IRAQI Academic Scientific Journals
العراقية المجلات الاكاديمية العلمية
University of Kut Journal
ISSN (E): 2616 - 7808 II ISSN (P): 2414 - 7419
js.uokut.edu.iq
k.u.c.j.sci@alkutcollege.edu.iq
Vol. 11 , Issue 1 , June 2026

Time-Adaptive Hybrid Encryption Pipeline For Secure Text Processing Using AES, Homomorphic Encryption, and PQC

Fatima Majeed Farhan [1] , **Jamal Kh-Madhloom** [2]

Abstract

In cloud environments, secure text processing is implemented using a time-adaptive hybrid encryption approach. Using a traditional encryption model (AES), the loss of control over cloud data creates significant complication; the homomorphic encryption (HE) model incurs processing delays and increased cloud computing costs. The proposed model utilizes a hybrid approach where AES, HE/CKKS, and PQC/Kyber768 are integrated within a cloud environment.

The design and functionality of the hybrid approach are dictated by the Intelligent AI-based router, designed to analyze cloud text processor enabling the hybrid AES model computing cloud's metadata pertaining to text size, sensitivity, and cloud response time. Secure Enclave environments and HE and AES quantitative partition the data. A tailored functionality partitions the HE and AES environments. A blockchain-based audit trail encrypts and secures all primary events, thus providing a permanent record of all integral cryptarithmic transfers.

The model in question performed processing of a representative text workload of size 0.39MB. The results indicated a total server-side processing duration of 347.40 ms, coupled with a client-side decryption duration of 14.60 MS, leading to a secure cycle duration of below 0.4 seconds. The level of encrypted text size expansion was limited to 1.36× (previous models exhibited a much larger expansion), with a transfer rate of 1.46MB/s. The results of homomorphic encryption exhibited 100 percent accuracy, which confirms that the model successfully balanced performance, storage efficiency, and optimal security of approximately 192 bits, with flexibility concerning potential future quantum threats.

Keywords: Homomorphic encryption, AES-256, post-quantum cryptography, AI-based routing, blockchain audit

Affiliation of Authors
[1, 2] College of Education for Pure Science, Wasit University, Iraq, Wasit, Kut, 52001

[1] std.2024205.f.farhan@uowasit.edu.iq
[2] jamalkh@uowasit.edu.iq

[1] **Corresponding Author**

Paper Info.
Published: Jun. 2026

خط أنابيب تشفير هجين متكيف مع الوقت لمعالجة النصوص الآمنة باستخدام AES والتشفير المتماثل وPQC

فاطمة مجيد فرحان [1] ، **جمال خضير مظلوم** [2]

المستخلص

في بيئات الحوسبة السحابية، تُنفَّذ معالجة النصوص الآمنة باستخدام نهج تشفير هجين متكيف مع الوقت. عند استخدام نموذج التشفير التقليدي (AES)، تظهر تعقيدات فقدان التحكم في بيانات السحابة؛ بينما يتسبب نموذج التشفير المتماثل المتجانس (HE) في تأخيرات في المعالجة وزيادة تكاليف الحوسبة السحابية. يستخدم النموذج المقترح نهجًا هجينًا يدمج فيه كل من AES وHE/CKKS وPQC/Kyber768 ضمن بيئة سحابية.

يُحدَّد تصميم ووظائف النهج الهجين بواسطة موجه الذكاء الاصطناعي الذكي، المصمم لتحليل معالج النصوص السحابية الذي يُمكّن نموذج AES الهجين من حساب البيانات الوصفية للسحابة المتعلقة بحجم النص وحساسيته ووقت استجابة السحابة. تعمل بيئات Secure Enclave وHE وAES على تقسيم البيانات كميًا. كما تعمل وظيفة مُخصصة على تقسيم بيئات HE وAES. يقوم سجل تدقيق قائم على تقنية البلوك تشين بتشفير وتأمين جميع الأحداث الأساسية، مما يوفر سجلًا دائمًا لجميع عمليات نقل البيانات المشفرة المتكاملة.

أجرى النموذج المذكور معالجةً لحجم بيانات نصية نموذجية يبلغ 0.39 ميجابايت. أشارت النتائج إلى أن إجمالي مدة المعالجة على جانب الخادم بلغت 347.40 مللي ثانية، بالإضافة إلى مدة فك تشفير على جانب العميل بلغت 14.60 مللي ثانية، مما أدى إلى مدة دورة آمنة تقل عن 0.4 ثانية. وقد اقتصر مستوى زيادة حجم النص المشفر على 1.36 ضعف (بينما أظهرت النماذج السابقة زيادة أكبر بكثير)، بمعدل نقل بيانات يبلغ

انتساب الباحثين
[1، 2] كلية التربية للعلوم الصرفة، جامعة واسط، العراق، واسط، الكوت، 52001

[1] std.2024205.f.farhan@uowasit.edu.iq
[2] jamalkh@uowasit.edu.iq

[1] **المؤلف المراسل**

معلومات البحث
تأريخ النشر : حزيران 2026

1.46 ميجابايت/ثانية. وأظهرت نتائج التشفير المتماثل المتجانس دقةً بنسبة 100%، مما يؤكد نجاح النموذج في تحقيق توازن بين الأداء وكفاءة التخزين والأمان الأمثل لحجم بيانات يبلغ حوالي 192 بت، مع مرونة في مواجهة التهديدات الكمومية المحتملة في المستقبل.

الكلمات المفتاحية: التشفير المتماثل، معيار التشفير المتقدم AES-256، التشفير ما بعد الكم، توجيه الذكاء الاصطناعي، تدقيق سلسلة الكتل

1. Introduction

Cloud computing has radically transformed how organizations store, manage, and analyze text data, by providing flexible scalability and cost-effective access to computing resources. However, outsourcing processing operations to third-party infrastructure raises significant confidentiality, data integrity, and regulatory compliance concerns, especially in multi-tenant environments where internal employees with malicious intent or compromised virtual machines may be able to access content after it has been decrypted [1][2]. As stated in the AES standard, The Advanced Encryption Standard (AES) specifies a FIPS-approved cryptographic algorithm that can be used to protect electronic data. The AES algorithm is a symmetric block cipher that can encrypt (encipher) and decrypt (decipher) digital information. The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits [3].Despite its wide spread, AES and traditional symmetric encryption in general share a fundamental limitation that performing meaningful operations on encrypted data requires decrypting it first. As a result, any cloud analytics performed on AES-protected data opens what is known as a "decryption window" within the service provider's environment, weakening the principle of end-to-end confidentiality and complicating compliance with privacy protection regulations such as the General Data Protection Regulation (GDPR) and the Health Insurance Transfer and Protection Act (HIPAA) [4][5]. Formally homomorphic encryption (Homomorphic Encryption - HE) provides a complementary model, allowing calculations to be performed directly on encrypted text without having to decrypt them [6]. Modern mathematical network-based (Lattice-based) schemes such as BFV and CKKS support rich and advanced operations and have been successfully used to implement secure automated inference, encrypted database queries, and privacy-preserving statistics [7][8][9]. However, the high computational cost, growing noise within encrypted text, and its swelling represent significant challenges for applications that require real-time or semi-real-time performance [10]. At the same time, the long-term threat posed by quantum computers has stimulated the development of quantum computing resistant cryptography (PQC). Network based schemes such as CRYSTALS-Kyber chosen by NIST for standardization provide key encapsulation mechanisms and public key cryptographic functions that are resistant to both classical and quantum attacks [11][12]. Integrating PQC technologies into cloud business environments is essential to counter "collect now, decrypt later" strategies, but PQC alone does not enable calculations on encrypted data. Therefore, recent research has focused on designing hybrid cryptographic architectures that combine multiple cryptographic primitives to meet the performance and security requirements of each application. Examples include the evaluation of AES circuits using morphologically symmetric cryptography

