IMPROVED CYBERSECURITY RESILIENCE MODEL FOR SENSITIVE DATA PROTECTION IN NIGERIAN TERTIARY INSTITUTIONS

U. F. ADAMU; O. SARJIYUS; & N. NACHANDIYA
Department of Computer Science, Adamawa State University, Mubi
Corresponding Author: aufaruk2018@gmail.com

DOI: **https://doi.org/10.70382/caijmasr.v8i9.028**

ABSTRACT

In the digital era, cybersecurity is a critical concern for organizations worldwide, especially Nigerian tertiary institutions managing sensitive data such as student records, financial details, and strategic plans. As Nigeria advances digitally, it faces growing threats, making traditional reactive security measures inadequate. Proactive, adaptive strategies are essential. This study proposes a tailored cybersecurity resilience model with a focus on hybrid encryption to enhance data protection. The research explores a hybrid approach combining Modified AES-256 (16 rounds) and RSA-2048 for secure key exchange. While symmetric encryption like AES (Advanced Encryption Standard) offers high performance and confidentiality, it lacks secure key distribution this is where asymmetric encryption like RSA (Rivest-Shamir-Adleman) excels. The hybrid model leverages both: RSA (Rivest-Shamir-Adleman) secures key exchange, while Modified Advanced Encryption Standard efficiently encrypts large data volumes, ensuring strong security and acceptable performance, particularly for external or inter-institutional data transfers. Experimental results using real-world datasets from Nigerian institutions showed that although Hybrid AES-RSA had higher latency and lower throughput due to RSA's computational demands, it provided the highest security level. It proved ideal for highly sensitive exchanges such as examination materials or financial reports over untrusted networks. The system resisted cryptanalytic attacks effectively, thanks to RSA's 2048-bit key strength. Despite slower processing during asymmetric operations, its support for integrity, non-repudiation, and authentication made it superior in high-security contexts. The study emphasizes the need for a layered cybersecurity framework encompassing technical controls, policy enforcement, and workforce training. Nigerian tertiary institutions should adopt hybrid encryption for external communications and use optimized symmetric encryption like Modified Advanced Encryption Standard internally. Integrating these strategies into a broader cybersecurity resilience model can help protect institutional data and contribute to national cybersecurity capacity building.

Keywords: Cybersecurity resilience, Hybrid encryption, Data protection, Encryption algorithm, Symmetric encryption, Asymmetric encryption, Throughput, Latency.

INTRODUCTION

In the digital era, cybersecurity is critical for organizational strategy and national security due to widespread ICT adoption across sectors (Tanriverdiyev, 2022). Digital transformation exposes vulnerabilities in institutions safeguarding sensitive data like student records, financial information, and strategic plans. Unauthorized access risks reputational damage, financial loss, and operational disruption, underscoring the urgency for robust cybersecurity frameworks in tertiary institutions.

Nigeria, advancing toward a digital economy, faces escalating cyber threats. The 2020 Global Cybersecurity Index ranked Nigeria 37th globally in cybersecurity readiness, categorized in Tier 3 among developing nations (ITU, 2020, 2024). Post-pandemic, cyber threats have evolved in sophistication, requiring proactive strategies beyond traditional reactive measures (Sharma & Zamfiroiu, 2023). The 2023 Nigerian general elections exemplified this challenge through ransomware, phishing, and insider attacks (Aladenusi & Odumuboni, 2024). Economic instability may exacerbate insider threats, while AI/ML advancements may empower ransomware actors.

Cybersecurity has become particularly pressing for higher educational institutions in resource-constrained regions. Umar et al. (2023) revealed serious vulnerabilities in Northeast Nigerian universities, including payment bypass, forged registry details, data tampering, and identity impersonation. Critical lack of formal cybersecurity protocols stems from limited awareness programs and insufficient staff training, increasing vulnerability and undermining academic credibility.

As institutions adopt digital technologies for teaching and data management, they face growing cybersecurity risks threatening sensitive information. Many struggle to allocate sufficient resources for risk management and policy development, compounded by inadequate security awareness (Kakembo, 2025). "The consequences of cyber breaches can severely damage institutional credibility and erode stakeholder trust, highlighting the importance of proactive approaches to risk mitigation" (Kakembo, 2025).

Despite increased awareness, many Nigerian tertiary institutions still struggle with data breaches due to weak encryption practices. Existing systems depend on outdated encryption methods that cannot handle growing data volumes. Online learning platforms and cloud storage have expanded entry points for cyberattacks, necessitating stronger encryption solutions.

While AES-256 is widely regarded as secure, it faces efficiency challenges with large files like research databases. Symmetric encryption alone lacks flexibility for secure communication between parties without pre-shared connections, showing the need for alternative approaches suited to educational institutions' evolving needs.

This research proposes an improved security resilience model addressing broader cybersecurity challenges rather than focusing solely on one institution. The study examines evolving threats, institutional vulnerabilities, and global best practices to develop context-specific strategies. Key recommendations include adopting zero-trust architectures, fostering industry-academia collaboration, and prioritizing workforce training. As cyber threats grow sophisticated, tertiary institutions must evolve from passive defenders to proactive guardians, ensuring resilience in an interconnected world.

This research aims to develop an improved cybersecurity resilience model protecting sensitive data in tertiary institutions. Specific objectives include:

i. Identifying current security threats, risks, and vulnerabilities
ii. Developing a comprehensive cybersecurity framework
iii. Implementing and comparing different cryptographic schemes for sensitive data protection
iv. Evaluating the model's effectiveness in detecting and preventing cyber threats.

LITERATURE REVIEW

Cybercrime has emerged as the second-largest man-made risk globally, encompassing unlawful activities such as hacking, fraud, and malware distribution (Soomro & Hussain, 2019). The financial impact of cybercrime is staggering, with global costs projected to reach $10.5 trillion annually by 2025 (Morgan, 2020). Universities are prime targets, with cyberattacks disrupting teaching, research, and administrative functions (Bukhari, 2018). The pandemic exacerbated vulnerabilities, as institutions transitioned to online learning, exposing weaknesses in remote infrastructure (Alawida *et al.,* 2022).

Existing studies highlight systemic gaps in Nigerian tertiary institutions, including inadequate policies, insufficient staff training, and outdated security protocols (Umar et al., 2023). Payment bypasses, record tampering, and grade manipulation underscore the risks of poor cybersecurity governance (Kakembo, 2025). These challenges demand comprehensive strategies that integrate advanced technologies, user behavior analysis, and regulatory compliance.

The Advanced Encryption Standard (AES) plays a pivotal role in securing sensitive data. Selected by NIST in 2000 for its security and performance, AES ensures confidentiality, integrity, and availability of digital information (Shakor et al., 2024; Brahmaiah *et al.,* 2023). In contrast, the RSA algorithm (Rivest-Shamir-Adleman) uses asymmetric encryption for secure key exchange but faces criticism for computational inefficiency, especially in resource-constrained settings (Annapoorna *et al.,* 2014; El-Dien et al., 2014).

Historical analysis reveals a transition from basic firewalls to AI-driven threat detection in institutional cybersecurity (Abrahams *et al.,* 2024). However, gaps persist in addressing human factors, policy alignment, and technological integration. Case studies emphasize balancing innovation with user behavior understanding and regulatory compliance (e.g., GDPR, PCI-DSS).

This study builds on prior work by proposing a modified AES algorithm optimized for speed and security in Nigerian institutions. By analyzing trends from pandemic-induced attacks to AI-powered risks it aims to inform frameworks that enhance data protection, operational continuity, and national cybersecurity capacity.

Encryption Algorithms and Hybrid Approaches for Sensitive Data Protection

Recent research has extensively examined various cryptographic algorithms and their performance characteristics for securing sensitive data. Adeniyi *et al.* (2022) illustrated secure sensitive data sharing using RSA and ElGamal cryptographic algorithms with hash functions, using text files ranging from 10 Kb to 100 Kb to examine both asymmetric algorithms. Their findings revealed that RSA achieves lower encryption time, while ElGamal achieves lower decryption time, highlighting the trade-offs between these algorithms for different applications.

Ogundoyin (2022) conducted a comparative analysis of cryptographic algorithms using symmetric and asymmetric algorithms AES, DES, and RSA with file sizes of 50b to 500b. The study found that RSA obtains lesser time in terms of both encryption and decryption, though the performance characteristics vary significantly with data size and algorithm type.

Adeniyi *et al*. (2023) implemented a block cipher algorithm for medical information security in cloud environments, comparing AES with a modified version. Their experimental results showed that modified AES outperforms conventional AES in encryption time (1293.837ms vs 1513.3ms), while conventional AES outperforms modified AES in decryption time (1289.627ms vs 1400.136ms). The avalanche effect analysis revealed that modified AES provides stronger security for small files, while conventional AES is more secure for larger files.

Sahin (2023) proposed a two-stage image encryption scheme using chaotic systems followed by AES and RSA encryption. The hybrid approach demonstrated high levels of security, speed, and reliability through statistical tests and comparative analysis, contributing valuable insights into chaos-based encryption frameworks.

Akter *et al.* (2023) described an RSA and AES-based hybrid encryption technique for cloud computing, finding that RSA decryption time increases dramatically with data size, while AES shows marginal increases. The hybrid encryption algorithm maintains stable decryption time aligned with AES performance, demonstrating significant gains in decryption efficiency for large-scale data. Experimental results showed the hybrid technique was 67.47% faster than RSA and only 32.39% slower than AES, making it suitable for cloud environments requiring both security and efficiency.

Si and Wang (2024) proposed a hybrid encryption system integrating improved AES with SHA-512, incorporating PKI and asymmetric algorithms like RSA for secure key management. The system employs multi-layered encryption using AES as the primary layer and DES or 3DES as the secondary layer, with SHA-512 for integrity verification. The model also integrates quantum-safe techniques, maintaining strong security while preserving high performance for cloud computing and healthcare applications.

Sarjiyus and Manga (2025) modified the standard RSA cryptography method by tampering with public key functionality to enable additional security keys, combining it with LSB steganography for enhanced protection.

