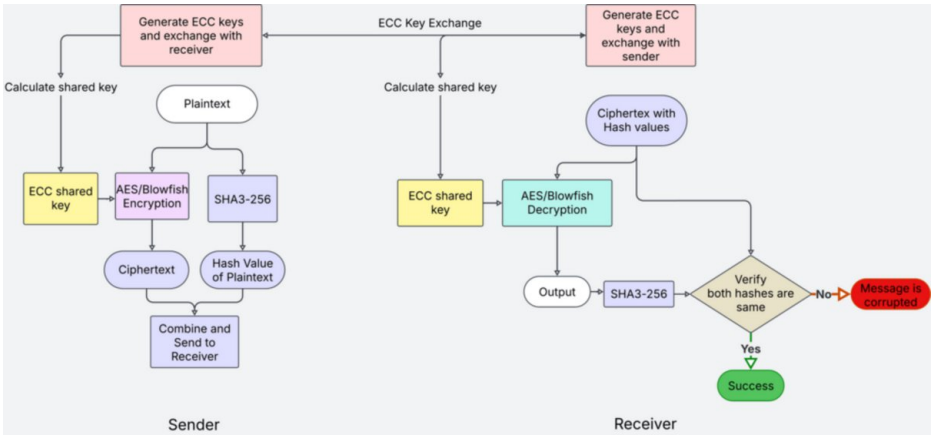

FIGURE 1: ECC Combined With AES/Blowfish

AES, Advanced Encryption Standard; ECC, Elliptic Curve Cryptography

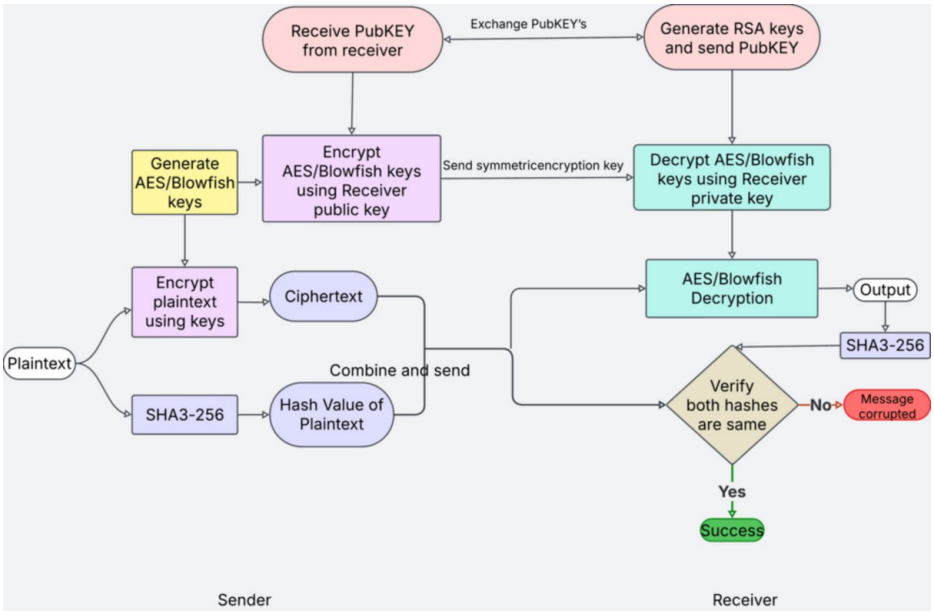

FIGURE 2: RSA Combined With AES/Blowfish

AES, Advanced Encryption Standard; RSA, Rivest-Shamir-Adleman

The encryption setups used in this study are carefully chosen for their robust security features: SHA3-256 for hashing, RSA with a 3,072-bit key for asymmetric encryption, ECC based on the Curve25519 for public key cryptography, AES-256 in GCM for authenticated encryption, and Blowfish configured with a 448-bit key in EAX mode for both encryption and integrity checks. Within the ECC-based approaches, the encryption process begins by generating a private-public key pair. A common secret is derived from these keys and converted into a 256-bit key, which is then used with both AES and Blowfish encryption algorithms. This approach leverages the strengths of both asymmetric and symmetric encryption to ensure secure and efficient data protection. Asymmetric encryption, using ephemeral ECC keys and 3,072-bit RSA key pairs, is employed for secure key management and to maintain confidentiality during transmission. Symmetric encryption is then used for high-performance data protection, ensuring both confidentiality and data integrity.