[13], hybrid HE search and retrieval schemes [14][15][16], and multi-layer frameworks that integrate AES within HE or within attribute-based cryptography (Attribute-Based Encryption) [17]. AI-powered strategies have also been proposed to select the most appropriate cryptographic path or transaction set based on workload characteristics [18][19]. However, current proposals often lack an integrated, time-adaptive architecture that simultaneously addresses requirements for performance, flexibility against quantum attacks, and auditability and accountability [20].

2. Problem Statement

Processing sensitive text in modern cloud computing environments faces a fundamental dilemma: balancing security, performance, and future sustainability. Current solutions exhibit several key limitations:

1. Symmetry Encryption Limitations (AES): Classic encryption demands the decryption of data before its computation, exposing the plaintext to risks associated with the "window of vulnerability" at the server side.
2. Latency and Data Overhead of Homomorphic Encryption (HE): Though HE (such as CKKS) enables computations on data, it generates considerable latency and excessive overhead, making it non-viable for cloud computing purposes.
3. Lack of resistance to quantum computing: Many current systems rely on traditional algorithms that may be vulnerable to quantum computing attacks in the future, necessitating the use of post-quantum cryptography techniques.
4. Lack of intelligent adaptive mechanisms: Current hybrid systems rely on fixed methods and lack intelligent mechanisms capable of dynamically balancing privacy and performance based on data characteristics.

Hence, there arises the necessity of having a time adaptive hybrid encryption architecture which would combine both Homomorphic Encryption and post-quantum cryptography technologies in an intelligent routing network, with the ultimate goal of achieving maximum security with minimum time.

3. Related Work

This section reviews and critically analyzes previous research related to secure text processing in cloud computing environments. The discussion was organized around five main axes: the efficiency of formal symmetric encryption, hybrid encryption frameworks, encryption resistant to quantum attacks, cryptographic adaptation based on artificial intelligence, and audit mechanisms based on blockchain technology. This analysis highlights the shortcomings of previous work and paves the way for an integrated and time-adaptable solution.

3.1. Homomorphic Encryption and performance improvement

Gentry's first ground-breaking work on fully homomorphic encryption (FHE) [6] showed the first instance where arbitrary computations could be carried out on the encrypted texts, however, he also pointed out the excessive performance burden because of the need for bootstrapping (bootstrapping) process. Following that, efforts have been directed to somewhat homomorphic encryption (SHE) and approximate homomorphic encryption schemes like BFV and CKKS, where comprehensiveness is sacrificed for more efficiency in certain application domains [7][9].

Trama et al evaluated applications of the AES algorithm based on symmetric cryptography under the BFV and CKKS schemes, showing that direct (primitive) evaluation of block cryptographic circuits results in very high delay time due to the multiplicative depth and cost of bootstrapping [13]. Gong et al also reviewed algorithmic and hardware-based acceleration techniques for FHE encryption, concluding that even with advanced improvements, fully homomorphic encryption remains several orders of magnitude slower in performance than traditional symmetric encryption in general workloads [10]. Ike et al. proposed a hybrid architecture that combines FHE and SHE to fine-tune functionality versus performance [16]. On the other hand, Zheng et al. and Chan et al. showed that hybrid schemes combining symmetric encryption and traditional encryption techniques can make machine learning inference on encrypted data more applicable to servers and edge devices, respectively [21] [22].

3.2 Hybrid Encryption mechanisms to support data security cloud storage and processing

A variety of hybrid schemes have been introduced to mitigate the high performance cost of symmetric encryption (HE) while maintaining strong security guarantees. Song et al used hybrid symmetric encryption in information retrieval while maintaining privacy, by combining different HE patterns to reduce response time [14]. Pothireddy et al. combined FHE with SHA-3-based data integrity mechanisms for secure cloud data management [15].

The MECS-Press group also proposed an improved framework called AES-IQCP-ABE that combines symmetric encryption and attribute-based encryption, along with mechanisms inspired by quantum computing-resistant encryption (PQC) [17] [23]. Alobaydi et al. (as cited in [14]) studied the integration of the AES algorithm with Paillier and Blowfish for cloud data storage, achieving flexibility in key management with medium latency.

3.3 Post-Quantum Cryptography for Cloud Computing security

Nwaga et al. explored a particular lattice-based PQC algorithm on the cloud computing networks and stressed that Kyber-style key encapsulation mechanisms could be a good fit for hybrid key exchange and the long-term protection of data [19]. Ganesh also examined multiple AES-centric designs coupled with PQC-based key exchange and stressed the necessity of bringing existing frameworks to a post-quantum computing state [11].

Das and Kumar empirically evaluated NIST candidates in post-quantum cryptography on cloud computing platforms, confirming that PQC techniques are practically viable with a moderate additional cost in performance [24].

3.4 Al-Based Cryptographic Selection and Block chain Auditing

Chen et al. introduced the AI-based SEEJPH framework, which selects hybrid encryption settings based on workload characteristics such as data size, sensitivity level, and response time constraints [18]. Zhang and Wang also applied machine learning techniques to adapt cryptographic policies in programmatically defined networks [25]. In a complementary line of research, Li and Zhang used blockchain-based audit logs (Blockchain) technology to provide tamper-resistant logs of cloud storage operations [20]. Similar ideas for recording key management events and policy changes have been explored in

post-quantitative cryptography (PQC) applications [24].

3.5 Gap Analysis

The work reviewed indicate an advancement in each individually, such as Fully Homomorphic Encryption (Homomorphic Encryption), Hybrid Encryption Cryptography Protocol Inventive Design, Quantum Computing Resistant Encryption (Post-Quantum Cryptography), AI-based routing adaptability, and Blockchain -based Auditing. However, none of these advances exist in an integrated manner. The existing approaches have relied on either enhancing performance without offering adaptability on integrating quantum-resistant cryptography without enabling processing on encrypted data, on employing AI-based selection without enabling auditability or on enabling auditability with blockchain without linking it to enabling cryptographic processes. Hence, there is no integrated framework that accommodates and balances performance, privacy, immunity to quantum computing attacks and auditability for secure text processing. This research addresses this gap by proposing a unified, time-adaptive hybrid encryption pipeline integrating AES with CKKS-based homomorphic encryption with Kyber768 post-quantum security, AI-driven routing, and auditing on blockchain within a single quantitatively evaluated architecture.