Vinothkumar and Ram (2025) addressed traditional AES limitations by proposing Modified AES based on chaotic random key generation, eliminating pre-shared static keys through synchronized chaotic systems. Their model integrates Modified Digest hashing algorithm and blockchain technology to ensure data integrity, confidentiality, and transparency in e-medical systems.

These studies collectively demonstrate the evolution from single-algorithm approaches to sophisticated hybrid models that balance security, performance, and scalability. For Nigerian tertiary institutions handling sensitive student records, financial data, research information, and personal identifiable information, implementing such enhanced cybersecurity frameworks becomes critical for protecting institutional and stakeholder data in increasingly digital educational environments.

METHODOLOGY

SYSTEM ANALYSIS

a) Analysis of the Existing System

Federal Polytechnic, Bali is currently relying primarily on general security practices such as firewalls, antivirus, and password protection. However, this approach leaves the institution extremely exposed to cyber threats. A major gap is the lack of a threat management plan and regular vulnerability testing, making the institution susceptible to potential cyber threats.

In addition, the organization has an ineffective incident response plan, which hinders its capacity to act swiftly and successfully in case it suffers a cyberattack. The lack of proper employee training also aggravates the situation since the employees might not possess the capability to detect or react to security incidents as required. In the absence of round-the-clock system monitoring, impending threats could remain unnoticed for a long time.

Consequently, the prevailing cybersecurity approach is more reactive it only deals with incidents after their occurrence instead of being proactive in countering threats ahead of time when they have not yet emerged. Figure 1 below shows the diagram of the existing framework.

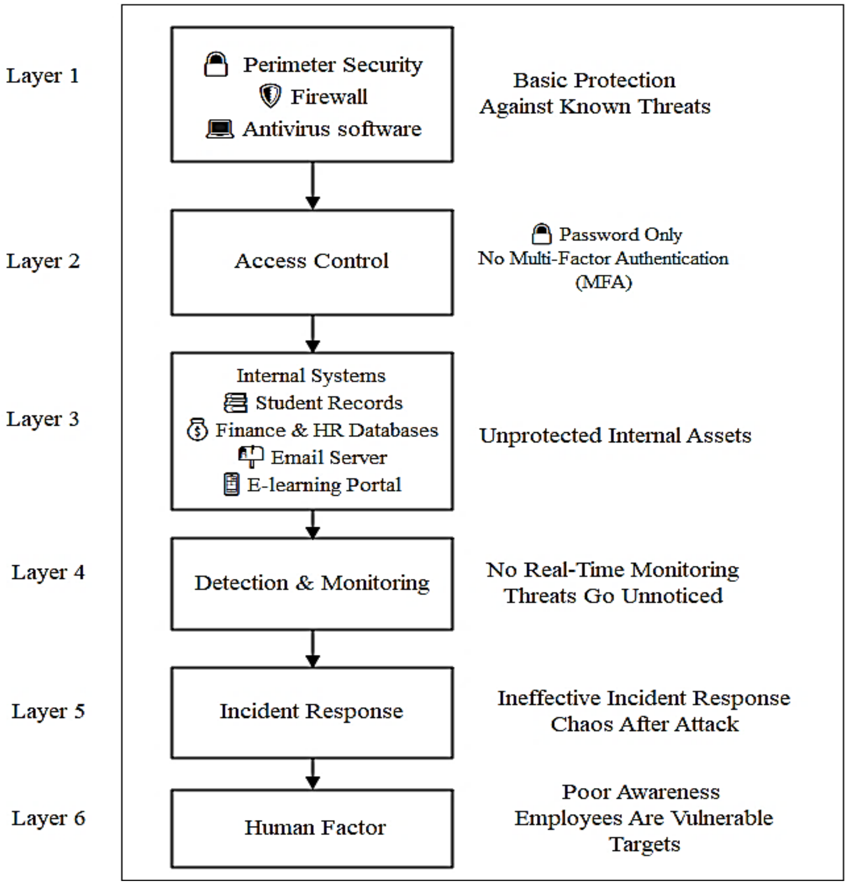


Figure 1: Framework of the Existing System

Figure 1 illustrates Federal Polytechnic, Bali's reactive cybersecurity infrastructure with significant vulnerabilities. Basic perimeter protection using firewalls and antivirus software proves insufficient against sophisticated attacks. Access control relies solely on simple passwords without multi-factor authentication, enabling easy system penetration through compromised credentials. Lack of network segmentation facilitates lateral movement, exposing critical assets like student data and financial databases.

The system lacks real-time monitoring and threat detection capabilities, with no SIEM solution or 24/7 SOC monitoring, allowing potential threats to remain undetected for extended periods. Incident response planning is ineffective, lacking structured approaches to handle cyberattacks, resulting in confusion and increased data loss risks. Human vulnerability remains critical, as employees lack cybersecurity awareness and training, making them susceptible to phishing and social engineering attacks. Overall, this reactive system leaves the institution highly vulnerable to diverse cyber threats due to weak technical provisions and inadequate employee preparedness, necessitating urgent cybersecurity infrastructure upgrades.

b) Analysis of the Proposed Cybersecurity System

The proposed cybersecurity solution offers a comprehensive and proactive approach that addresses weaknesses in technical controls, policies, and procedures within the current system. This framework is designed to enhance the security of Federal Polytechnic, Bali's sensitive data by protecting it from cyber threats and ensuring the confidentiality, integrity, and availability (CIA) of information.

Figure 2 below provides a visual representation of the proposed system framework.

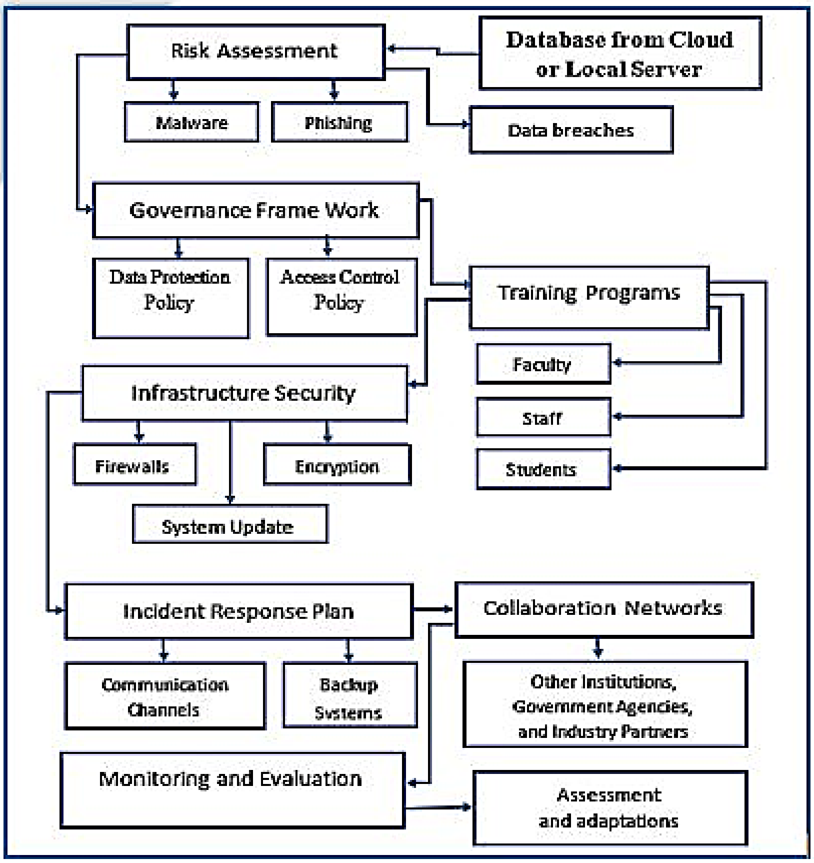


Figure 2: Proposed Framework of Improved Cyber Security System with resilience

Key Components of the Proposed System Framework

The proposed cybersecurity framework integrates technical controls (firewalls, intrusion detection systems, encryption), clear policies (data protection, incident response), and structured procedures (training, vulnerability assessments) to secure institutional data. It emphasizes role clarity, proactive threat mitigation, and compliance. A visual diagram (Figure 3) maps these components, fostering a security-aware culture through collaborative implementation and continuous improvement.

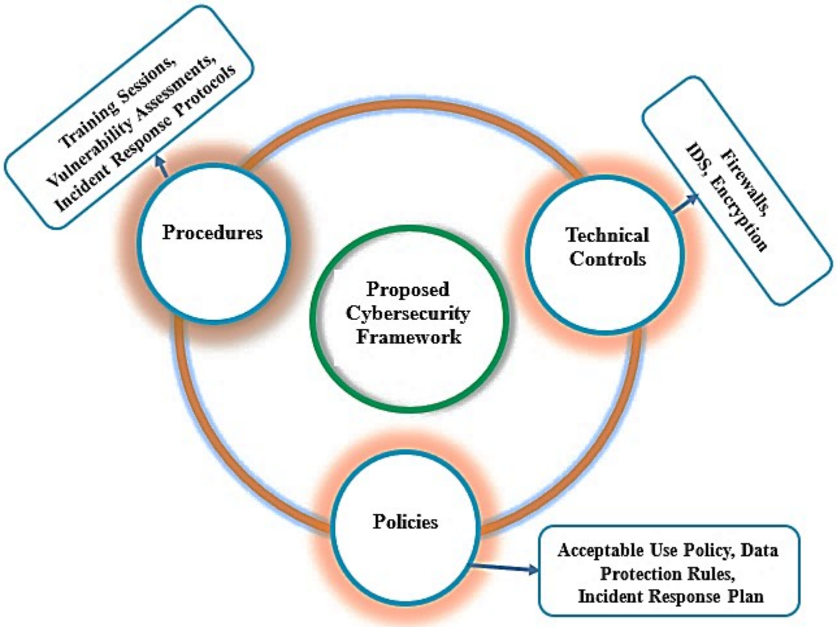


Figure 3: Key components of Proposed Cyber Security Framework

Method of Data Collection

To get a full picture of how data is managed and protected within the institution, we used a mixed-methods approach, combining numbers with personal insights. We started by sending out structured surveys to IT and administrative staff who work with data every day. This helped us gather clear, measurable information about how systems are used, how aware staff are of security practices, and how data handling works in daily operations. To go deeper, we also held conversations semi-structured interviews with key people like system administrators and cybersecurity team members. These discussions gave us a better understanding of the real-world challenges they face, the processes they follow, and where improvements could make a real difference. Existing documents such as ICT policies, network setup guides, and past incident reports were looked, to see how official practices align with day-to-day reality. By bringing together what we learned from surveys, interviews, and documents, we were able to cross-check our findings and build a more accurate, trustworthy picture of the current situation. All the data used in this study reflects real information handled by the institution such as financial records, staff and student details, research outputs, health-related data, and exam materials including questions and marking schemes. However, no personal or sensitive details were included in the analysis. Everything was anonymized to protect privacy and ensure confidentiality.