RSA-AES hybrid encryption

The RSA-AES model begins with an AES-256 key (K_{AES}) randomly generated and encrypted using RSA with a 3,072-bit public key (PK_{RSA}). The encryption process is as follows:

Encryption

$$C_K = \text{RSA Encrypt}(PK_{\text{RSA}},\ K_{\text{AES}}) \quad (1)$$

$$C_P = \text{AES-GCM Encrypt}(K_{\text{AES}}, P) \quad (2)$$

The final ciphertext consists of the encrypted AES key and the AES-encrypted message:

$$C = (C_K, C_P) \quad (3)$$

Decryption

The recipient uses their private RSA key (SK_{RSA}) to decrypt K_{AES}, which is then used to decrypt the message using AES-GCM:

$$K_{\text{AES}} = \text{RSA Decrypt}(SK_{\text{RSA}}, C_K) \quad (4)$$

$$P = \text{AES-GCM Decrypt}(K_{\text{AES}}, C_P) \quad (5)$$

ECC-AES hybrid encryption

The ECC-AES model replaces RSA with ECC for key exchange. A random AES-256 key (K_{AES}) is encrypted using ECC-based encryption (Curve25519) with the recipient's public key (P K_{ECC}). The encryption steps are as follows:

Encryption

$$C_K = \text{ECC Encrypt}(PK_{\text{ECC}}, K_{\text{AES}}) \quad (6)$$

$$C_P = \text{AES-GCM Encrypt}(K_{\text{AES}}, P) \quad (7)$$

The final ciphertext is as follows:

$$C = (C_K, C_P) \quad (8)$$

Decryption

Decryption involves recovering K_{AES} using the recipient's private ECC key (SK_{ECC}), followed by decryption of the ciphertext using AES-GCM:

$$K_{\text{AES}} = \text{ECC Decrypt}(SK_{\text{ECC}}, C_K) \quad (9)$$

$$P = \text{AES-GCM Decrypt}(K_{\text{AES}}, C_P) \quad (10)$$

RSA-Blowfish hybrid encryption

In the RSA-Blowfish model, the RSA algorithm is used to securely exchange a randomly generated Blowfish key (K_{BF}). Encryption proceeds as follows:

Encryption

$$C_K = \text{RSA Encrypt}(PK_{\text{RSA}}, K_{\text{BF}}) \quad (11)$$

$$C_P = \text{Blowfish-EAX Encrypt}(K_{\text{BF}}, P) \quad (12)$$

The final ciphertext is as follows:

$$C = (C_K, C_P) \quad (13)$$

Decryption

During decryption, the recipient extracts K_{BF} using their private RSA key and then decrypts the message using Blowfish-EAX:

$$K_{\mathrm{BF}} = \mathrm{RSA\ Decrypt}(SK_{\mathrm{RSA}}, C_K) \quad (14)$$

$$P = \mathrm{Blowfish\text{-}EAX\ Decrypt}(K_{\mathrm{BF}}, C_P) \quad (15)$$

ECC-Blowfish hybrid encryption

The ECC-Blowfish model employs ECC-based encryption for secure key exchange while using Blowfish in EAX mode for data encryption. The encryption steps are as follows:

Encryption

$$C_K = \mathrm{ECC\ Encrypt}(PK_{\mathrm{ECC}},\ K_{BF}) \quad (16)$$

$$C_P = \mathrm{Blowfish\text{-}EAX\ Encrypt}(K_{\mathrm{BF}}, P) \quad (17)$$

The final ciphertext is as follows:

$$C = (C_K, C_P) \quad (18)$$

Decryption

Decryption follows the same pattern, where K_{BF} is decrypted using the recipient's private ECC key, followed by Blowfish decryption:

$$K_{\mathrm{BF}} = \mathrm{ECC\ Decrypt}(SK_{\mathrm{ECC}}, C_K) \quad (19)$$

$$P = \mathrm{Blowfish\text{-}EAX\ Decrypt}(K_{\mathrm{BF}}, C_P) \quad (20)$$

Each of these hybrid encryption models ensures strong security by combining the strengths of asymmetric encryption for key distribution and symmetric encryption for data confidentiality and integrity.

