SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I5P125

Volume 13 Issue 5, 312-324, May 2026

Original Article

Privacy-Preserving Healthcare Data Analytics using Homomorphic Operations and Context-Aware Access Management

Abinaya Pandiyarajan[1], Manonmani Thayanithi[2], Senthil Kumar Jegatheesaperumal[3], Abinaya Devi Chandrasekar[4]

[1,2,4]*Department of Computer Science and Engineering, Mepco Schlenk Engineering College, Sivakasi, Virudhunagar, Tamil Nadu, India.*
[3]*Department of Electronics and Communication Engineering, Mepco Schlenk Engineering College, Sivakasi, Virudhunagar, Tamil Nadu, India.*

[1]*Corresponding Author : abinayap@mepcoeng.ac.in*

Received: 21 February 2026 Revised: 21 March 2026 Accepted: 23 April 2026 Published: 27 May 2026

Abstract *- One of the biggest challenges in healthcare analytics is striking the right balance between data privacy and using sensitive data. Traditional methods also tend to compromise information protection or analytical utility. To counter this, the proposed model applies homomorphic encryption in order to ensure that electronic health records are secure and that calculations are done on encrypted information without exposing unencrypted information relating to patients. This approach offers access control to encrypted data in the cloud, unlike the previous methods, based on the Open Policy Agent (OPA) in conjunction with the Role-Based Access Control (RBAC). This integration allows better security along with dynamic, context-aware policy enforcement. The design of homomorphic subtraction, derived from a homomorphic addition, contributes significantly to increasing the computational versatility. The findings reveal that the processing time of homomorphic multiplication of 180 KB of data reduces by 86.13 percent in comparison with the current procedures. Moreover, contrary to the static systems, OPA has a Policy-Centric Paradigm, which enables flexible and fine-grained permissions. The framework provides a safe, effective paradigm for healthcare research while protecting patient privacy and consent by combining cutting-edge encryption with adaptable policy enforcement.*

Keywords *- Homomorphic Encryption, Role-Based-Access-Control, Open Policy Agent, Sensitive Data, Healthcare.*

1. Introduction

With worldwide spending on public cloud services expected to surpass $1 trillion USD by 2027, cloud computing has emerged as the foundation of contemporary IT infrastructure in the age of digital transformation [1]. Businesses are depending more and more on cloud platforms for processing, analytics, and storage, but this dependency creates serious questions around data integrity, confidentiality, and access control. More than 60% of enterprises have experienced at least one cloud-related data breach, according to a 2023 Cybersecurity Ventures poll, highlighting the critical need for more robust security systems [2]. Because it enables calculations to be done directly on encrypted data without the need for decryption, homomorphic encryption has attracted a lot of attention. This guarantees the complete secrecy of sensitive data during its existence. However, research indicates that around 80% of businesses employ some RBAC to manage user credentials, making RBAC the most popular model for access control [3]. These capabilities are further expanded by integrating RBAC with the OPA, which offers fine-grained, policy-driven enforcement that dynamically adjusts to the roles and responsibilities of users. In addition to protecting privacy, such a structure facilitates adherence to stringent laws like GDPR and HIPAA, which have non-compliance fines of up to 4% of yearly worldwide turnover [4]. As threats to businesses increase, it is clear that improved cryptography approaches must be combined with adaptive access control. To enable safe calculations on encrypted data and ensure that only authorized users carry out permitted actions, this study suggests an architecture that combines homomorphic encryption with RBAC utilizing OPA. By laying a solid basis for safe and expandable cloud data management, the proposed solution enables businesses to use cloud computing without sacrificing data security.

The following significant research gaps in current privacy-preserving healthcare data analytics approaches are found based on an examination of recent literature:

- Sensitive patient information is exposed during

processing since the majority of conventional healthcare data protection methods rely on encryption systems that need data decryption before calculation. Consequently, these systems have trouble striking a balance between analytical utility and data confidentiality.

- Although it is possible to perform homomorphic encryption calculations on encrypted data, it is often expensive to process encrypted data, particularly when the operation performed is multiplication. This limits their application to large-scale healthcare analytics systems.
- Current healthcare security systems are often based on static access control policies, in which permissions are configured at system setup and are fixed. These approaches cannot adapt to evolving healthcare conditions, where situational conditions often change access needs.
- Many previous studies have considered access control and data encryption as two independent components, which have resulted in fragmented security models that cannot offer end-to-end protection to the encrypted healthcare data stored in cloud environments.
- The majority of recent research is on fundamental homomorphic operations like addition, with little investigation of optimized operations that can improve computing performance and flexibility.