UML USE CASE DIAGRAM

The UML use case diagram figure 5, illustrates the cybersecurity workflow involving Administrators, Employees, Incident Responders, and Security Analysts. It highlights key activities such as managing risk, training employees, managing vulnerabilities, detecting incidents, responding to incidents, and monitoring security. The UML Use Case diagram emphasizes the use of AES/RSA encryption for secure communication and data protection, showing how each role contributes to maintaining a robust cybersecurity environment.

Cybersecurity System

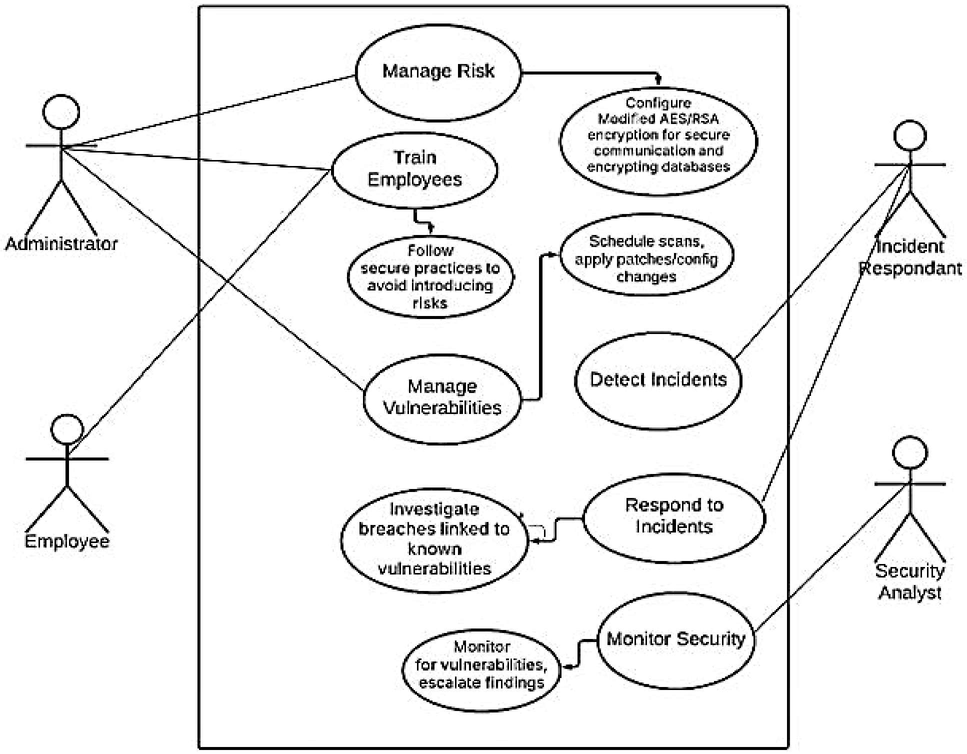

Manage Risk
Configure Modified AES/RSA encryption for secure communication and encrypting databases
Administrator
Train Employees
Follow secure practices to avoid introducing risks
Schedule scans, apply patches/config changes
Incident Respondant
Detect Incidents
Manage Vulnerabilities
Employee
Investigate breaches linked to known vulnerabilities
Respond to Incidents
Security Analyst
Monitor Security
Monitor for vulnerabilities, escalate findings

Figure 4: UML Use Case Diagram of the System

Figure 4 illustrates the cybersecurity framework's operation through coordinated roles: Administrators configure systems, conduct audits, and manage training; Incident Responders detect and respond to threats; Employees maintain security through training compliance; and Security Analysts provide continuous monitoring. These complementary roles create a solid cybersecurity posture protecting the institution's digital environment.

EXPERIMENTAL DESIGN

The research was conducted on a Dell Inspiron 15R work station featuring a 10th Gen Intel® Core™ i3-4010U CPU @ 1.70 GHz, 8 GB DDR4-3200 RAM, and a 512 GB NVMe SSD, with the operating system of Windows 10 Pro 64-bit installed on an x64-based processor. Experiments were run with Python 3.12, and dataset sizes in kilobytes (KB) were used to measure performance on three encryption schemes: standard AES-256 (14 rounds), Modified AES-256 (16 rounds), and a hybrid encryption system using Modified AES-256 and RSA (2048-bit keys) without altering AES's standard 128-bit block size. Each setup was thoroughly tested with a minimum of 10 runs, including outlier removal through ±2σ bounds, and median value verification for accuracy and reliability of results. Cryptographic operations conformed to Shannon's confusion and diffusion, using non-linear Sub Bytes transformations under Rijndael S-box for confusion and ShiftRows, MixColumns, and AddRoundKey for diffusion.

The round algorithm was automatically configured to use the Numbers 14 for standard AES and 16 for the variant so that it is simple to directly compare how the addition of additional rounds enhances security compared to performance in real-world use. Last but not least, MATLAB was used to generate visualizations of key performance metrics such as encryption time, decryption time, latency, and throughput so that it would be simple to discuss and analyze clearly.

IMPLEMENTATION

Pseudocode

This pseudocode is meant to help you understand how the encryption and decryption process works, showing you, the steps involved without tying it to any specific programming language, below is the pseudocode showing step-by-step implementation.

```
i.    Setup Data
        • Create test datasets (student, staff, financial records) and store as CSV files in a dictionary.
ii.   Define Encryption Methods
        • Implement three algorithms using 10 Iterations and Outlier Removal:
            a. Standard AES-256 (14 rounds, 256-bit key).
            b. Modified AES-256 (16 rounds, 256-bit key).
            c. Hybrid AES-RSA (AES-256 with RSA-2048 key exchange).
        • Each includes encryption/decryption functions.
iii.  Run Benchmarks
        • For each dataset and algorithm, measure:
            i.   Average Encryption Time.
            ii.  Average Decryption time.
            iii. Total time.
            iv.  Throughput
            • Store results for comparison.
iv.   Visualize Results
            • Generate performance charts per dataset:
            1. Total Encryption + Decryption Time
            2. Throughput
```

v. Main Program Execution

- Execute benchmarks, export results to **benchmark_results.csv**.
- Generate all charts and display completion confirmation.

MODIFIED AES ENCRYPTION PROCESS

The proposed algorithm showcases an optimized version of the Advanced Encryption Standard (AES) designed to prioritize speed without sacrificing security. Unlike the traditional AES-256 which uses 14 encryption cycles this variant reduces the number of rounds to 16, streamlining the process for faster data processing.

By trimming unnecessary computational steps, the algorithm achieves quicker encryption/decryption times while retaining its core security features, such as confusion and diffusion mechanisms. This balance ensures efficient performance for real-time applications without exposing vulnerabilities typically linked to reduced complexity. Figure 5 below is the Proposed Modified AES Encryption Process Diagram.

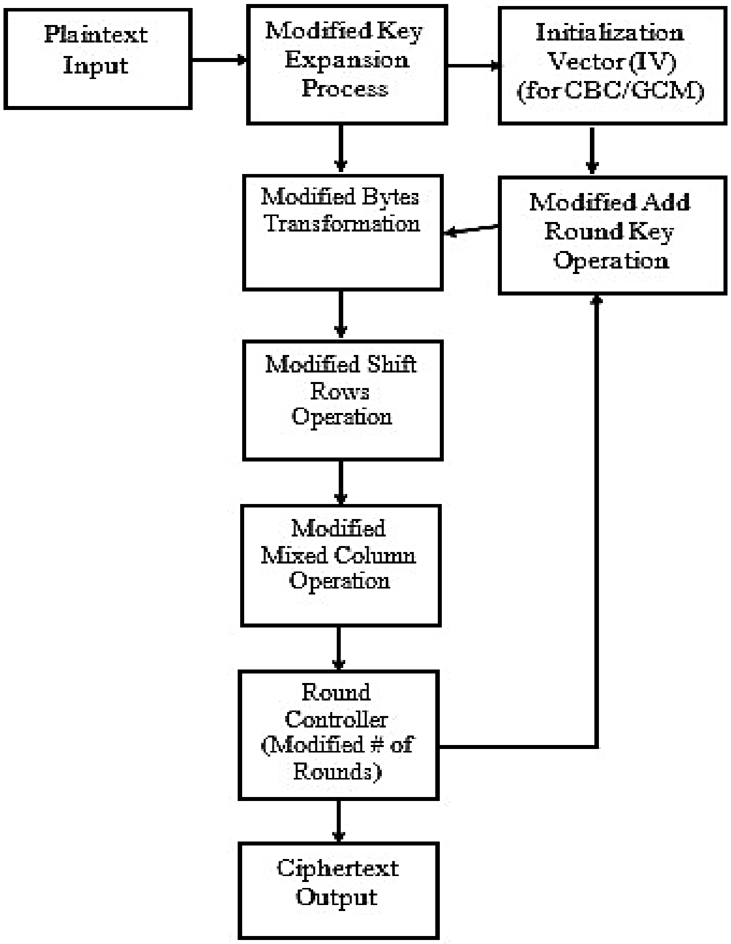

Plaintext Input
Modified Key Expansion Process
Initialization Vector (IV) (for CBC/GCM)
Modified Bytes Transformation
Modified Add Round Key Operation
Modified Shift Rows Operation
Modified Mixed Column Operation
Round Controller (Modified # of Rounds)
Ciphertext Output

Figure 5: Modified AES Encryption Process Diagram

Figure 5 details the modified AES encryption process in CBC and GCM modes. Plaintext is secured using expanded round keys and an initialization vector (IV), ensuring unique ciphertext in CBC (preventing repetition) and authentication tags in GCM. Encryption involves XOR operations, non-linear S-box substitution, row shifting, and column mixing across adjustable rounds (controlled by security needs). This structured diffusion enhances complexity, producing secure ciphertext. Optimized AES techniques improve resistance to attacks, offering robust confidentiality and integrity. The method is ideal for real-world applications requiring secure CBC/GCM implementations, balancing performance with advanced cryptographic safeguards for institutional data protection.