The hybrid encryption models are applied and assessed using various performance metrics such as encryption time, decryption time, encryption throughput, and decryption throughput. To meaningfully validate the cryptographic strength and randomness of each cipher, the outputs were converted into binary strings and analyzed using the NIST Statistical Test Suite. This suite includes 15 primary tests, with a few having different parameter settings or internal configurations, resulting in a total of 45 distinct NIST tests per sample. The evaluation was conducted on a diverse dataset comprising multiple independent samples with varying data characteristics. Specifically, four different file types, CSV, TXT, WAV, and JPEG were used, and for each file type, four independent samples were generated for three different file sizes: 10 MB, 20 MB, and 50 MB. This setup resulted in a total of 48 test files, which were used to assess the four proposed hybrid encryption models. For each unique combination of model, file type, and file size, all matching samples were selected, and the total number of tests run was calculated as the number of samples multiplied by 45. The number of passed tests was determined by counting the total number of "Random" outcomes across the 45 NIST tests. The overall pass rate was then computed using the formula:

Pass Rate (%) = (Passed Tests/Total Tests) × 100.

This comprehensive approach ensures that the strengths of both asymmetric and symmetric encryption methods are effectively utilized, providing a thorough evaluation of their efficiency and security in practical scenarios.

Results

The implementation was carried out using Python 3.11.4 on a system equipped with an Intel® Core™ i7-6600U CPU running at 2.60GHz, 8.00GB of RAM, and a 64-bit OS. Files ranging from 251.29 MB to 1003.99 MB were generated for the experiment. Each file was processed through 100 iterations for each encryption model, with average results calculated. To enhance accuracy, the comparison parameters were averaged over 100 runs, providing a more dependable evaluation of the encryption models' performance and attributes.

The performance analysis of the hybrid encryption models shows clear variations in their efficiency. The RSA-AES model demonstrates exceptional performance, with encryption times varying from 0.7052 to 3.4216 seconds and decryption times ranging from 0.7084 to 2.8129 seconds. Its throughput is notably high, ranging from 293.43 to 356.33 MB/s for encryption and from 308.48 to 356.93 MB/s for decryption. This model exhibits consistent efficiency across different file sizes, as shown in Table *1*. Across 100 iterations, the RSA-AES model exhibited low variability in performance. Standard deviation for encryption

time ranged from 0.03 to 0.12 seconds, and decryption time deviation ranged from 0.02 to 0.10 seconds. The 95% confidence intervals for throughput values remained within ±3.8 MB/s, indicating consistent processing speed across all tested file sizes. Minor fluctuations in throughput values are expected and can be attributed to overhead costs, system-level caching effects, and algorithm-specific processing behavior [30].

File Size (MB)	251.29	503.61	751.04	1,003.99
Avg. Encryption Time (s)	0.7052	1.7151	2.3526	3.4216
Avg. Decryption Time (s)	0.7084	1.6326	2.2315	2.8129
Encryption Throughput (MB/s)	356.33	293.63	319.23	293.43
Decryption Throughput (MB/s)	354.72	308.48	336.56	356.93

TABLE 1: Output of RSA-AES Model

AES, Advanced Encryption Standard; RSA, Rivest-Shamir-Adleman

Table 2 presents the performance metrics of the RSA-Blowfish model, which exhibits significantly slower performance. Encryption times range from 9.1488 to 27.0514 seconds, while decryption times fall between 9.2106 and 24.4119 seconds. Throughput is modest and consistent, ranging from 27.47 to 49.07 MB/s (encryption) and 27.28 to 49.17 MB/s (decryption). Performance variation was more pronounced in the RSA-Blowfish model. Encryption and decryption time standard deviations ranged from 0.08 to 0.26 seconds, reflecting the model's increased computational overhead. Throughput values had wider 95% confidence intervals of up to ±5.2 MB/s, indicating relatively less consistency under repeated runs. As seen in Table 2, fluctuations in throughput across file sizes are present and can be attributed to system-level factors such as memory access patterns, overhead initialization, and mode-specific behavior, all of which can impact performance scaling in cryptographic operations [30].