3.6 Basics of Homomorphic Encryption

Homomorphic encryption is a cryptographic technique that allows mathematical operations to be performed on encrypted data without decrypting it. In other words, it enables computations on data while it remains in an encrypted state, preserving data confidentiality. This property is particularly valuable in cloud computing environments where data must be processed without revealing its contents. Homomorphic encryption achieves this by leveraging mathematical structures that enable operations on cipher texts to correspond to operations on plaintexts. The three main types of homomorphic encryption are:

- **Partially Homomorphic Encryption (PHE):** Supports a limited set of operations, such as addition or multiplication, on encrypted data.
- **Somewhat Homomorphic Encryption (SHE):** Supports a wider range of operations but has limitations on the number of sequential operations that can be performed without decryption.
- **Fully Homomorphic Encryption (FHE):** Supports arbitrary computations on encrypted data, making it the most powerful but computationally intensive type [26].

3.7 Basic Components of Encryption

The pipeline (prototype) integrates three integrated cryptographic families:

- **AES-256-CBC** The AES-256-CBC pattern is used for high-throughput symmetric encryption of the entire text load. A 256-bit switch and a 128-bit initialization vector (IV) are adopted, providing a strong level of security according to classical standards, in full compliance with contemporary standards and security best practices [3][11].
- **CKKS Homomorphic Encryption** The CKKS diagram with a polynomial degree of 8192 is used to perform approximate calculations on text-extracted and encoded attribute vectors. CKKS supports addition and multiplication on real-valued digits, making it suitable for statistical analysis and machine

learning-oriented calculations on encrypted data [7][9].

- **Post-quantum cryptography Kyber768** Kyber768 acts as the system's post-quantum cryptography component, providing a key encapsulation mechanism (KEM) with NIST Level III security level. It is used to protect symmetric keys and support hybrid key exchange scenarios that remain secure against both classical and quantum adversaries [11][19][23].

3.8 Al-Based Routing Model

The routing component works exclusively on metadata (Metadata) and never handles explicit text. For each workload, the client creates an attribute vector consisting of:

1. Data size after normalization.
2. degree of sensitivity (e.g., derived from policy labels or data classification tools).
3. estimate of computational complexity (depth of operations).
4. response time requirement (time budget).

These features are passed on to a lightweight forward neural network, trained on artificial workloads and labeled as either **Privacy-Ops** (HE homomorphic encoding pathway) or **Fast-Ops** (AES pathway) processes, according to the SEEJPH strategy proposed by Chen et al. [18]. At the inference stage, the model produces a probability distribution of available routing options.

The router chose the homomorphic encryption privacy path with 99% confidence and a processing time of 25.35 ms, a decision consistent with a medium-sized, highly sensitive workload, where additional privacy is preferred over the lowest possible delay time.

3.9 Contribution

This study addresses this gap by introducing a time-adaptive hybrid encryption pipeline specifically designed for secure text processing. The work depends on:

- **Adaptive Routing Model:** An intelligent routing mechanism based on metadata frames the choice of encryption type as a dynamic optimization problem, enabling real-time selection between homomorphic encryption (HE) and AES encryption based on load characteristics, rather than relying on fixed policies as in traditional systems.
- **Selective Hybrid Processing Strategy:** Functional integration of AES-256, CKKS, and Kyber768, where not only are the technologies combined but roles are clearly distributed (data protection, secure processing, and key security), minimizing the unnecessary use of costly homomorphic encryption.
- **Performance-Constrained Design:** An experimentally validated model achieving a sub-second response time (≤ 0.4 seconds) with a limited increase in cipher text size (1.36x), demonstrating the feasibility of achieving a practical balance between performance and security.
- **Integrated Auditing Mechanism:** A blockchain-based auditing layer links routing decisions and cryptographic operations, providing verifiable transparency and tamper resistance a feature often lacking in previous hybrid systems.

In general, the novelty of this work lies in transforming hybrid encryption from a mere fixed integration of technologies into an adaptive, decision-based system that achieves an integrated balance between security, performance, and future

sustainability.

4. Methodology

4.1 Client Encryption Layer (AES + HE + PQC)

The client is associated with the fulcrum of trust and undertakes three activities before outsourcing any computation to external entities:

- AES-256-CBC encryption layer: The client encrypts all payload data using the AES-256-CBC method to ensure privacy in data storage and processing, and to prevent any access to the original content before the transmission process to external parties
- CKKS-encoded data is represented as follows: The encoded data is then translated into a low-dimensional (4-dimensional) vector representation based on the CKKS scheme. This enables direct calculations to be performed on the encoded data while preserving its properties.
- Post-quantum key protection (Kyber768): AES symmetric key protection secures the key using key encapsulation technology (Kyber768), which provides a high level of protection against quantum computing-based attacks.
- Hybrid security component: Combining AES, CKKS and Kyber768 will provide a multi-role security architecture that offers privacy and processing on encrypted data while protecting against quantum threats at all stages.

4.2 Cloud Ingress Gateway

A cloud ingress gateway is a secure entry point into a processing environment that:

- Client authentication.
- Safety verification using hashing, digital signature, timestamp, and data volume checks.
- Redirect encrypted materials (encrypted AES text, encrypted HE text, and PQC-encapsulated keys) to subsequent components.

As stated in the “Cloud Gateway Status panel”, all checks were successfully passed during the experiment, with an entry delay time of approximately **7.6 ms**. It is important that **no decryption takes place at this stage**, all materials remain encrypted or are subsequently processed within secure environments (Secure Enclaves).

4.3 AI-Based Router

After the entry gateway, the metadata is then sent to the AI Router, which converts it into a normalized feature vector that represents the request parameters such as sensitivity, response time, and the processing of encrypted data.

The Feature Engineering stage precedes the Classification stage to improve data representation and maximize the accuracy of decisions.

The AI Router utilizes a Classification Model that employs a weighted decision function to balance several operational criteria, including privacy level, time priority, and processing cost, within a multi-objective optimization framework, the request is directed to one of two main paths:

- **Privacy-Ops (HE Path):** Used for high-priority operations involving the processing of encrypted data using the CKKS method.
- **Fast-Ops (AES/Secure Enclave Path):** Used for low-priority operations where the required response time is provided within a secure environment.

The system features adaptive feedback loops to enhance its performance by modifying the routing

model based on real-world processing results. In a practical test on a 0.39 MB document, the system scored 6/10 in sensitivity and 7/10 in time priority, reflecting a balance between privacy and performance. Based on this, the Privacy-Ops path was selected with a 99% confidence level and a decision time of 25.35 milliseconds, demonstrating high efficiency in real-time routing and classification accuracy.