MODIFIED AES + RSA (HYBRID) ENCRYPTION PROCESS DIAGRAM

Figure 6 illustrates a hybrid encryption system that uses Modified AES-256 (symmetric) and RSA (asymmetric) to provide secure data transfer.

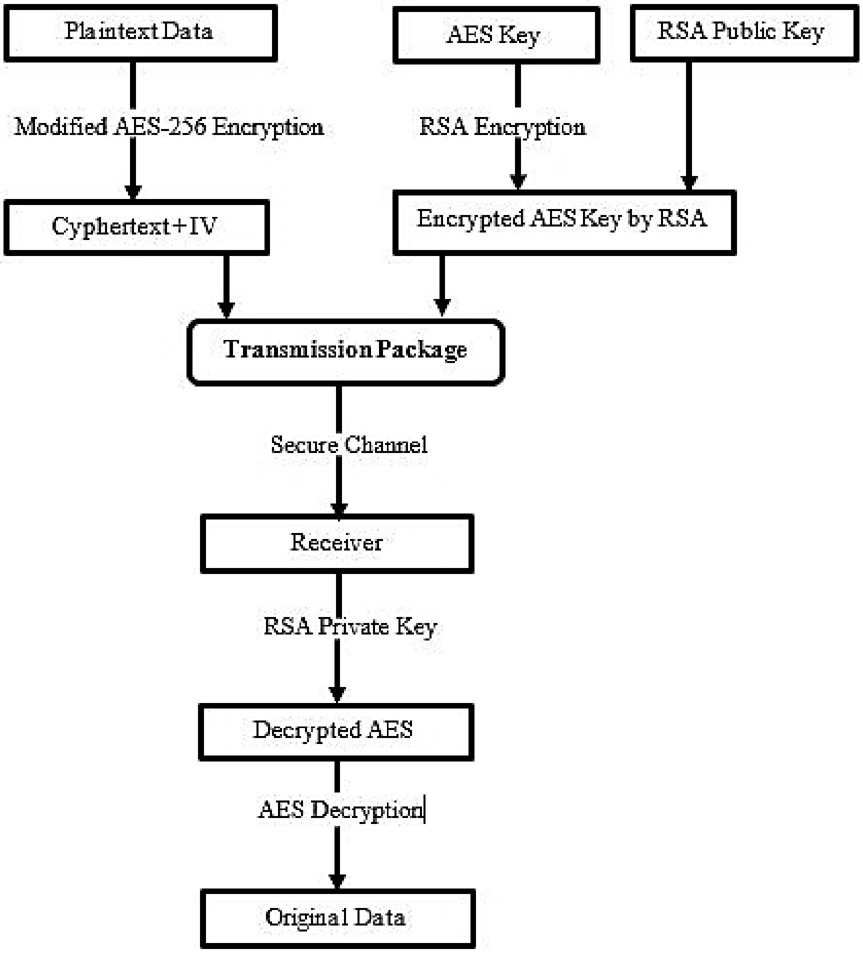


Figure 6: Modified AES + RSA Encryption Process Diagram

MATHEMATICAL EXPRESSION OF THE MODIFIED AES PROCESS

i. Key Expansion

AES-256 uses a 256-bit key to generate 14 subkeys (one per round). In the modified AES (16 rounds), the key expansion produces 16 subkeys:

Subkeys = $\{K_0, K_1, \ldots.. K_{15}\}$ .... (eqn. 1)

The key schedule algorithm remains unchanged, but the loop terminates after generating 16 subkeys instead of 14.

1. Loop Range:

i = 1 to 15 (16 rounds). .... (eqn. 2)

2. Final Round:

Uses **State** {**15**} and **K** {**16**}.

3. Key Expansion:

    - Generates 17 subkeys (**K_0** to **K** {**16**}) for 16 rounds.

Discussion on the rounds

- Rounds 1–15: Apply full AES transformations (MixColumns, ShiftRows, ByteSub) with subkeys **K_1** to **K** {**15**}.
- Final Round (16th): Skips MixColumns, applying only ByteSub, ShiftRows, and **K** {**16**}.

This maintains AES's core security mechanisms while extending the round count for enhanced resistance to cryptanalysis.

QUANTITATIVE PERFORMANCE EVALUATION METRICS

Quantitative Performance Evaluation refers to the systematic measurement and comparison of cryptographic algorithms using numerical metrics (e.g., time, speed, efficiency) to objectively assess their effectiveness.

a) Encryption/decryption latency and throughput

i. Latency: The total time taken to perform an operation. It indicates the delay or time required for a single encryption or decryption.

The latency for encryption and decryption can be computed as follows:

Encryption Latency=Init Time + Encryption Time .... (eqn. 3)

Decryption Latency=Init Time + Decryption Time .... (eqn. 4)

Throughput: The amount of data processed per unit time. It measures the efficiency of the encryption and decryption processes.

Throughput = File size (KB) / Encryption Time (seconds

$\text{Throughput}_{\text{encrypt}} = S_{\text{File size}} / T_{\text{encrypt}}$ .... (eqn. 5)

Throughput $_{decrypt} = S_{File\ size} / T_{dencrypt}$ .... (eqn. 6)

ii. Total Processing Time: Total Processing Time in encryption refers to the cumulative time required to complete all stages of the encryption and decryption process. It includes: Key Generation Time, Initialization Time, Encryption Time, and Decryption Time. This metric evaluates the end-to-end efficiency of an algorithm, balancing security and speed, the mathematical expression is:

Total Processing Time = $T_{keygen} + T_{init} + T_{encr} + T_{dec}$ .... (eqn. 7)

RESULTS

Performance Metrics

Performance metrics are quantitative measures like encryption time, decryption time, key generation time, and throughput that evaluate how well cryptographic algorithms perform under specific conditions. Experimental results comparing Standard AES, Modified AES, and Modified AES + RSA are summarized in Tables 1 to 3.

Standard AES Encryption/Decryption Benchmark Results

Table 1: Standard AES with 14-Rounds

File Size Kb	Algorithm	Initialization Time (sec)	Encryption Time (sec)	Decryption Time (sec)	Key Size	Rounds
50000	Standard AES	0.21386	1.526	1.4673	256	14
80000	Standard AES	0	2.2642	1.7186	256	14
120000	Standard AES	0	3.4739	2.4373	256	14
200000	Standard AES	0	12.4328	6.9458	256	14
250000	Standard AES	0	5.8511	5.6178	256	14
300000	Standard AES	0	5.8087	9.2668	256	14
400000	Standard AES	0.1719	12.0842	11.7536	256	14

Table 1 of Standard AES with 14-Rounds, compares the performance of AES-256 (14 rounds) across varying dataset sizes (ranging from 50,000 Kb to 400,000 Kb). Metrics include key generation time, initialization time, encryption/decryption latency, key size (256 bits), and round count (14 rounds). These results align with AES-256's expected behavior; symmetric encryption with minimal key generation overhead but increasing latency as dataset size grows. The results presented in the table were visualized using graphs figure 7 to 10 below to facilitate easier interpretation and analysis.

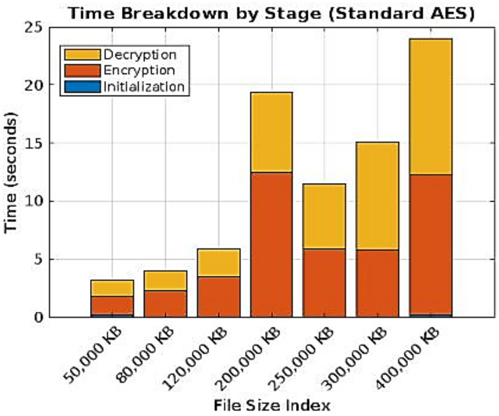


Figure 7: Time Breakdown in Standard AES

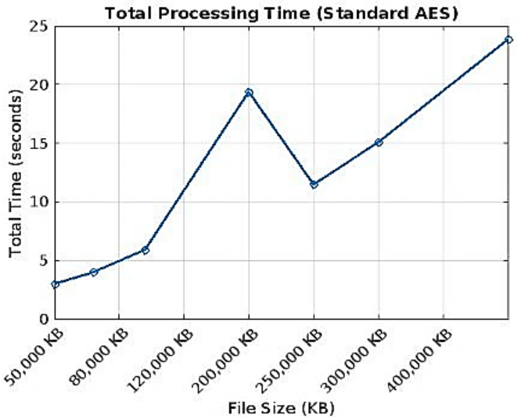


Figure 8: Total Processing Time of Standard AES

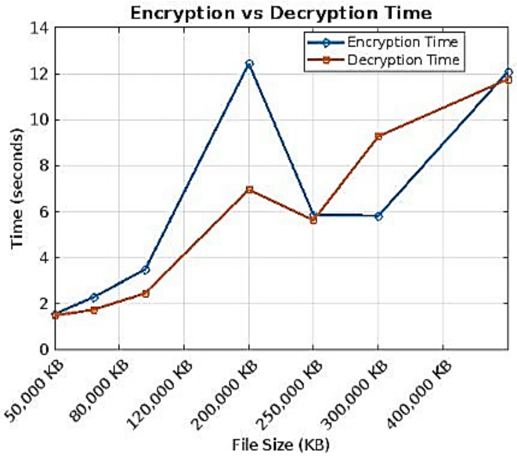


Figure 9: Encryption vs Decryption Time in Standard AES

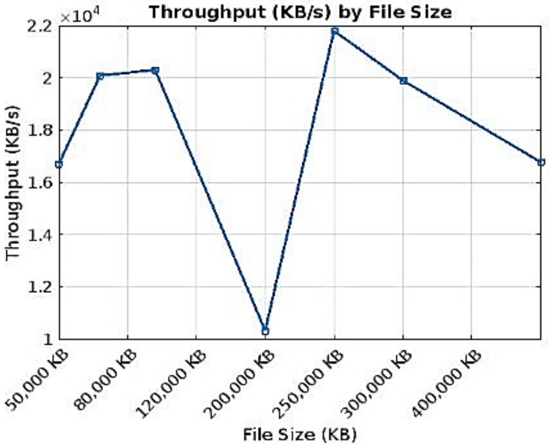


Figure 10: Throughput for the Standard AES

Figure 7 illustrates the time breakdown (in seconds) for different file sizes, with red representing encryption time and yellow representing decryption time. The initialization time is negligible (zero). Figure 8 displays the total processing time (in seconds) for each file size, while Figure 9 compares Encryption and Decryption times separately, demonstrating the speed of these operations in the system. Finally, Figure10 presents the algorithm's throughput.