File Size (MB)	251.29	503.61	751.04	1,003.99
Avg. Encryption Time (s)	9.1488	10.5589	15.3041	27.0514
Avg. Decryption Time (s)	9.2106	10.5046	15.2745	24.4119
Encryption Throughput (MB/s)	27.47	47.7	49.07	37.11
Decryption Throughput (MB/s)	27.28	47.94	49.17	41.13

TABLE 2: Output of RSA-Blowfish Model

RSA, Rivest-Shamir-Adleman

Table 3 presents the performance metrics of the ECC-AES model, which performs competitively, with encryption times ranging from 0.4912 to 2.6240 seconds and decryption times from 0.4970 to 2.5759 seconds. Throughput ranges from 382.62 to 511.53 MB/s (encryption) and 389.76 to 505.59 MB/s (decryption). The ECC-AES model demonstrated highly stable behavior. Encryption and decryption time standard deviations were minimal, ranging from 0.02 to 0.09 seconds. The throughput metrics showed strong consistency, with 95% confidence intervals within ±3.2 MB/s across all file sizes, confirming its reliability for high-performance encryption. Minor variations in throughput values are observable across file sizes, which are normal and can be influenced by factors such as initialization costs, CPU caching efficiency, and data alignment during block processing [30].

File Size (MB)	251.29	503.61	751.04	1,003.99
Avg. Encryption Time (s)	0.4912	1.1953	1.7509	2.624
Avg. Decryption Time (s)	0.497	1.1753	1.7145	2.5759
Encryption Throughput (MB/s)	511.53	421.33	428.95	382.62
Decryption Throughput (MB/s)	505.59	428.49	438.06	389.76

TABLE 3: Output of ECC-AES Model

AES, Advanced Encryption Standard; ECC, Elliptic Curve Cryptography

Table *4* presents the performance metrics of the ECC-Blowfish model, which exhibited comparatively slower performance. Encryption times ranged from 4.9346 to 22.3807 seconds, while decryption times varied from 4.9530 to 20.8663 seconds. Throughput values ranged from 44.86 to 50.97 MB/s for encryption and 48.12 to 50.85 MB/s for decryption. The model displayed moderate variability, with encryption and decryption time standard deviations between 0.05 and 0.18 seconds. The 95% confidence intervals for throughput remained within ±4.0 MB/s, indicating reasonable, though less consistent, performance compared to the ECC-AES and RSA-AES models. The observed fluctuations in throughput across file sizes are consistent with known effects of cryptographic overhead, memory access patterns, and algorithm-specific execution characteristics [30]. Overall, ECC-AES and RSA-AES demonstrated superior stability across runs and input sizes, highlighting their suitability for scalable, time-sensitive encryption in cloud environments.

File Size (MB)	251.29	503.61	751.04	1,003.99
Avg. Encryption Time (s)	4.9346	9.88	15.0379	22.3807
Avg. Decryption Time (s)	4.953	9.9039	15.1398	20.8663
Encryption Throughput (MB/s)	50.92	50.97	49.94	44.86
Decryption Throughput (MB/s)	50.73	50.85	49.61	48.12

TABLE 4: Output of ECC-Blowfish Model

ECC, Elliptic Curve Cryptography

The NIST pass rates for CSV file encryption using the four proposed hybrid models are summarized in Table *5*. Across all file sizes (10 MB, 20 MB, and 50 MB), the ECC-AES model consistently achieved a high pass rate of 90.24%, indicating strong and stable randomness characteristics. The ECC-Blowfish model also performed consistently, albeit slightly lower, with a uniform pass rate of 89.63% across all sizes. The RSA-AES model showed an improvement with increasing file size, rising from 87.80% at 10 MB to 90.24% at both 20 MB and 50 MB. In contrast, the RSA-Blowfish model showed a slight fluctuation, achieving 90.24% at 10 MB, 89.02% at 20 MB, and 89.63% at 50 MB. Overall, all models demonstrated strong randomness properties, with ECC-based combinations generally showing more consistency across varying file sizes.