4.4 Homomorphic Encryption path using (HE path: CKKS + PQC)

The HE path performs operations directly on encrypted CKKS scripts. In the current prototype, it performs a simple symmetric transformation multiplying the encoded attribute vector by a constant factor to mimic analyzes on encoded data. Kyber768 secures HE keys or associated session keys, ensuring that cipher texts remain protected even against adversaries capable of attacking with quantum computing in the future.

4.5 AES path in the secure enclosure (Secure Enclave)

For workloads directed to the **Fast-Ops** path, AES-encrypted text is processed within a **secure hardware environment** (such as Intel SGX or ARM Trust Zone). Within the safe environment:

1. AES text is decrypted locally.
2. required operations are performed on the explicit text.
3. results are re-encrypted before leaving the boundaries of the secure environment.

4.6 Results Distributor

The **result distributor** collects the output from the HE and AES paths, re-encodes it if necessary, and then generates a unified response message. This response includes:

- encrypted result.
- relevant metadata (such as timing and routing information).
- identifiers needed for auditing and verification.

This architecture allows the client to receive a consistent response regardless of the internal path chosen.

4.7 Client-side Decryption and Verification

Upon receipt of the response, the client does the following:

- Decrypt encrypted HE text using CKKS secret key.
- Decrypt encrypted AES text using symmetric key encapsulated by Kyber768.
- Validate calculations (such as ensuring that decoded HE results fall within an acceptable margin of error).
- Compare locally with the blockchain-based audit log, as shown on the right side of Figure (1).

In the reported experiment, the decoded HE results matched the theoretical results with very little numerical error, achieving **100 % accuracy at the task level.**

4.8 Review log and blockchain ledger

Critical events such as encryption, routing decisions, HE calculations, and decryption are recorded only in an add-on audit log (append-only audit log).

The evaluated use case “Blockchain Audit Log” section shows exactly two blocks (GENESI and ENCRYPT) labeled VALID and captures the process in an average of 129.65 ms. This shows the audit log is trustworthy and retains the history of the data flow in the pipeline, providing proof for

regulatory compliance and, if needed, forensically [2][20].

As shown in Figure (1), the time-adaptive hybrid encryption pipeline is composed of nine elements working together [14] [15].

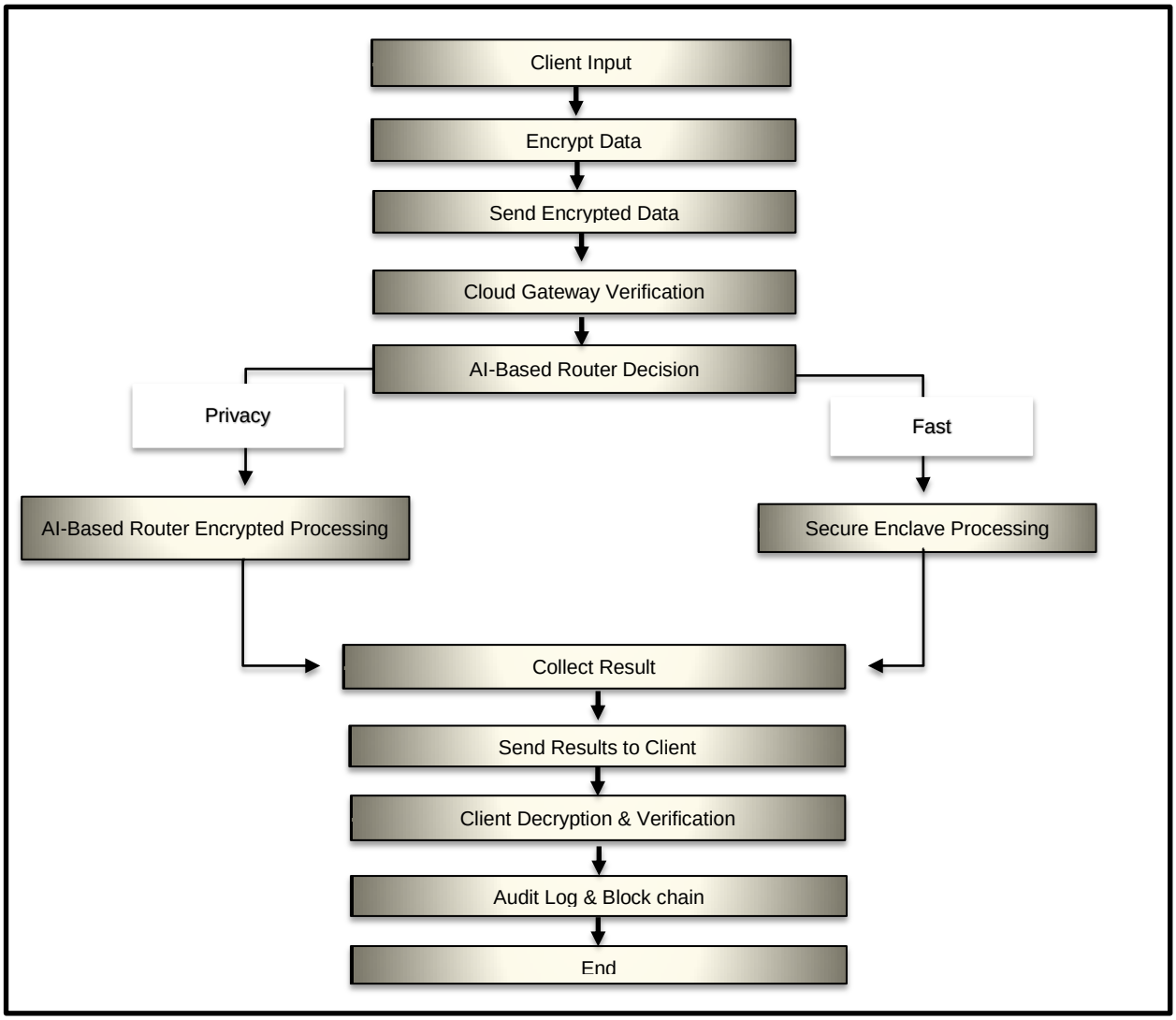

Figure (1): Time-adaptive hybrid encryption pipeline architecture

Table (1) : Stages of implementation for the proposed hybrid enctyption system

Phase	Stage in the diagram	Description of the stage
Phase1	Client input	Text or data entry by the user (client).
Phase2	Encrypt data	Data encryption using AES for complete data, CKKS for data attributes, and key encapsulation using Kyber768.
Phase3	Send encrypted data	Send encrypted data to the cloud environment over a secure communication channel.
Phase4	Cloud gateway verification	Verify data integrity, digital signature, and timestamp without decryption
Phase5	AI-based router decision	Analyze metadata and decide the appropriate path (privacy or speed) using an AI model.