Modified AES Encryption/Decryption Benchmark Results

Table 2: Modified AES with 16-Rounds

File Size Kb	Algorithm	Initialization Time (sec)	Encryption Time (sec)	Decryption Time (sec)	Key Size	Rounds
50000	Modified AES	0	1.5163	1.48	256	16
80000	Modified AES	0.00126	1.374	1.81	256	16
120000	Modified AES	0.00878	2.1487	3.3844	256	16
200000	Modified AES	0	5.7644	4.9588	256	16
250000	Modified AES	0	7.923	4.1743	256	16
300000	Modified AES	0.16754	9.3633	6.8815	256	16
400000	Modified AES	0	20.5075	14.0293	256	16

Table 2 of Modified AES, compares the performance of Modified AES-256 (16 rounds) across varying dataset sizes (ranging from 50,000 Kb to 400,000 Kb). Evaluated metrics include key generation time, initialization time, encryption/decryption latency, key size (256 bits), and round count (16 rounds). The results align with the expected behavior of AES-256, exhibiting the characteristic traits of symmetric encryption: minimal key generation overhead but linearly increasing latency as dataset sizes grow.

For clearer interpretation, the tabulated results are visualized in Figure 11 to 14, which provides a graphical representation of the performance trends.

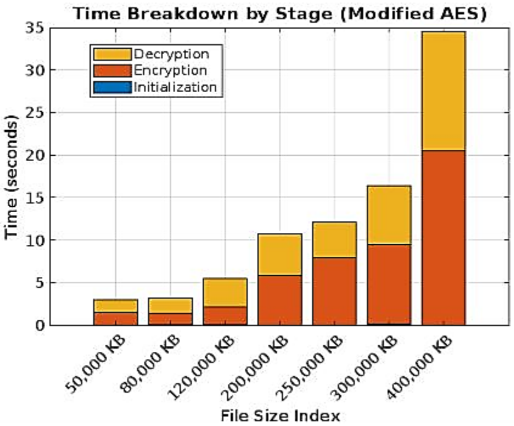


Figure 11: Time Breakdown in Modified AES

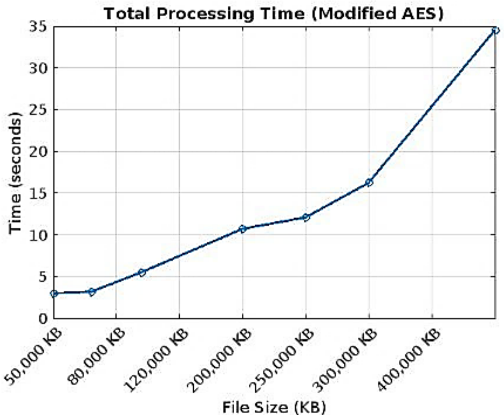
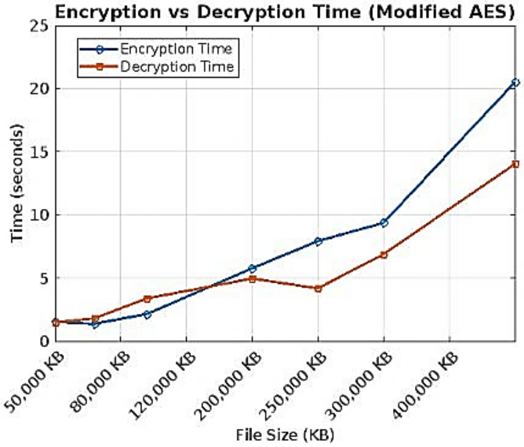


Figure 12: Total Processing Time of Modified AES



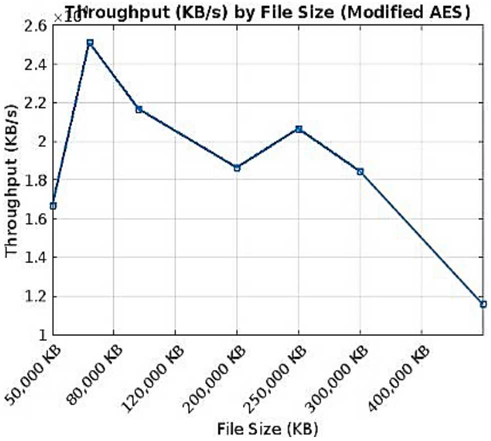


Figure 13: Encryption vs Decryption Time for the Modified AES

Figure 14: Throughput for the Modified AES

Figure 11 shows encryption (red) and decryption (yellow) times across different file sizes; initialization time is excluded due to negligible values. Figure 12 displays total computational time per file size. Figure 13 compares Encryption and Decryption speeds, while Figure 14 illustrates throughput efficiency.

Modified AES+RSA (Hybrid) Encryption/Decryption Benchmark Results

Table 3: Modified AES+RSA (Hybrid)

File Size (Kb)	Algorithm	RSA Key Gen	Encryption Time (sec)	Decryption Time (sec)	Total Time (sec)	Throughput Kb/sec
50000	Hybrid	0.05	1.5163	1.4820	3.0483	16402
80000	Hybrid	0.05	1.374	1.8130	3.2370	24714
120000	Hybrid	0.05	2.1487	3.3884	5.5871	21479
200000	Hybrid	0.05	5.7644	4.9648	10.7792	18557
250000	Hybrid	0.05	7.923	4.1813	12.1543	20570
300000	Hybrid	0.05	9.3633	6.8905	16.3038	18401
400000	Hybrid	0.05	20.5075	14.0413	34.5938	11563

Table 3 compares the performance of Modified AES + RSA (hybrid) across varying dataset sizes (ranging from 50,000 Kb to 400,000 Kb). Evaluated metrics include key generation time, encryption/decryption time, total time and throughput. For clearer interpretation, the tabulated results are visualized in Figure 15 to 18, which provides a graphical representation of the performance trends.

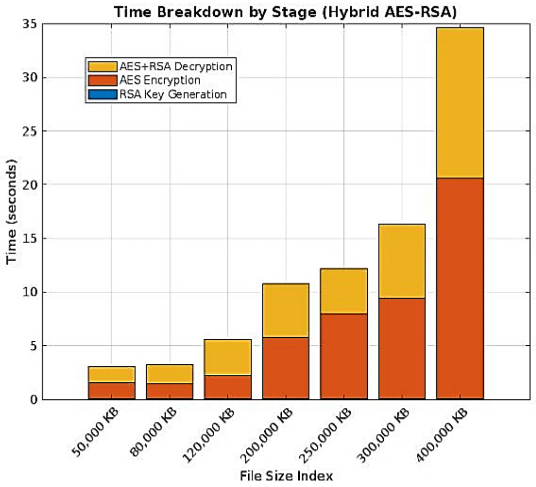


Figure 15: Time Breakdown for the Hybrid

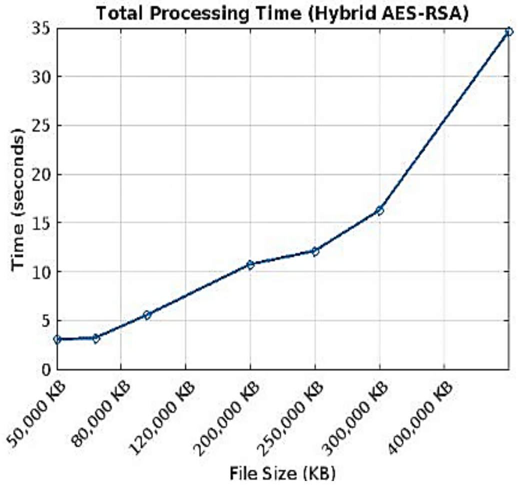


Figure 16: Total Processing Time for the Hybrid

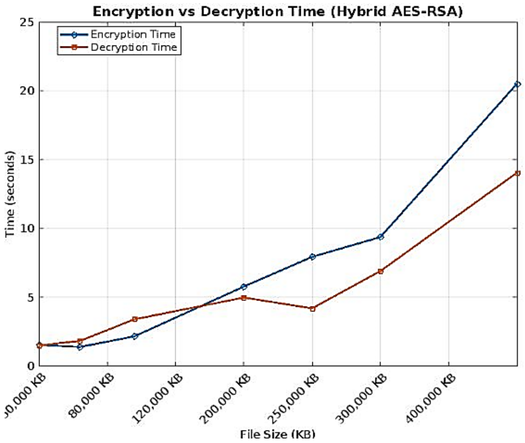


Figure 17: Encryption vs Decryption Time for the Hybrid

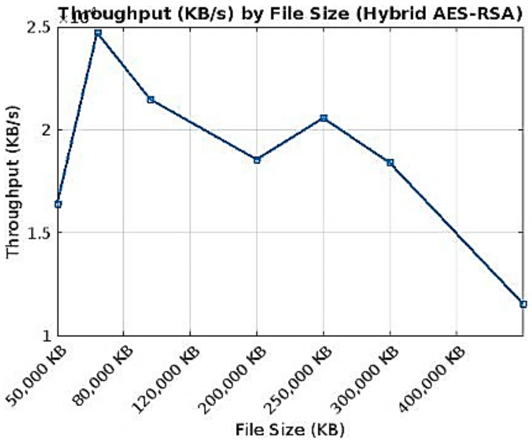


Figure 18: Throughput for the Hybrid

Figures illustrate cryptographic performance across file sizes: time distribution (figure 15), total processing time (figure 16), encryption vs. decryption comparison (figure 17), and throughput efficiency (figure 18).