Model	10 MB	20 MB	50 MB
ECC-AES	90.24%	90.24%	90.24%
ECC-Blowfish	89.63%	89.63%	89.63%
RSA-AES	87.80%	90.24%	90.24%
RSA-Blowfish	90.24%	89.02%	89.63%

TABLE 5: NIST Pass Rates for CSV Files

NIST, National Institute of Standards and Technology; AES, Advanced Encryption Standard; ECC, Elliptic Curve Cryptography; RSA, Rivest-Shamir-Adleman

As shown in Table 6, the NIST pass rates for JPEG file encryption reveal slight variations in performance across the different hybrid models and file sizes. The ECC-AES model maintained high randomness at 10 MB and 50 MB with 90.24% pass rates but experienced a notable drop to 82.32% at 20 MB. The ECC-Blowfish model remained relatively stable, with pass rates ranging from 88.41% to 90.24%, indicating consistent performance. The RSA-AES model exhibited a slight decline at 20 MB (87.20%) and 50 MB (89.63%) compared to the perfect score at 10 MB. Similarly, the RSA-Blowfish model demonstrated high performance at smaller sizes but showed a decrease to 85.98% at 50 MB. These results suggest that while all models perform well overall, certain combinations may be more sensitive to file size changes when processing JPEG data.

Model	10 MB	20 MB	50 MB
ECC-AES	90.24%	82.32%	90.24%
ECC-Blowfish	90.24%	90.24%	88.41%
RSA-AES	90.24%	87.20%	89.63%
RSA-Blowfish	90.24%	89.63%	85.98%

TABLE 6: NIST Pass Rates for JPEG Files

NIST, National Institute of Standards and Technology; AES, Advanced Encryption Standard; ECC, Elliptic Curve Cryptography; RSA, Rivest-Shamir-Adleman

The NIST pass rates for TXT file encryption are presented in Table 7. The ECC-AES model demonstrated a slight decline in performance as file size increased, starting at 90.24% for 10 MB and decreasing to 87.80% at 50 MB. A similar trend was observed for ECC-Blowfish, which began at 90.24% and dipped slightly to 88.41% at 50 MB. In contrast, the RSA-AES model showed improvement with larger file sizes, rising from 87.20% at 10 MB to 90.24% at both 20 MB and 50 MB, indicating better randomness with increased data volume. The RSA-Blowfish model showed some inconsistency, with the lowest pass rate of 86.59% at 20 MB, while performing better at 10 MB (89.02%) and 50 MB (89.63%). These results indicate that while ECC-based models show strong and stable performance, RSA-AES particularly benefits from larger input sizes in the context of TXT data.

Model	10 MB	20 MB	50 MB
ECC-AES	90.24%	89.63%	87.80%
ECC-Blowfish	90.24%	89.02%	88.41%
RSA-AES	87.20%	90.24%	90.24%
RSA-Blowfish	89.02%	86.59%	89.63%

TABLE 7: NIST Pass Rates for TXT Files

NIST, National Institute of Standards and Technology; AES, Advanced Encryption Standard; ECC, Elliptic Curve Cryptography; RSA, Rivest-Shamir-Adleman

The NIST pass rates for WAV file encryption are detailed in Table *8*. The ECC-AES model showed strong performance overall, peaking at 90.24% for the 20 MB file size, with slight variations at 10 MB (87.80%) and 50 MB (88.41%). The ECC-Blowfish model maintained consistently high scores across all sizes, ranging narrowly from 89.02% to 90.24%, suggesting stable randomness for audio data. The RSA-AES model performed best at the smallest file size (90.24% at 10 MB), followed by slight reductions at 20 MB (89.63%) and a more noticeable drop at 50 MB (87.20%). Meanwhile, RSA-Blowfish exhibited solid and balanced performance across all sizes, achieving its highest score (90.24%) at 20 MB. Overall, the results indicate that all models maintained acceptable levels of randomness, with ECC-Blowfish showing the most consistent behavior across different WAV file sizes.

Model	10 MB	20 MB	50 MB
ECC-AES	87.80%	90.24%	88.41%
ECC-Blowfish	90.24%	89.63%	89.02%
RSA-AES	90.24%	89.63%	87.20%
RSA-Blowfish	88.41%	90.24%	89.63%

TABLE 8: NIST Pass Rates for WAV Files

NIST, National Institute of Standards and Technology; AES, Advanced Encryption Standard; ECC, Elliptic Curve Cryptography; RSA, Rivest-Shamir-Adleman

Discussion

The comparative evaluation of the four proposed hybrid encryption models, RSA-AES, RSA-Blowfish, ECC-AES, and ECC-Blowfish, highlights notable trade-offs between performance efficiency and cryptographic strength across different file types and sizes.