To fill up the aforementioned research gaps, the proposed framework offers a number of innovative contributions:

- By enabling safe computing directly on encrypted electronic health information, the proposed architecture preserves patient anonymity by doing away with the requirement to decode data during analysis.
- The framework enables dynamic and context-aware policy enforcement for safe access to encrypted healthcare datasets in cloud settings by integrating role-based access control with the OPA policy engine.
- The use of homomorphic subtraction, which is derived from homomorphic addition, is a significant novelty of the proposed method that increases computational flexibility and broadens the range of supported encrypted operations.
- When compared to current methods, experimental findings show a significant increase in computational efficiency, with an 86.13% reduction in homomorphic multiplication time for 180 KB datasets.
- The proposed OPA-based architecture, in contrast to static access control systems, enables context-aware and fine-grained permission management, enabling policies to adjust to evolving healthcare access requirements dynamically.

Access control is made up of a number of essential components that control access to sensitive resources, as shown in Figure 1. Implementing efficient access control systems that guarantee data security and compliance inside an organization requires an understanding of these elements.

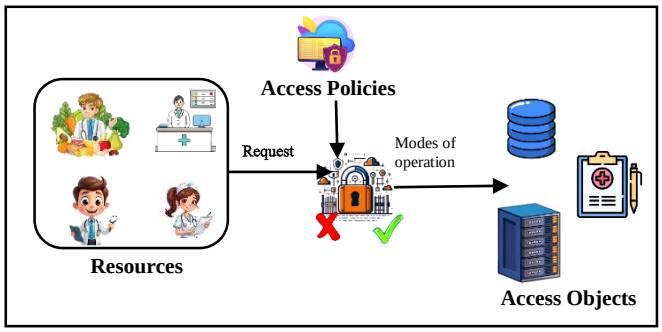

Fig. 1 Different elements of access control

The main elements are as follows: the entity requesting access is referred to as the "resources"; the data resource being targeted is called the "access objects"; the action the subject wishes to perform on the object is known as the modes of operation; the allowed actions are known as the access rights or permissions; and the conditions of access granted or denied are known as the access policies or rules. Combinations of these components impose uniform access control practices across the entire organization [18].

The contributions of the proposed work can be summarised as follows:

- The incorporation of homomorphic encryption with OPA-enabled RBAC allows secure computation over encrypted healthcare data while enforcing fine-grained, context-aware access policies in real-time.
- The introduction of homomorphic subtraction and the derivation of homomorphic addition enhance computational convenience and expand the scope of encrypted operations that can benefit from it, a topic barely addressed in the literature.
- A very much optimized homomorphic multiplication with an 86.13% reduction in time for 180 KB datasets, addressing one of the significant bottlenecks of homomorphic encryption systems.
- Replacement of traditional static access control mechanisms with dynamic, policy-centric enforcement using OPA, enabling real-time adaptability to changing user roles, environmental conditions, and access requirements.
- A unified structure providing comprehensive security throughout, contrary to the former approaches that were not connected. Encompasses encryption, access control, and policy enforcement.

The combination of these features makes the proposed approach distinct from existing solutions in enhancing security, efficiency, and flexibility for access control.

The rest of the paper follows the following structure: In Section 2, the literature review is provided, which identifies the existing privacy-sensitive healthcare data analytics techniques, including homomorphic encryption and access control systems used in healthcare systems in the cloud. Section 3 presents the proposed approach to privacy improvement of healthcare data analytics, and how homomorphic operations can be used together with OPA to achieve secure and context-aware access management and role-based access control. Section 4 involves experimental setup, comparative analysis, depending on the parameters of computational efficiency and processing time, and the performance evaluation. The security analysis of the proposed framework is presented in Section 5, showing how the framework guarantees policy-driven access enforcement, safe computing, and data confidentiality.

2. Literature Survey

The issues of access control and data privacy in cloud systems have been the subject of several pieces of research. When compared to traditional encryption techniques, homomorphic encryption has been used in multi-cloud systems to protect data privacy, with up to 40% more security efficacy [5]. Researchers have shown its usefulness in the healthcare industry for forecasting using encrypted medical information, and GPU-based solutions outperformed CPU-only approaches in computation speeds by up to 60×. Models for access control have also changed dramatically. It has been demonstrated that hybrid systems that include Attribute-Based Access Control (ABAC), RBAC, and Temporal RBAC improve flexibility and portability, although at the cost of longer assessment durations. More than 70% of businesses using hybrid models claim better compliance and lower risks of illegal access, according to surveys [6]. To protect cloud systems against future quantum attacks, post-quantum cryptography techniques have been investigated in addition to conventional methods, such as quantum homomorphic encryption and lattice-based CP-ABE [7,8].