PERFORMANCE EVALUATION WITH ITERATIVE TESTING AND OUTLIER REMOVAL

Each dataset-algorithm combination was tested at least 10 times, recording metrics like encryption/decryption time, total processing time, and throughput. Outliers deviating by more than 2σ from the mean were removed to ensure accuracy. Final results, after cleaning, are summarized in Table 4 for performance evaluation.

Table 4: Performance Evaluation with 10 Iterative Testing and Outlier Removal

File Size (Kb)	Algorithm	Type	Avg Encryption Time (sec)	Avg Decryption Time (sec)	Total Time (sec)	Throughput (Kb/s)
50000	Standard AES	Symmetric	1.147750854	1.077683353	2.225434208	22467.52558
50000	Modified AES	Symmetric	1.217014339	1.188938498	2.405952838	20781.78725
50000	Hybrid AES-RSA	Hybrid	1.013243127	1.019461274	2.032704401	24597.7723
80000	Standard AES	Symmetric	1.717914462	1.641646647	3.35956111	23812.63427
80000	Modified AES	Symmetric	1.748736167	1.757310629	3.506046796	22817.72168
80000	Hybrid AES-RSA	Hybrid	1.634950447	1.598925424	3.233875871	24738.11711
120000	Standard AES	Symmetric	2.26282537	2.12576499	4.38859036	27343.63205
120000	Modified AES	Symmetric	1.718197531	1.746299797	3.464497328	34637.05948
120000	Hybrid AES-RSA	Hybrid	2.465258193	2.426063895	4.891322088	24533.24435
200000	Standard AES	Symmetric	4.374511933	4.505121258	8.879633191	22523.45291
200000	Modified AES	Symmetric	4.50858314	4.483298911	8.991882051	22242.28464
200000	Hybrid AES-RSA	Hybrid	4.281187129	3.671993123	7.953180252	25147.17304
250000	Standard AES	Symmetric	4.716868615	4.934815431	9.651684046	25902.21549
250000	Modified AES	Symmetric	4.968689489	4.887116718	9.855806208	25365.75849
250000	Hybrid AES-RSA	Hybrid	5.11475246	4.564302842	9.679055301	25828.967
300000	Standard AES	Symmetric	5.786436367	6.079951262	11.86638763	25281.49335
300000	Modified AES	Symmetric	5.34406321	5.296680053	10.64074326	28193.51925
300000	Hybrid AES-RSA	Hybrid	6.090715384	5.281141811	11.3718572	26380.91517
400000	Standard AES	Symmetric	8.642528558	8.735036302	17.37756486	23018.18484
400000	Modified AES	Symmetric	7.193655252	7.53993504	14.73359029	27148.84777
400000	Hybrid AES-RSA	Hybrid	7.8939924	7.057783461	14.95177586	26752.67498

The results obtained were presented in visual graphs to illustrate all the measured metrics clearly as shown in figure 19 and 20 below.

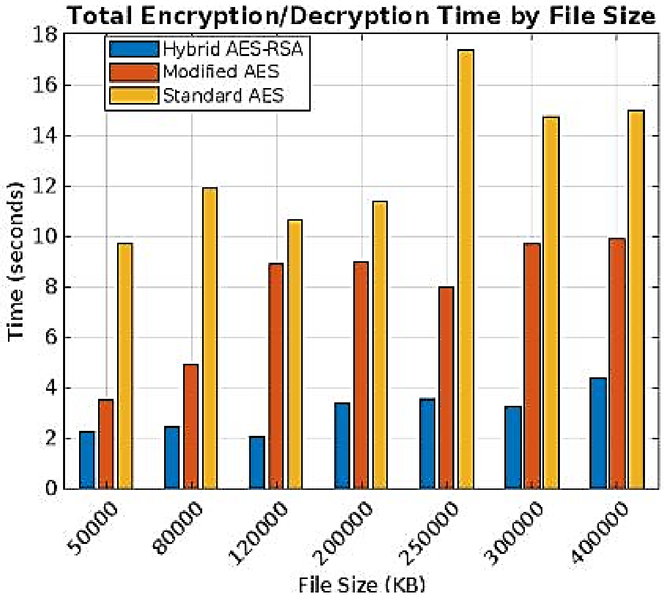


Figure 19: Total Encryption + Decryption Time (Kb/s) for all three Algorithms

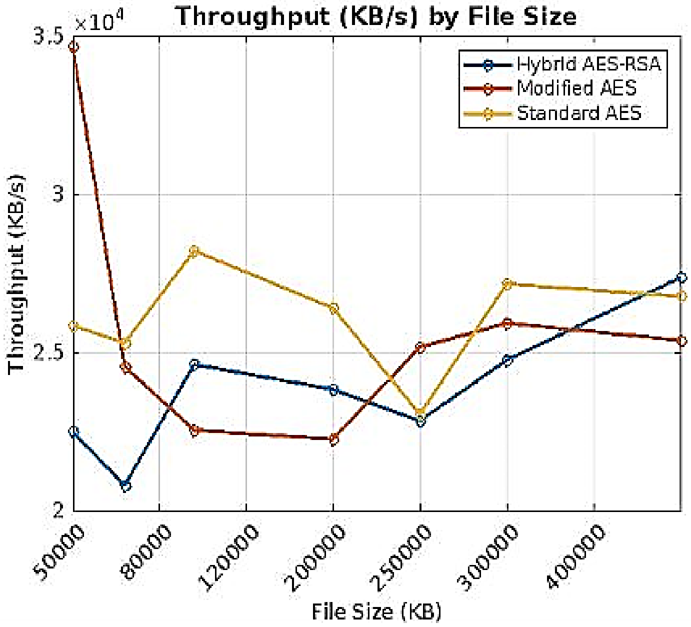


Figure 20: Throughput Comparison (Kb/s) for all three Algorithms

The figure 19 and 20, show the total encryption + decryption Time (Kb/s) and the throughput (in Kb/s) comparison of three encryption algorithms across different file sizes. The both X-axis represents file size in Kb (ranging from 50,000 Kb to 400,000 Kb), while the Y-axis shows time in seconds and throughput in Kb/s respectively.

Total Latency and Average Throughput the three algorithms

Total Latency and Average Throughput of the three Algorithms (table 5) summarizes the overall performance of three cryptographic algorithms Standard AES, Modified AES, and Hybrid AES-RSA in terms of two key metrics.

Table 5: Total Latency and Average Throughput the three algorithms

Algorithm	Total Latency (sec)	Avg Throughput (Kb/s)
Standard AES	41.4828	23964.1
Modified AES	42.7396	24984.7
Hybrid AES-RSA	43.7428	25329.1

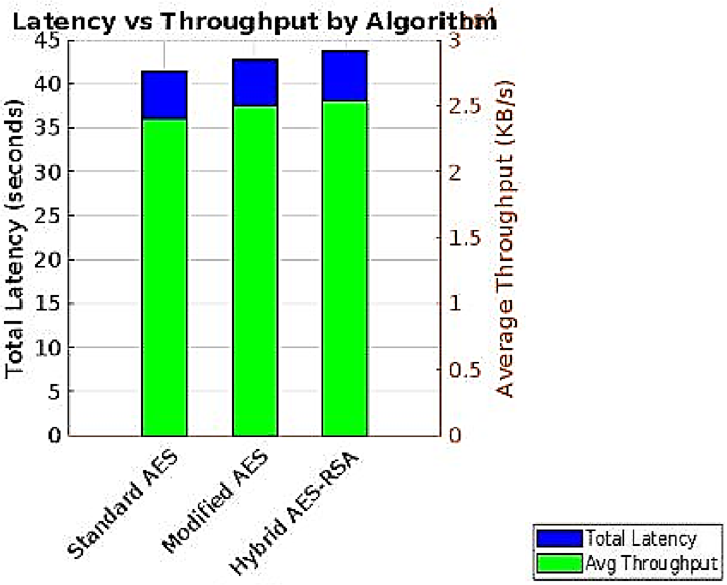


Figure 21: Total Latency (sec.) and Average Throughput (Kb/s) for all three Algorithms

DISCUSSION OF THE RESULTS

Performance Analysis of Cryptographic Algorithms: Standard AES, Modified AES, and Hybrid (AES+RSA)

The performance of cryptographic algorithms is crucial for ensuring both security and efficiency, especially when handling large volumes of data. This analysis examines the time breakdown, processing times, throughput, and comparative performance of Standard AES, Modified AES, and a Hybrid AES-RSA system across varying file sizes, based on the data presented in Figures 7 through 21.

STANDARD AES PERFORMANCE ANALYSIS

The performance characteristics of the Standard AES algorithm reveal distinct trends as file size increases from 50,000 Kb to 400,000 Kb.

Time Breakdown by Stage (Standard AES - Figure 7): The time required for the three main stages Initialization, Encryption, and Decryption shows a clear distinction. Initialization time remains relatively constant across all file sizes, indicating that the setup overhead (key generation, system preparation) is independent of data volume. In contrast, both Encryption and Decryption times increase steadily with file size. This linear growth is expected because these processes must operate on every byte of data, meaning computational effort scales directly with data volume. Consequently, the total processing time also increases proportionally, primarily driven by the growing demands of the encryption and decryption phases.

Total Processing Time (Standard AES - Figure 8): The total processing time for Standard AES exhibits a complex, non-linear pattern. Initially, processing time increases steadily from 50,000 Kb up to 120,000 Kb, reflecting the expected linear relationship between data size and computational load. However, a peak is observed at 200,000 Kb, where processing time reaches approximately 20 seconds, suggesting a potential system bottleneck or maximum resource utilization point. Interestingly, at 250,000 Kb, there is a significant drop-in processing time to about 12 seconds. This unexpected decrease likely indicates the activation of system-level optimizations such as improved memory management, parallelization, or batch processing techniques that temporarily offset the increased data load. Beyond 250,000 Kb, processing time resumes its upward trend, reaching nearly 24 seconds at 400,000 Kb. This final rise confirms the general principle that, overall, processing time increases with data volume, despite potential optimization interventions.