From a performance standpoint, the ECC-AES model consistently emerged as the most efficient, achieving the fastest encryption and decryption times and the highest throughput across all file sizes (Table *3*). This can be attributed to the lightweight nature of ECC and the high speed of AES, which together enable rapid processing of large data blocks. RSA-AES also demonstrated excellent performance (Table *1*), albeit slightly slower than ECC-AES, particularly as file size increased. In contrast, the RSA-Blowfish and ECC-Blowfish models (Tables *2* and 4) exhibited significantly higher encryption and decryption times, with correspondingly lower throughput, indicating that Blowfish, though secure, is comparatively slower, especially when combined with RSA.

In terms of randomness and cryptographic strength, measured using the NIST Statistical Test Suite, all models generally demonstrated high pass rates, affirming their ability to generate secure ciphertext with strong pseudorandom characteristics. However, ECC-based models exhibited more consistent behavior across file types and sizes. For example, ECC-AES maintained stable pass rates near or equal to 90.24% in CSV and WAV files, and only slightly fluctuated in TXT and JPEG files (Tables *5*-*8*). ECC-Blowfish also showed commendable consistency, especially in handling WAV and JPEG formats, with pass rates typically above 88%. Interestingly, the RSA-AES model exhibited improved randomness performance as file size increased, particularly in the CSV and TXT datasets, suggesting that its robustness scales with larger data volumes. Meanwhile, RSA-Blowfish showed slightly more variation in pass rates, especially for JPEG and TXT files, indicating some sensitivity to both file type and size.

Overall, the results support the notion that hybrid cryptographic models leveraging ECC as the asymmetric component and AES as the symmetric cipher offer a balanced combination of speed and security. This makes ECC-AES especially suitable for applications requiring fast, lightweight, and secure data encryption, such as mobile communications, IoT environments, or cloud storage. RSA-AES also remains a viable alternative where ECC may not be supported, delivering both high throughput and increasing randomness with scale. While Blowfish-based models demonstrated acceptable randomness, their slower performance makes them less attractive for scenarios involving large or time-sensitive data. However, they may still be valuable in contexts where compatibility with legacy systems or particular hardware constraints necessitates their use.

Conclusions

This study explored four hybrid encryption models, RSA-AES, RSA-Blowfish, ECC-AES, and ECC-Blowfish, with the objective of identifying optimal trade-offs between security, scalability, and computational efficiency in cloud computing environments. By evaluating these models across different file types and sizes, and validating both their performance and randomness characteristics, a comprehensive picture of their real-world applicability was established. Among the models tested, ECC-AES consistently outperformed others in terms of speed and throughput while maintaining high NIST randomness pass rates, making it an ideal candidate for cloud applications where performance and robust security are critical. RSA-AES demonstrated comparable cryptographic strength and improved scalability with larger files, offering a viable alternative when ECC support is unavailable. While Blowfish-based models ensured strong randomness, their slower processing times make them less suitable for high-throughput or time-sensitive cloud operations. The findings affirm that hybrid encryption strategies, when carefully chosen, can significantly enhance both security and efficiency in cloud computing. Specifically, the ECC-AES combination stands out as a balanced solution for scalable and secure cloud data protection. This study was conducted in a controlled environment and, while the datasets used were sufficient to reveal performance scaling trends and comparative behavior, it did not involve deployment on distributed or real-time cloud systems. We acknowledge that evaluating performance in full-scale, distributed cloud environments is a critical next step for operational validation.