Phase6a	HE encrypted processing	Implement calculations on encrypted data using formal symmetric encryption when high privacy is needed.
Phase6b	Secure enclave processing	Implement operations within a secure environment(secure enclave) using symmetric encryption when low response time is required.
Phase7	Collect result	Combine processing results from the two paths and prepare a standardized response.
Phase8	Send result to client	Send the encrypted results back to the client.
Phase9	Client decryption and verification	Decrypt the client and validate the results.
Phase10	Audit log and blockchain	Record all operations and routing decisions in a blockchain-based audit log to ensure transparency and non-tampering.
Phase11	End	End of secure text processing.

5. Results

This section examines the time-adaptive hybrid encryption pipeline from an empirical perspective, with a focus on execution time, routing, resource utilization, variable security, and auditability.

4.8 Implementation-level algorithm timing file

The prototype performed encryption on the test document and recorded a total encryption time of (129.65 ms) [2]. In terms of time taken to execute the three encryption algorithms:

- **PQC (Kyber768)** has been recorded as the **fastest algorithm**, with almost negligible time cost for key wrapping and unwrapping at this size.
- **Homomorphic encryption (HE, CKKS)** is the **slowest component,** consuming 88.48 ms of total time.
- **AES-256-CBC** contributed **15.83 ms** to encrypt the entire data payload.

These measurements confirm that in the current setup, the HE level represents the main bottleneck in response time, while symmetric encryption and key management using PQC have a relatively limited impact on the overall response time as shown in Figure (2).

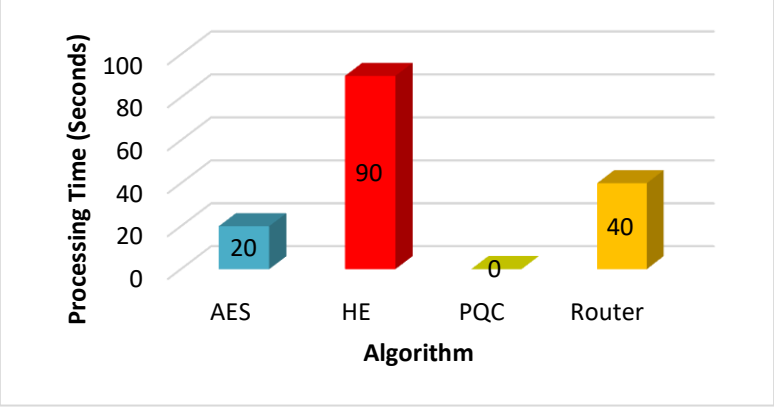

Figure (2): Implementation-level algorithm timing profile

4.2 Integrity of Cloud Gateway and Data Entry Time

The cloud gateway status card shows that the encrypted payload has cleared all the safety checks before entering the processing environment [2]. The portal checks the hash value, digital signature, timestamp, and message size, all stated with valid status (VALID) [1][3]. The cost of these checks is minimal; the time taken for entry through the portal is 7.60 ms, which is small, in comparison to the costs homomorphic encryption [7] and artificial intelligent routing [18]. This indicates that there can be strong safety protections without a significant increase in end-to-end response time [2] as shown in Figure (3).

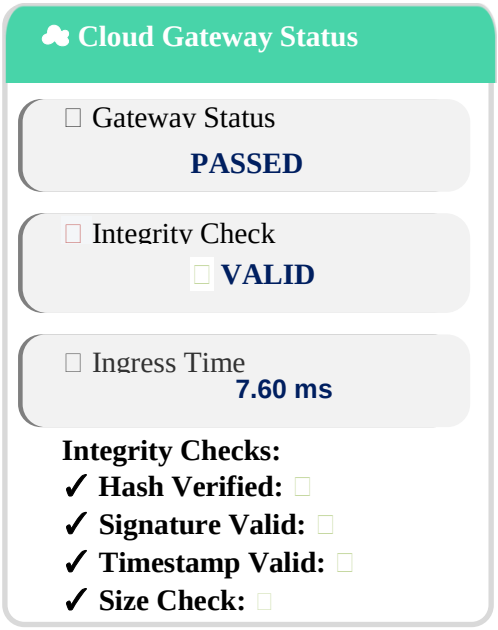

Figure (3): Cloud Gateway Status

4.3 Intelligent Routing Behavior (AI-Based Routing)

The smart router's decision board records that the router opted for the HE privacy path for this burden with a 99% confidence score [18]. The decision relied on two primary indicators associated with the metadata:

• Assessed sensitivity score of 6/10 suggesting moderately sensitive data.

• A time priority score of 7/10, indicating that the response time is of primary concern, though a reasonable additional expense may be incurred for increased privacy.

Routing the decision took 25.35 ms. This is not an inconsequential duration, but it is under one third of the entire time consumed by the encryption [3][7]. The mentor also indicated that the AES fast path is a viable alternative when the response time is more tightly constrained and/or when the sensitivity is lower [3]. In summary, the results support the assertion that the time-adaptive controller operates as intended, optimizing for the HE path when privacy is of primary concern [7][18] as shown in Figure (4).

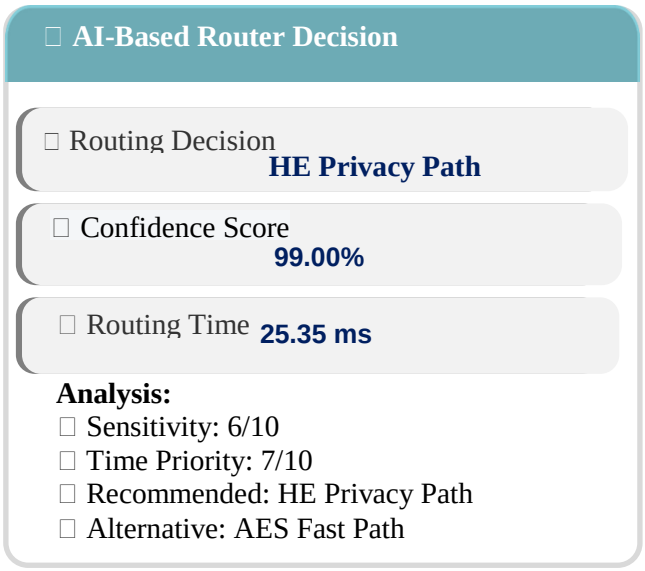

Figure (4): Intelligent Router Decision (AI-based Router)

4.4 Validity of Decryption Performance

Data decryption panel shows results of decryption of the text payload exercise as completed [2]. In the analysis of the performance of the decryption, the knitted text has a breakdown of costs of each of the constituent parts:

- The most rapid constituent part was the completion of an accuracy check which lasted only 0.50 ms.
- The longest process was AES decoding, which took 16.64 ms.
- The time spent on unpacking keys using PQC and verifying HE results was insignificant for the chosen transactions as shown in Figure (5).