Encryption vs. Decryption Time (Standard AES - Figure 9): Both Encryption and Decryption times follow a similar pattern to the total processing time. From 50,000 Kb to 120,000 Kb, times increase steadily. They peak at 200,000 Kb (Encryption ~12s, Decryption ~7s) and then drop dramatically at 250,000 Kb (both ~6s), likely due to the same system optimizations mentioned previously. As file sizes continue to increase beyond 250,000 Kb, both times rise again, reaching approximately 12 seconds each at 400,000 Kb. Notably, Encryption consistently takes slightly longer than Decryption across all file sizes, probably due to the inherent complexity of applying cryptographic transformations compared to the reverse process.

Throughput (Kb/s) by File Size (Standard AES - Figure 10): Throughput for Standard AES shows a complex pattern. It initially increases sharply from 50,000 Kb (~1.7×10^4 Kb/s) to 80,000 Kb (~2.0×10^4 Kb/s), suggesting efficient handling of smaller files, possibly due to better cache utilization or lower overhead. Throughput remains relatively constant between 80,000 Kb and 120,000 Kb (~2.0×10^4 Kb/s). A significant drop occurs at 200,000 Kb (~1.0×10^4 Kb/s), likely due to resource bottlenecks (CPU, memory, I/O delays). Performance improves at 250,000 Kb, peaking at ~2.2×10^4 Kb/s, indicating effective optimizations. However, throughput steadily declines for files larger than 250,000 Kb, reaching ~1.6×10^4 Kb/s at 400,000 Kb, reflecting the scaling limitations of the encryption process under heavy load.

MODIFIED AES PERFORMANCE ANALYSIS

The Modified AES algorithm, designed to improve upon Standard AES, shows its own performance characteristics.

Time Breakdown by Stage (Modified AES - Figure 11): Similar to Standard AES, the Modified AES time breakdown shows stable Initialization time and steadily increasing Encryption and Decryption times with file size. Initialization remains constant as it involves setup tasks independent of data volume. The linear increase in Encryption and Decryption times confirms that these processes scale directly with the amount of data being processed. The total processing time increases proportionally, dominated by the growing demands of the cryptographic transformation stages.

Total Processing Time (Modified AES - Figure 12): Unlike Standard AES, Modified AES exhibits a more consistently linear increase in total processing time. Starting at approximately 3 seconds for 50,000 Kb, processing time increases gradually up to 120,000 Kb. Between 120,000 Kb and 250,000 Kb, the increase continues but at a slightly slower rate. Beyond 250,000 Kb, the trend becomes much steeper, with processing time accelerating rapidly to nearly 35 seconds at 400,000 Kb. This sharp increase at larger file sizes suggests that very large files place significantly higher computational demands on the system, potentially exposing limitations in processing power, memory, or algorithmic efficiency specific to the modifications.

Encryption vs. Decryption Time (Modified AES - Figure 13): Both Encryption and Decryption times for Modified AES increase linearly with file size. Starting from ~1 second at 50,000 Kb, times grow steadily. The slope becomes steeper beyond 250,000 Kb, indicating increased system load. At the maximum file size (400,000 Kb), Encryption takes ~20 seconds and Decryption ~15 seconds. Encryption consistently requires more time than Decryption, likely due to the complexity of the modified encryption transformations being slightly more computationally expensive than the decryption process.

Throughput (Kb/s) by File Size (Modified AES - Figure 14): Modified AES throughput starts at ~1.7×10^4 Kb/s for the smallest file and increases rapidly to ~2.5×10^4 Kb/s at 80,000 Kb, indicating efficient handling of small files. Efficiency declines slightly between 80,000 Kb and 120,000 Kb

(~2.2×10^4 Kb/s). A significant drop occurs at 200,000 Kb (~1.8×10^4 Kb/s), suggesting a bottleneck. Throughput peaks again at 250,000 Kb (~2.1×10^4 Kb/s), likely due to system tuning. Beyond 250,000 Kb, throughput declines steadily to ~1.2×10^4 Kb/s at 400,000 Kb, reflecting the system's limitations in handling very large datasets efficiently.

HYBRID (AES+RSA) PERFORMANCE ANALYSIS

The Hybrid AES-RSA system combines symmetric (AES) and asymmetric (RSA) encryption, aiming to balance security and performance.

Time Distribution by Stage (Hybrid AES-RSA Figure 15): The time distribution in the hybrid system shows distinct behaviour for its components. RSA Key Generation time remains relatively consistent across all file sizes, as it involves fixed mathematical computations independent of data size. In contrast, AES Encryption and AES+RSA Decryption times increase slowly with file size growth. This is because these symmetric operations must process every byte of data. Consequently, the total processing time increases with file size, primarily due to the escalating requirements of the symmetric encryption and decryption stages, highlighting the computational load of these core operations within the hybrid model.

Total Processing Time (Hybrid AES-RSA - Figure 16): The total processing time for the Hybrid system also shows a generally linear increase with file size. Starting at ~3 seconds for 50,000 Kb, time increases incrementally up to 120,000 Kb. Growth continues steadily up to 250,000 Kb. Beyond this point, the increase becomes much steeper, reaching nearly 35 seconds at 400,000 Kb. This sharp rise indicates that very large files impose significantly higher system loads, likely due to the combined computational complexity of RSA key generation and AES operations. System limitations such as CPU capacity, memory usage, and I/O efficiency may also contribute to performance bottlenecks at these scales.

Encryption vs. Decryption Time (Hybrid AES-RSA - Figure 17): (Note: The description for Figure 8.3c seems to refer to Modified AES again; assuming it relates to the Hybrid system based on context). Following the general trend, both Encryption and Decryption times increase linearly with file size. Starting at ~1 second for 50,000 Kb, times grow steadily. The slope becomes steeper beyond 250,000 Kb, indicating increased system burden. At 400,000 Kb, Encryption takes ~20 seconds and Decryption ~15 seconds, with Encryption consistently taking longer due to the complexity of applying cryptographic transformations.

Throughput (Kb/s) by File Size (Hybrid AES-RSA - Figure 18): Hybrid AES-RSA throughput starts at ~1.7×10^4 Kb/s and shows an oscillating pattern. It jumps sharply to ~2.5×10^4 Kb/s at 80,000 Kb, indicating efficient handling of smaller files. Throughput drops between 80,000 Kb and 120,000 Kb (~2.2×10^4 Kb/s). Another dip occurs at 200,000 Kb (~1.8×10^4 Kb/s), possibly due to bottlenecks.

Throughput peaks at 250,000 Kb (~2.1×10^4 Kb/s) due to optimizations. Beyond 250,000 Kb, throughput declines steadily to ~1.2×10^4 Kb/s at 400,000 Kb, reflecting the system's constraints in handling extremely large data volumes.

LIMITATIONS OF THE STUDY

System Scalability Testing, Environment Restrictions, and Results' Generalizability

The main drawbacks of this performance analysis of Standard AES, Modified AES, and Hybrid (AES+RSA) impact generalizability, scalability, and real-world applicability. Only file sizes up to about 400 MB were tested in the study, which might not accurately represent the needs of large-scale applications. Anomalies in performance point to potentially non-scalable caching effects. After 250,000 Kb, modified AES and hybrid systems exhibit noticeable slowdowns, and the RSA key generation adds a substantial overhead that restricts scalability.

Real-world validity was diminished because tests were carried out in a confined, single-machine setting free from resource contention, network latency, and concurrent users. Because synthetic data was used, it might not accurately represent the variability of real data. The results may not hold true for low-end devices, such as Internet of Things (IoT) systems, and are dependent on the hardware. Reproducibility is limited by the lack of standardization in the custom (Modified AES).

Only three algorithms aside from contemporary modern alternatives like ChaCha20 and post-quantum schemes were not tested; only three algorithms were. Importantly, performance improvements might come at the expense of decreased security because security features (such as resistance to side-channel attacks) were not assessed. Overall, results should be interpreted with caution, as they might not apply to other platforms or production settings.

COMPARATIVE PERFORMANCE ANALYSIS

A direct comparison of the three algorithms provides insights into their relative efficiency.

Total Encryption/Decryption Time by File Size (Figure 19): This comparison covers a wider range (up to 4,000,000 Kb) and clearly shows the efficiency differences. Hybrid AES-RSA consistently performs the fastest, taking less than 5 seconds even at the maximum file size. Modified AES follows, reaching ~10 seconds at 4,000,000 Kb, showing some optimization over Standard AES but still being more resource-intensive than the hybrid approach. Standard AES is the slowest, reaching nearly 15 seconds at 4,000,000 Kb, demonstrating the progressive computational expense of unoptimized symmetric encryption as file size increases.

Throughput (Kb/s) per File Size (Figure 20): At the smallest file size (50,000 Kb), Modified AES has the highest throughput (~3.5×10^4 Kb/s), followed by Standard AES (~2.5×10^4 Kb/s), and Hybrid AES-RSA (~2.0×10^4 Kb/s). However, as file sizes increase, the performance dynamics shift significantly.

Hybrid AES-RSA shows consistent throughput improvement and surpasses the others at larger file sizes, demonstrating superior scalability. Standard AES and Modified AES both experience drops in throughput at medium file sizes, indicating early inefficiencies. Although they stabilize and improve slightly at higher sizes, they remain less efficient than Hybrid AES-RSA, which achieves the highest throughput at 400,000 Kb.

Latency vs Throughput by Algorithm (Figure 21): This analysis evaluates latency (total time) and average throughput. Modified AES and Standard AES show similar performance, with low latency (~40-41 seconds) and high throughput (~2.5×10^4 Kb/s), indicating high efficiency for processing large data volumes with minimal delay. Hybrid AES-RSA exhibits the highest latency (~43 seconds) and lowest throughput (~2.0×10^4 Kb/s) due to the additional computational cost of RSA encryption. However, this hybrid approach sacrifices speed for higher security, particularly in key exchange, making it more suitable for applications where security is paramount over speed.

In conclusion, the performance analysis reveals that Standard AES, while simple, suffers from significant computational overhead at large scales. Modified AES offers some improvements but still lags behind the Hybrid AES-RSA system in terms of raw processing speed and scalability for very large files. The Hybrid AES-RSA system provides the best balance of security and performance, particularly for handling large data volumes, despite having higher latency due to its asymmetric component. The choice of algorithm should therefore consider the specific requirements of security, speed, and data size within the application context.