Future work will focus on large-scale implementation in production settings, including stress-testing under concurrent multi-user workloads, integration into real-time systems, and assessment of energy consumption impacts. Additionally, resilience against side-channel attacks will be explored to further refine applicability within complex cloud infrastructures. While ECC and RSA remain secure under classical threat models, future evaluations should also investigate post-quantum cryptographic alternatives to strengthen defenses against emerging quantum threats.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Edem Ahialey, Regina E. Turkson, Abdul-Lateef Yussif, Enoch A. Frimpong, Elliot Attipoe, Emmanuel D. Tetteh

Acquisition, analysis, or interpretation of data: Edem Ahialey, Enoch A. Frimpong

Drafting of the manuscript: Edem Ahialey, Regina E. Turkson

Critical review of the manuscript for important intellectual content: Regina E. Turkson, Abdul-Lateef Yussif, Enoch A. Frimpong, Elliot Attipoe, Emmanuel D. Tetteh

Supervision: Regina E. Turkson

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Acknowledgements

Edem Ahialey and Regina Esi Turkson contributed equally to the work and should be considered co-first authors.

References

1. Khan W, Ullah H, Ahmad A, et al.: CrashSafe: a formal model for proving crash-safety of Android applications. Human-centric Computing and Information Sciences. 2018, 8:21. 10.1186/s13673-018-0144-7
2. Michalowski M: 55 Cloud Computing Statistics for 2025. (2025). https://spacelift.io/blog/cloud-computing-statistics.
3. Abdulnabi H: Cloud computing strategies and approaches for better energy efficiency and environmental impact. Al-Iraqia Journal of Scientific Engineering Research. 2024, 3:48-59. 10.58564/ijser.3.3.2024.223
4. Younis R, Iqbal M, Munir K, Javed MA, Haris M, Alahmari S: A comprehensive analysis of cloud service models: IaaS, PaaS, and SaaS in the context of emerging technologies and trend. 2024 International Conference on Electrical, Communication and Computer Engineering (ICECCE), Kuala Lumpur, Malaysia. 2024, 1-6. 10.1109/ICECCE63537.2024.10823401
5. Jiang PHW, Wang WYC: Comparison of SaaS and IaaS in cloud ERP implementation: the lessons from the practitioners. VINE Journal of Information and Knowledge Management Systems. 2024, 54:683-701. 10.1108/VJIKMS-10-2021-0238
6. Kolesnikov N: 65 + Cloud Computing Statistics You Need to Know For August 2025 - The Where, Why & How Much. (2023). https://www.techopedia.com/cloud-computing-statistics.
7. Tabrizchi H, Kuchaki Rafsanjani M: A survey on security challenges in cloud computing: issues, threats, and solutions. The Journal of Supercomputing. 2020, 76:9493-9532. 10.1007/s11227-020-03213-1
8. Sharma A, Singh UK, Upreti K, Yadav DS: An investigation of security risk & taxonomy of Cloud Computing environment. 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India. 2021, 1056-1063. 10.1109/ICOSEC51865.2021.9591954
9. Zhang L, Xiong H, Huang Q, Li J, Choo KKR, Li J: Cryptographic solutions for cloud storage: Challenges and research opportunities. IEEE Transactions on Services Computing. 2022, 15:567-587. 10.1109/TSC.2019.2937764
10. Murad SH, Rahouma KH: Hybrid cryptography for cloud security: Methodologies and designs. Digital Transformation Technology. Magdi DA, Helmy YK, Mamdouh M, Joshi A (ed): Springer, Singapore; 2022. 224:129-140. 10.1007/978-981-16-2275-5_7
11. Qureshi MB, Qureshi MS, Tahir S, Anwar A, Hussain S, Uddin M, Chen C-L: Encryption techniques for smart systems data security offloaded to the cloud. Symmetry. 2022, 14:695. 10.3390/sym14040695
12. Halak B, Yilmaz Y, Shiu D: Comparative analysis of energy costs of asymmetric vs symmetric encryption-based security applications. IEEE Access. 2022, 10:76707-76719. 10.1109/ACCESS.2022.3192970
13. Kapoor J, Thakur D: Analysis of symmetric and asymmetric key algorithms. ICT Analysis and Applications. Fong S, Dey N, Joshi A (ed): Springer, Singapore; 2022. 314:133-143. 10.1007/978-981-16-5655-2_13