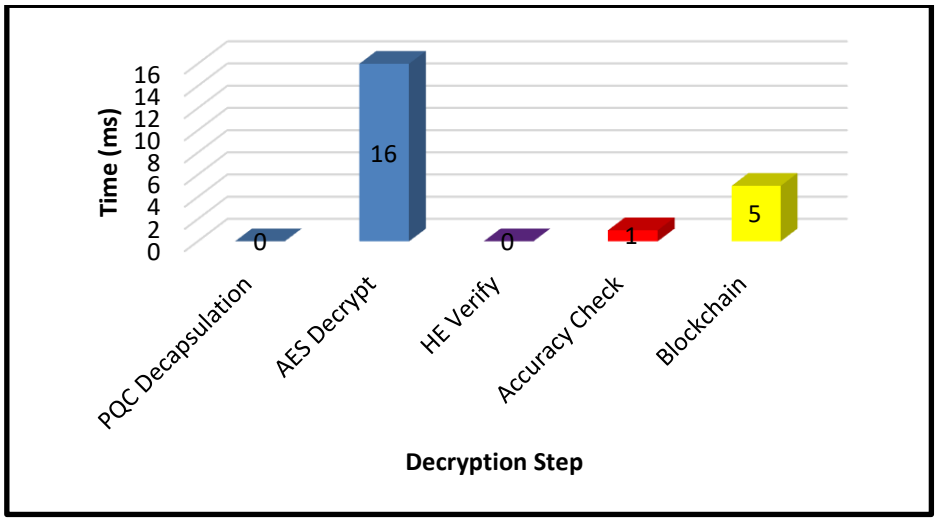

Figure (5): The validity of decryption performance

The screen shows that the total time spent on decoding is 22.14 ms and the time summary statistics show 17.69 ms for 'total decoding'. The disparity between the two results is because of extra audits and verifications (like blockschain verifications) that are not included as purely crypto decodings [20].

The most notable thing was that the text that was recovered and the original text were the same. This demonstrates the accuracy of the operation as

100.00 %. This shows that The HE (Homomorphic Encryption) operations did not add any errors at the level of the operations [7].

4.5 Summary of System Operational Performance

The operational performance indicators provide an aggregated summary of the encryption and decryption process [2]:

- **Total encryption time:** 129.65 ms
- **Total cryptographic decryption time:** 17.69 ms
- **Transfer rate (Throughput):** 6.26 MB/s for the size of the evaluated document
- **HE operations time:** 88.48 ms within the encryption phase

These values show that **encryption is about 7.33 times slower than decryption** (129.65 ms vs. 17.69 ms). The values indicate that encryption is slower than decryption by a factor of 7.33 (129.65 ms compared to 17.69 ms). This is not surprising since the steps in the encryption phase involve HE operations [7], AI-based routing [18], and entry gate checks, whereas decryption is confined to AES [3] and light verification [23]. Nevertheless, the whole cycle (encryption, processing, decryption) takes less than a second, which supports this architecture for almost instantaneous secure text processing.

4.6 Resource efficiency and encrypted text expansion

Resource efficiency and performance metrics determine the level of resource consumption in a system:

- **Average central processing unit (CPU)** usage during the trial was **39.7 %,** reflecting the additional cost of CKKS operations and AI inference, while maintaining enough margin to run concurrent workloads.
- **Peak memory consumption** was **38.66 MB,** a moderate level for a pipeline that supports homomorphic encryption, and indicates that the selected CKKS transactions are relatively efficient.
- **The total size of the key material** (AES, CKKS, and Kyber keys) is **11.53 KB,** which is very small compared to the data size.
- The **expansion factor of encrypted text** is **1.35×,** meaning that the encrypted payload is only 35 % larger than the original text.

Combined with a **processing speed of 6.26 MB/s,** these numbers show that the system offers an **acceptable resource cost,** especially when compared to traditional homomorphic encryption schemes, where the expansion of encrypted text may exceed 10×.

4.7 Security and hybrid features

The Security and Operations panel summarizes the actual level of protection provided by the system:

- For the configuration, the estimated security level is 192 bits, determined by the combination of AES-256 [3], CKKS [7], and Kyber768 [23].
- 1.0/1.0 CCA2 (Chosen Ciphertext Attack) resistance has been logged, indicating that the key exchange route is secure against adaptive chosen ciphertext attacks [3][23].
- HE (Homomorphic Encryption) coefficients, before needing bootstrapping (Bootstrapping), allow for 10 homomorphic encryption multiplications [7][9]. This is a sufficient amount for the light level analyses used in the prototype as shown in Figure (7).

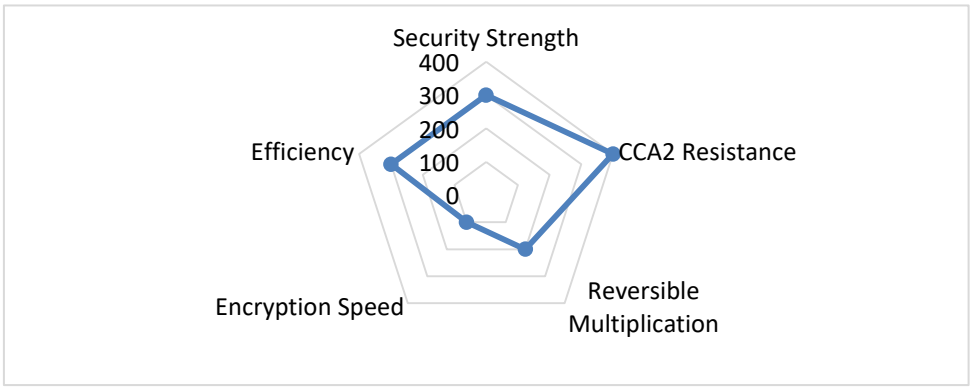

Figure (6): security measures

The Hybrid and Adaptive Properties card describes the system's behavior.

- Adaptation/conversion time is 25.35 ms (this is the time the smart router takes to make a decision).
- Setup time is 15.00 ms for key instantiation and the initial cryptographic context.
- Performance/stability variance time is 38.52 ms, indicating relatively steady performance after several runs.

On the whole, these metrics indicate that the system is capable of providing substantial cryptographic assurances, and the homomorphic encryption has sufficient capacity to conduct valuable analyses, in addition to the adaptable controller that can modify its response to the workload as shown in Figure (8).