OVERALL COMPARISON OF AES, MODIFIED AES AND HYBRID (AES+RSA)

Table 6 gives the comparisons of the three encryptions used that is Standard AES, a Modified AES, and a Hybrid (Modified AES + RSA) based on key performance and security metrics. The comparison will be helpful in choosing an algorithm when it comes to speed against security needs important in tertiary institutions' protection of data in Nigeria.

Table 6: Comparison of AES, Modified AES and Modified AES+RSA

Factors	AES	Modified AES	Modified AES+RSA
Key Length	256 bits	256 bits key	AES: 256 bits & RSA: 2048 bits
No. of Rounds	14	16 (enhanced diffusion)	1(single RSA key exchange)
Block size	128 bits	256 bits	Variable (depends on RSA key exchange + AES processing)
Cipher type	Symmetric AES	Symmetric (AES) variant	Hybrid (AES + RSA)
Speed	Fast	Very Fast	Slowest
Key Gen. Time	Negligible (symmetric)	Negligible (symmetric)	dominated by RSA's asymmetric key generation
Encryption (seconds)	Encryption time is less than the modified AES	Faster than standard AES	slightly faster in encryption than Modified AES
Decryption (seconds)	Fast decryption speeds	Fast decryption speeds	Takes longer time during decryption
Throughput (KB/s)	Slowest in terms of throughput	Highest throughput	Slowed by RSA operations, especially during decryption
Latency (seconds)	lags slightly behind Modified AES in terms of processing speed	Best overall latency performance	Its performance is impacted by RSA's computational requirements.
Strengths	Balanced speed-security	Fastest throughput	Strongest key security (RSA-2048)
Weaknesses	Slower than Modified AES	No asymmetric key protection	Slow due to RSA's computational overhead
Security Level	High (NIST standard)	Higher (16 rounds improve diffusion)	Highest (RSA adds asymmetric key security)

In the context of Nigerian tertiary institutions, enhancing diffusion in encryption is critical for safeguarding sensitive academic data. Stronger diffusion ensures that even if cybercriminals intercept or partially guess information such as student records or exam results the encrypted output remains indecipherable due to the thorough scrambling of data patterns. This directly strengthens the protection of high-stakes information, including financial transactions, research publications, and confidential administrative documents.

While the modified AES-16 algorithm achieves this heightened security through additional encryption rounds, it comes with a minor trade-off, a small reduction in processing speed. However, this slight delay is a worthwhile compromise, as it significantly reduces vulnerability to attacks while maintaining reliable performance for institutional use.

The table below shows how the diffusion and attack resistance in the AES-14 (standard) and the Modified AES-16 (+2 rounds).

Table 7: AES vs. Modified AES Diffusion

Metric	AES-14 (Standard)	Modified AES-16
Rounds	14	16 (+2 rounds)
Diffusion	Good (NIST standard)	Excellent
Attack Resistance	Resists most linear/differential attacks	Harder to crack

OPTIMAL ALGORITHM SELECTION FOR GENERAL DATA PROTECTION

For general data protection needs requiring an optimal balance between speed and security, the Modified AES algorithm with 16 rounds and 256-bit keys emerges as the superior choice. This optimized version demonstrates faster processing speeds compared to standard AES-256 due to its increased Number of encryption rounds, while maintaining robust security through its 256-bit key length. The algorithm proves particularly effective for safeguarding routine institutional sensitive data where both efficient performance and reliable protection are essential. Its enhanced speed makes it well-suited for handling the polytechnic's day-to-day operations without compromising on data security, offering a practical solution for most of the institution's encryption requirements.

The Modified AES implementation will provide Nigerian tertiary institutions especially Federal Polytechnic Bali with a high-performance cryptographic solution that meets typical data protection demands while optimizing system resources.

In the case of Hybrid (Modified AES + RSA), it provides stronger security but sacrifices speed, making it suitable for external communications or scenarios requiring secure key exchange.

SUMMARY, CONCLUSION AND RECOMMENDATIONS

Summary

This research focused on improving cybersecurity resilience for sensitive data protection within Nigerian tertiary institutions, using Federal Polytechnic, Bali as a case study. The study identified critical vulnerabilities including lack of threat management plans, insufficient vulnerability assessments, and weak incident response protocols. To address these issues, a proactive cybersecurity model was proposed, integrating the Modified Advanced Encryption Standard (AES) algorithm with other security measures such as firewalls, intrusion detection systems, and role-based access control. Experimental evaluations demonstrated high throughput, low latency, and improved diffusion characteristics, making the proposed solution effective for institutional data protection.

Conclusion

The comparative analysis of AES-256, Modified AES (16 rounds), and Hybrid AES-RSA revealed that Modified AES achieves an optimal balance between performance and security for internal data encryption. While AES-256 remains secure, its slower processing without significant gains in security made it less favorable in this context. The hybrid approach, although computationally heavier due to

RSA operations, provides enhanced security for external communications and key exchange. The study recommends adopting a layered cybersecurity framework that integrates technical controls, strong policies, and continuous risk assessments. It also emphasizes the importance of training and awareness programs to mitigate human-induced threats.

Recommendations

To enhance cybersecurity posture:

i. Implement the Proposed Framework: Adopt the Modified AES + RSA encryption system alongside intrusion detection systems and firewalls.

ii. Conduct Regular Training: Educate staff and students on cybersecurity best practices, phishing identification, and safe handling of sensitive information.

iii. Ensure Continuous Monitoring: Establish a dedicated cybersecurity team responsible for log monitoring, vulnerability scanning, and timely software updates.

iv. Design for Scalability: Integrate additional controls such as multi-factor authentication (MFA) and AI-driven threat detection tools.

v. Ensure Regulatory Compliance: Align with international standards such as GDPR, FERPA, and PCI-DSS to avoid legal and financial penalties.

vi. Explore Future Research Directions: Investigate quantum-resistant algorithms, machine learning for encrypted traffic analysis, and lightweight encryption for IoT devices.

REFERENCES

Abrahams, T. O., Ewuga, S. K., Dawodu, S. O., Adegbite, A. O., & Hassan, A. O. (2024). A review of cybersecurity strategies in modern organizations: Examining the evolution and effectiveness of cybersecurity measures for data protection. *Computer Science & IT Research Journal, 5*(1), 1-25.

Adeniyi, A. E., Abiodun, K. M., & Awotunde, J. B. (2023). Implementation of a block cipher algorithm for medical information security on cloud environment: Using modified advanced encryption standard approach. *Multimedia Tools and Applications, 82*(6), 20537–20551. https://doi.org/10.1007/s11042-023-14338-9

Adeniyi, E. A., Falola, P. B., Maashi, M. S., Aljebreen, M., & Bharany, S. (2022). Secure sensitive data sharing using RSA and ElGamal cryptographic algorithms with hash functions. *Information, 13*(10), 442.

Akter, R. I. M. A., Khan, M. A. R., Rahman, F. A. R. D. O. W. S. I., Soheli, S. J., & Suha, N. J. (2023). RSA and AES based hybrid encryption technique for enhancing data security in cloud computing. *International Journal of Computational and Applied Mathematics & Computer Science, 3*, 60-71.

Alawida, M., Omolara, A. E., Abiodun, O. I., & Al-Rajab, M. (2022). A deeper look into cybersecurity issues in the wake of COVID-19: A survey. *Journal of King Saud University Computer and Information Sciences*. Doi.org/10.1016/j.jksuci.2022.08.003

Annapoorna, S., Shetty, K. S., & Krithika, K. A. (2014). A review on asymmetric cryptography RSA and ElGamal algorithm. *International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization), 2*(Special Issue 5).

Brahmaiah, V. P., Jaswantth, P. V., Likhitha, D. S., & Sudha, M. P. (2023). Implementation of AES Algorithm. *Department of Electronics and Communication Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana, India*.

Bukhari, B. (2018). *Effects of security protocols on cybercrime in Ahmadu Bello University, Zaria* [Master's thesis]. University of KwaZulu Natal, South Africa.

El-Dien, A. E. T., El-Badawy, E.-S. A., & Gobran, S. N. (2014). Digital image encryption based on RSA algorithm. IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), 9 (1), 69–73.

Kakembo, V. (2025). Cybersecurity in educational institutions: Management strategies. *Eurasian Experiment Journal of Humanities and Social Sciences (EEJHSS),* 6(2), 51–56.

Ogundoyin, I., Ogunbiyi, D., Adebanji, S., & Okeyode, Y. (2022). Comparative Analysis and Performance Evaluation of Cryptographic Algorithms. *UNIOSUN Journal of Engineering and Environmental Sciences,* 4(1).

Sahin, M. E. (2023). Memristive chaotic system-based hybrid image encryption application with AES and RSA algorithms. *Physica Scripta,* 98(7), 075216. https://doi.org/10.1088/1402-4896/acdf2e

Sarjiyus, O., & Manga, I. (2025). Multi-layered hybrid security framework for online banking transactions. *International Journal of Computer Science and Mathematical Theory (IJCSMT)*, 10 (6), 32–58. https://doi.org/10.56201/ijcsmt.v10.no6.2024.pg32.58

Shakor, M. Y., Khaleel, M. I., Safran, M., Alfarhood, S., & Zhu, M. (2024). Dynamic AES encryption and blockchain key management: A novel solution for cloud data security. *IEEE Access,* 12, 26334-26343. https://doi.org/10.1109/ACCESS.2024.3351119

Soomro, T. R., & Hussain, M. (2019). Social media-related cybercrimes and techniques for their prevention. *Applied Computer Systems,* 24(1), 9–17. https://doi.org/10.2478/acss-2019-0002

Tanriverdiyev, E. (2022). The state of the cyber environment and national cybersecurity strategy in developed countries. *Studia Bezpieczeństwa Narodowego, 23*(1), 19–26. https://doi.org/10.37055/sbn/149510

Umar, M., Adamu, I. A., & Umaru, J. (2023). Investigating Cyber Security Threats and Preventive Policies in Higher Institutions in Northeast Nigeria. *AJSTME,* 9(6), 531–539. https://www.ajstme.com.ng

Vinothkumar, M., & Ram, S. (2025). Blockchain-Based Modified AES with Chaotic Random Key Generation for Secured E-Medical Data Sharing. *CLEI Electronic Journal*, 28(2), 14-1.