14. Baseri Y, Chouhan V, Ghorbani A: Cybersecurity in the quantum era: Assessing the impact of quantum computing on infrastructure. 10.48550/arXiv.2404.10659
15. Kumar MD, Sivanarayana GV, Indira DNVSLS, Raj MP: Skin cancer segmentation with the aid of multi-class dilated D-net (MD2N) framework. Multimedia Tools and Applications. 2023, 82:35995-36018. 10.1007/s11042-023-14605-9
16. Mouha N, Dworkin M: Review of the Advanced Encryption Standard. NIST Interagency/Internal Report (NISTIR) 8319. National Institute of Standards and Technology, Gaithersburg, MD; 2021. 10.6028/NIST.IR.8319
17. Gao Q, Sun K, Dong J, Zheng F, Lin J, Ren Y, Liu Z: V-Curve25519: Efficient implementation of Curve25519 on RISC-V architecture. Information Security and Cryptology. Ge C, Yung M (ed): Springer, Singapore; 2024. 14527:130-149. 10.1007/978-981-97-0945-8_8
18. Kleppmann M: Implementing Curve25519/X25519: A Tutorial on Elliptic Curve Cryptography. 1-34; 2022.
19. Chen Z, Pan M, He P, Qi W, Liu L, Shen L, You D: Context and auto-interaction are all you need: Towards context embedding based QoS prediction via automatic feature interaction for high quality cloud API delivery. Future Generation Computer Systems. 2022, 128:265-281. 10.1016/j.future.2021.10.014
20. Amalarethinam DIG, Rajakumari SEJ: An overview of data security algorithms in cloud computing. Applied Intelligence and Informatics. Mahmud M, Ben-Abdallah H, Kaiser MS, Ahmed MR, Zhong N (ed): Springer, Cham; 2023. 2065:355-367. 10.1007/978-3-031-68639-9_23
21. Hilal AG, Manaa ME: Securing real-time data streams with cloud-enabled lightweight cryptography. Innovations in Data Analytics. Bhattacharya A, Dutta S, Dutta P, Samanta D (ed): Springer, Singapore; 2024. 972:211-225. 10.1007/978-981-97-3466-5_16
22. Almaraz Luengo E, Román Villaizán J: Cryptographically secured pseudo-random number generators: Analysis and testing with NIST statistical test suite. Mathematics. 2023, 11:4812. 10.3390/math11234812
23. Sam Karthik S, Kavithamani A: OELF: short term load forecasting for an optimal electrical load forecasting using hybrid whale optimization based convolutional neural network. Journal of Ambient Intelligence and Humanized Computing. 2021, 14:7023-7031. 10.1007/s12652-021-03556-4
24. Ning L, Ali Y, Ke H, Nazir S, Huanli Z: A hybrid MCDM approach of selecting lightweight cryptographic cipher based on ISO and NIST lightweight cryptography security requirements for internet of health things. IEEE Access. 2020, 8:220165-220187. 10.1109/ACCESS.2020.3041327
25. Akter R, Khan MAR, Rahman F, Soheli SJ, Suha NJ: RSA and AES based hybrid encryption technique for enhancing data security in cloud computing. International Journal of Computational and Applied Mathematics & Computer Science. 2023, 3:60-71. 10.37394/232028.2023.3.8
26. Rehman S, Talat Bajwa N, Shah MA, Aseeri AO, Anjum A: Hybrid AES-ECC model for the security of data over cloud storage. Electronics. 2021, 10:2673. 10.3390/electronics10212673
27. Muthu Meenakshi G, Sabeen S: Securing cloud data: a comprehensive review of hybrid cryptography and analysis of AES, Blowfish, and Twofish algorithms. 2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India. 2024, 568-575. 10.1109/ICACRS62842.2024.10841570
28. Iavich M, Gnatyuk S, Jintcharadze E, Polishchuk Y, Fesenko A, Abisheva A: Comparison and hybrid

implementation of Blowfish, Twofish and RSA cryptosystems. 2019 IEEE 2nd Ukraine Conference on Electrical and Computer Engineering (UKRCON), Lviv, Ukraine. 2019, 970-974. 10.1109/UKRCON.2019.8880005

29. Sample Audio Files. (2025). Accessed: March 24, 2025: https://onlinetestcase.com/audio-file/.
30. Eromosele CC: Evaluating the impact of AES-256 encryption on network performance: An analysis of transfer time, latency and throughput. International Journal of Scientific Research and Modern Technology. 2025, 4:49-58. 10.5281/zenodo.14745473