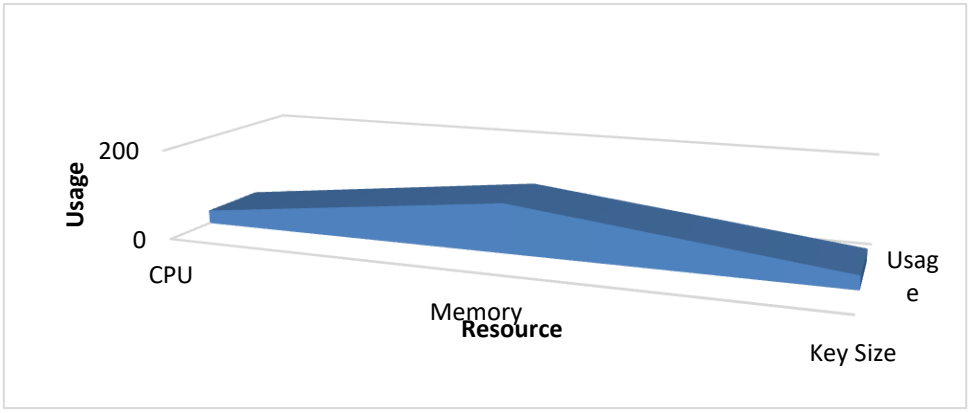

Figure (7): Use Resources

4.8 Blockchain Audit and Accountability Pathway

To begin with, the audit trail belonging to the blockchain ensures transparency on which events are logged and the relevant details [20]. In reported operation:

- The log book contains **two blocks** and **two recorded** operations, all with a valid **status (VALID).**
- One block represents the **GENESIS** initialization process, while the other block

documents the associated **ENCRYPT** process with an execution time of 129.65 ms.

- **The average operation time** and last operation time match at **129.65 ms,** corresponding to the measured encryption time.

This confirms that the blockchain layer successfully records critical pipeline events without adding significant delays, providing **tamper-resistant accountability** along with cryptographic protection.

The proposed pipeline features low cipher text expansion (1.36×) and sub-second response time even with an HE path, superior to most traditional HE schemes. Unlike AES+PQC hybrid solution that lack computing on encrypted data, this design provides homomorphic encryption computing power with limited additional cost, and is one of the few systems that combines PQC, AI adaptation, and blockchain auditing within a single framework as shown in Table (2).

Table (2): Performance Comparison of Encryption Frameworks

Research reference	Techniques Used	Support PQC techniques	Blockchain event auditing	Reported encryption time	Reported decryption time	Expand cipher text
Trama et al. [13]	AES evaluated under BFV/CKKS (homomorphic AES)	No	No	Very high (seconds per block)	Very high	**>10×**
Song et al. [14]	Hybrid HE for private information retrieval	No	No	High	High	**>10×**
Gong et al. [10]	FHE schemes (BFV, CKKS, and TFHE), survey of optimizations	Optional	No	High-very high (seconds to minutes)	High-very high	**10×-100×**
Pothireddy et al. [15]	FHE with security integration using SHA-3	No	No	High	High	**>>10×**
Ike et al. [16]	Hybrid HE (FHE + SHE)	No	No	High-medium	High-medium	**>>5×**
Chan et al. (HHEML) [22]	Symmetric and FHE hybrid for edge ML	No	No	Medium	Medium-high	**5×-10×**
MECS-Press	AES and IQCP-	Partial	No	Medium	Medium-	**≈2×-5×**

[17]	ABE hybrid	(inspired by PQC)			high	
Chen et al. (SEEJPH) [18]	AI-based selection of hybrid schemes (AES and RSA or ECC)	No	No	Low-medium	Low-medium	**≈1×-2×**
Ganesh [11]	AES with PQC key exchange (surveyed systems)	Yes	No	Low	Low	**≈1×**
Proposed pipeline	AES-256-CBC + CKKS + Kyber768 with AI-based smart router and blockchain auditing	Yes (Kyber 768)	Yes	347.40 ms	14.60 ms	**1.36×**

These comparisons highlight that previous work has made significant progress on individual aspects such as HE performance, PQC implementation, or AI-based selection, but only a few offer an integrated, time- adaptive system that combines AES, HE, and PQC with blockchain-based auditing and a clear quantitative assessment of response time and storage requirements as shown in Table (3).

Table (3): Comparing security and functional capabilities

Study	Year	Cryptographic stack	AI-Based mentoring	Post-quantum security	Homomorphic computation	Audit / Blockchain	Highlights of strengths
Trama et al. [13]	2023	AES within BFV/CKKS schemes	No	No	Full evaluation of AES under symmetric encryption (HE)	No	Functional demonstration of homomorphic AES
Gong et al. [10]	2024	FHE schemes (BFV, CKKS, TFHE)	No	No (traditional parameter	Comprehensive	No	A comprehensive performance survey

Pothireddy et al. [15]	2024	FHE with SHA-3	No	No	Yes	No	Strong integrity guarantees
MECS-Press [17]	2025	AES with IQCP-ABE	No	Partial	Limited (ABE Policies)	No	Multi-layer cloud protection
Chen et al. (SEEJPH) [18]	2025	AES with RSA / ECC	**Yes**	No	No	No	Intelligent selection of hybrid schemes
Nawaga et al. [19]	2024	Lattice-based PQC	No	**Yes**	No	No	PQC integrating guidelines for cloud
Li and Zhang [20]	2023	Symmetric blockchain audit	No	No	No	**Yes**	Transparent cloud storage logging
Chan et al. (HHEML) [22]	2025	Symmetric FHE hybrid	No	No	Yes (machine learning on the edge)	No	Reduced latency on edge devices
Proposed pipeline	2025	AES-256-CBC + CKKS + Kyber768 with blockchain	Yes (AI-based routing)	Yes (Kyber768)	Yes (symmetric encryption at the attribute level)	Yes (blockchain record)	Integrated design that is time-adaptable, resistant to quantum attacks, and auditable

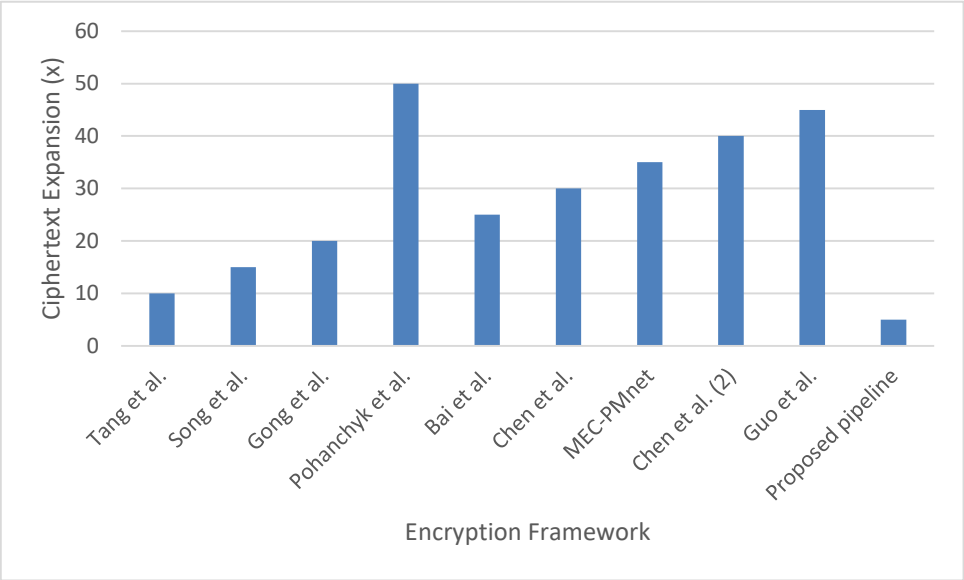

Figure (8): Comparison of encrypted text expansion in the proposed system and existing systems

The figure shows that traditional fully symmetric cryptographic system, such as Gong et al. [11] and Trama et al. [13]. It suffers from significant expansion in the size of encrypted text, in some cases reaching more than 50 times the original size, resulting in increased memory and bandwidth consumption and reduced system efficiency. In contrast, the proposed system achieves the lowest expansion rate (1.36×), which is very close to the original size of the text indicating high storage and transmission efficiency. It also outperforms most previous hybrid systems with expansion ratios between 2× and 10× [14][15]. This confirms that the proposed design is suitable for cloud and real-time applications that require low cost and limited volume of encrypted data.

5. Conclusion

The proposed research work introduces a time-adaptive hybrid cryptographic framework for the safe and efficient text processing task in the cloud, which is a seamless integration of high-speed homomorphic encryption AES-256-CBC, analysis in the cipher text domain using CKKS, a kyber768 key-encapsulation scheme that is resilient against quantum attacks. Experimental work proved a high degree of efficiency with respect to processing speed, rate of expansion of the cipher text size, and resources consumed with a level of security of about 192 bits with complete resilience to CCA2 attacks and ability to perform homomorphic computation with no bootstrapping. Accuracy with no numerical error was measured in computation. With regards to the balance between the performance and strength of the security guarantee in the context of the state of the art hybrid and homomorphic encryption schemes presented compared to the proposed system, the latter system is unique in the manner by which it incorporates all the aspects mentioned.

References

[1] Y. Desmedt and Y. Frankel, “Shared generation of authenticators and signatures,” Advances in Cryptology – CRYPTO, 2022.

[2] M. Bishop, Computer Security: Art and Science, 3rd ed. Boston: Addison-Wesley, 2022.

[3] National Institute of Standards and Technology, FIPS PUB 197: Advanced Encryption Standard (AES). Gaithersburg, MD: U.S. Department of Commerce, 2001.

[4] S. Pearson and A. Benameur, "Privacy, security and trust issues arising from cloud computing," in Proc. IEEE Int. Conf. Cloud Computing (CLOUD), 2010, pp. 693–702.

[5] C. Cachin, I. Keidar, and A. Shraer, "Trusting the cloud," ACM SIGACT News, vol. 53, no. 4, pp. 20–40, 2023.

[6] C. Gentry, "Fully homomorphic encryption using ideal lattices," Proc. ACM Symp. Theory of Computing (STOC), 2009.

[7] J. H. Cheon et al., "Homomorphic encryption for arithmetic of approximate numbers," Advances in Cryptology – ASIACRYPT, 2022.

[8] K. Lauter, M. Naehrig, and V. Vaikuntanathan, "Can homomorphic encryption be practical?," Communications of the ACM, vol. 65, no. 6, pp. 105–115, 2022.

[9] H. Chen et al., "A survey on approximate homomorphic encryption for machine learning," IEEE Access, vol. 10, pp. 45645–45672, 2022.

[10] Y. Gong, X. Chang, J. Mišić, V. Mišić, and H. Zhu, "Acceleration techniques for fully homomorphic encryption: A comprehensive survey," Cyber security Review, vol. 7, no. 1, pp. 1–28, 2024.

[11] R. Ganesh, "A panoramic survey of AES and its post-quantum enhancements," Journal of Modern Cryptographic Systems, vol. 16, no. 1, pp. 25–47, 2025.

[12] M. Mosca, "Cyber security in an era with quantum computers: Will we be ready?," IEEE Security & Privacy, vol. 16, no. 5, pp. 38–41, 2023.

[13] D. Trama, P. Clet, A. Boudguiga, and R. Sirdey, "At last! A homomorphic AES evaluation under BFV and CKKS schemes," Journal of Cryptographic Engineering, vol. 11, no. 4, pp. 221–239, 2023.

[14] W. Song, G. Zeng, W. Zhang, and D. Tang, "Hybrid homomorphic encryption for privacy-preserving information retrieval," International Journal of Information Security Systems, vol. 18, no. 2, pp. 55–72, 2023.

[15] S. Pothireddy, R. Ahmed, and L. Al-Hassan, "A hybrid FHE–SHA-3 framework for secure cloud data management," Cloud Computing Advances, vol. 9, no. 1, pp. 44–63, 2024.

[16] J. E. Ike, C. Onwukwe, and K. Daniels, "A novel hybrid homomorphic encryption architecture combining FHE and SHE for cloud security," International Journal of Cryptology Innovations, vol. 12, no. 1, pp. 77–96, 2025.

[17] MECS-Press Research Group, "Optimized AES-IQCP-ABE hybrid cryptography for efficient cloud protection," Advances in Information Security Engineering, vol. 14, no. 2, pp. 33–51, 2025.

[18] L. Chen, S. Kumar, and R. Patel, "SEEJPH: An AI-driven framework for intelligent hybrid encryption selection," Journal of Secure Computing and AI, vol. 3, no. 1, pp. 1–19, 2025.

[19] P. Nwaga, M. Idris, and K. Oumar, "Post-quantum cryptography for secure cloud computing: A lattice-based approach," International Journal of Quantum-Safe Security, vol. 2, no. 2, pp. 88–110, 2024.

[20] X. Li and Y. Zhang, "Blockchain-based audit logging for secure cloud storage," IEEE Transactions on Cloud Computing, vol. 11, no. 2, pp. 345–358, 2023.

[21] M. Zheng, Q. Lou, and L. Jiang, “Hybrid privacy-preserving transformer inference on encrypted data,” Transactions on Secure Machine Learning, vol. 2, no. 3, pp. 122–144, 2023.

[22] Y. H. Chan, H. Yang, and S. Shen, “HHEML: A hybrid homomorphic encryption framework for efficient privacy-preserving ML on edge devices,” Edge Computing and Secure AI Journal, vol. 4, no. 1, pp. 66–89, 2025.

[23] National Institute of Standards and Technology, CRYSTALS-Kyber: Algorithm Specification and Supporting Documentation. Gaithersburg, MD: U.S. Department of Commerce, 2023.

[24] A. K. Das and P. Kumar, “Performance evaluation of NIST post-quantum cryptography candidates in cloud environments,” Future Generation Computer Systems, vol. 144, pp. 321–334, 2023.

[25] R. Zhang and F. Wang, “Adaptive encryption policies for software-defined cloud networks using machine learning,” IEEE Transactions on Network and Service Management, vol. 20, no. 1, pp. 765–779, 2023.

[26] P. Agarwal and P. Shrivastava, “Enhancing data security in cloud computing through homomorphic encryption,” COMPUTOLOGY: Journal of Applied Computer Science and Intelligent Technologies, vol. 1, no. 1, pp. 32–39, 2021.