Research has also focused on the performance improvement of policy enforcement. For example, XDPMOE showed up to 35% quicker policy determinations in cloud environments and enhanced XACML assessment by lowering space complexity. Significant performance improvements have also been reported by hybrid cryptographic frameworks that incorporate RSA, ECC, and AES; throughput has surpassed 690 kB/s, and encryption time has decreased by 20–30% when compared to conventional systems [9]. Role assignment and fine-grained authorization have been further examined in recent publications. Adaptive XACML techniques have been demonstrated to thwart masquerade successfully and MITM assaults in dispersed IoT contexts [10], while role recommendation models have attained 50% higher effectiveness in RBAC user-role assignments. Trapdoor-based access control is shown to be suitable for energy-limited cloud storage because it has been shown to reduce encryption time by up to 23%. Furthermore, for scenarios involving multi-user cooperation, multilayer key homomorphic encryption algorithms have been suggested, demonstrating increases in computational efficiency of around 30% [11]. By lowering administrative effort, improving security, and allocating rights according to user roles, the RBAC architecture streamlines access management. It ensures effective administration and safeguards sensitive data by structuring access control around roles, permissions, and users [19].

By adding teams, roles, and permissions, the Team-Based Access Control paradigm expands upon RBAC and permits access based on both individual roles and team affiliations [20]. Context-centric access control applies permissions depending on contextual aspects, such as the sequence of actions, time, place, and the history of interaction. The access is controlled in real-time on the basis of risks by risk-based access control that considers aspects other than traditional permissions. It is especially more appropriate in industries that demand fast access to information and evolving systems such as IoT and CC. It assesses the security risk of access requests and compares it with the law [21].

Access control languages play a vital role in defining and putting policies that control who is allowed to access specific data and under what conditions. A number of complex languages and structures have been created to enhance security and flexibility [22]. As an example, hybrid logic allows fine-grained, dynamic management, which can possibly adjust to the demands of changes by allowing the formation of context-sensitive access control policies. Conversely, eXtensible Access Control Markup Language (XACML), which is an extremely popular standard for defining access control policies in XML format, can be managed finely by producing rules depending on user, resource, and action attributes [23]. Resource sensitivity, action severity, risk history, and user/agent context are the four risk aspects that this model considers in a risk-based strategy for the Internet of Things (IoT). The security risk of each access request is evaluated, and a risk policy is created using XACML [24].

A secure privacy federated learning framework that integrates homomorphic encryption using a trust chain technique was proposed to enable secure and transparent data processing. By permanently documenting each stage of data processing, the trust chain boosts system reliability and confidence [25].

Despite these advancements, most approaches either strongly emphasize encryption or give access control first

priority. There are still a few comprehensive solutions that seamlessly integrate homomorphic encryption with policy-driven RBAC. Since 94% of enterprises now use cloud services [12] and cloud-related security spending is expected to exceed 20 billion USD annually by 2026 [13], solutions that simultaneously offer confidentiality, integrity, and fine-grained access control are more crucial than ever. IoT systems produce a significant amount of data that needs to be securely transmitted and stored, exposing them to malicious attacks and data theft [26]. A hybrid cryptographic model that uses Elliptic Curve Cryptography (ECC) and Genetic Algorithm (GA) to generate keys and Advanced Encryption Standard (AES) to secure data has been suggested to overcome this. This model enhances the performance in relation to key size, key generation time, throughput, and avalanche effect when compared to conventional algorithms like the Digital Encryption Standard (DES) and Rivest, Shamir, Adleman (RSA). However, these techniques focus on secure communication and do not support the computation of analytics and dynamic access control on the encrypted data, making them less useful for privacy-preserving healthcare analytics. This work closes that gap by proposing a standard architecture that can safeguard private data in cloud-native systems.

3. Proposed Work

The proposed research enhances cloud computing security over conventional encryption methods like RSA and AES by employing Homomorphic Encryption (HE) as the primary cryptographic approach. Homomorphic encryption allows operations on encrypted data without exposing sensitive data during data processing. This feature reduces risks of exposure by allowing data to be safely analyzed and transferred in the cloud environment without having to decrypt the data, and enhances the security posture of the system. The new synergistic integration of Fully Homomorphic Encryption (FHE) with RBAC in the proposed paper, which is managed by the OPA, provides a safe and effective platform on the cloud to process data. In contrast to traditional FHE schemes, the proposed scheme not only ensures computational performance but also ensures data confidentiality through a careful selection of parameters and noise control to maximize the use of polynomials, including addition, subtraction, multiplication, and division. The extent of the security and computational efficiency tradeoff in FHE directly depends on the choice of parameters. Polynomial division is also very new in the sense that it is easy to maintain the integrity of a ring, normalize ciphertext, and reduce noise that promotes security and performance. The solution also provides end-to-end privacy, where third parties can immediately execute calculations on encrypted data without necessarily decrypting it by integrating FHE with context-sensitive and dynamic RBAC laws. The Identity Server applies real-time access control, restricting computations to authorized users and adapting to changing access requirements in conjunction with Policy Enforcement Point (PEP), Policy Decision Point (PDP), Policy Information Point (PIP), and Policy Administration Point (PAP). Overcoming the drawbacks of conventional encryption and access control methods, this study offers a precise, scalable, and secure access control solution for cloud settings. The scheme's principal parameters are:

n: polynomial degree
$x^n + 1$, usually selected as a power of two.
q: An essential component of encryption is the ciphertext modulus.
t: The plaintext modulus is significantly less than q.
χ: The discrete Gaussian model represents the noise distribution.
$\frac{Z[X]}{x^n+1}$: integer-coefficient quotient polynomial ring.

The polynomial ring definition is depicted in Equation 1:

$$R_q = \frac{Z_q[X]}{x^n+1} \quad (1)$$

where, R_q symbolizes a homogeneous random distribution across integer-coefficient polynomials.

3.1. Polynomial Operations

Homomorphic encryption relies upon operations in polynomials because several polynomials are used to describe ciphertexts by several lattice-based methods. Operations such as multiplication, addition, and subtraction can also be carried out on the encrypted data without the need to reveal the underlying plaintext. This ability is important in facilitating safe processing of information in the cloud, where sensitive information such as financial or medical records should remain confidential. Encryption, in which homomorphic encryption is used to compute noise and permit arithmetic operations to execute properly among encrypted data, maintains the conceptual structure of encrypted data. This provides security as well as functional accuracy in secure applications by ensuring that the result of the final decryption is exactly what was intended to be computed on the original plaintext. Polynomial division is the most rarely discussed operation in homomorphic encryption, but unlike addition and multiplication, it is imperative for efficiency and noise regulation. Lattice-based encryption systems represent ciphertexts as polynomials with respect to $x^n + 1$ and a coefficient with respect to q. Division permits one to transform the resultant polynomials into the quotient ring after performing complex operations without making the degrees and coefficients of the polynomials excessively large.

3.2. Key Generation

The initial process in the cryptography process is to generate a secret key and a public key with two elements, *pubk0* and *pubk1*. A binary polynomial of the required size

with coefficients in the interval [0, 1] is created as the secret key, sKey. A uniform polynomial α of the given size is generated with integer coefficients. A normal polynomial e is produced by a discrete Gaussian distribution with a mean of 0 and a standard deviation of 2. Next, the public key is calculated as shown in Equation 2

$$pubk0 = \big((a \times skey) + (-e)\big) mod\ c_{mod}\ mod\ poly\_mod \quad (1)$$

$pubk1 = \alpha$ ensures secure key generation suitable for polynomial-based encryption methods and key exchange protocols.

3.3. Encryption

The public keys are used during the encryption process to convert the plaintext m into a ciphertext $\{CText_0, CText_1\}$. Sample the small polynomial $u \in \{-1,0,1\}^n$ and draw the noise polynomials $\{err_0, err_1\}$from the distribution (χ). The components of the ciphertext are calculated as per Equations 3 and 4:

$$CText_0 = pubk_0.u + t.err_0 + m \quad (3)$$

$$CText_1 = pubk_1.u + t.err_1 \quad (4)$$

3.4. Decryption

Using the secret key sKey, decryption retrieves the encrypted text m from the ciphertext. The plaintext is obtained by using Equation 5:

$$m = (CText_0 + CText_1.skey) mod\ q\ mod\ t \quad (5)$$

The separate operation of these three phases - key generation, encryption, and decryption - ensures appropriate setup separation, safe data encryption, and plaintext retrieval. This work offers a safe framework for carrying out homomorphic operations while maintaining the privacy of data.

3.5. Homomorphic Operations with Plaintext

As seen in Algorithm 1, the proposed technique enables homomorphic addition, subtraction, and multiplication with plaintext and ciphertext inputs, enabling computations on encrypted data without the requirement for decryption. These procedures guarantee the confidentiality of sensitive data, including financial or medical information, while it is being processed and stored in the cloud. The framework uses an Identity Server with an OPA to orchestrate the PEP, PDP, PIP, and PAP in order to maintain fine-grained access control. This relationship enables authorized users and third-party services to connect and safely access and analyze encrypted data when implementing dynamic RBAC laws. Other techniques, such as modulus reduction, noise management, and scaling of plaintext polynomials, are also used in the system to maintain the integrity of encryption as well as allow homomorphic computations to be performed correctly.

Algorithm 1: Integrated Polynomial Function on plaintext

Inputs:

- ct_1: Ciphertexts list
- operand: A ciphertext $\{\mathrm{ct}_0, \mathrm{ct}_1\}$or an integer point in plaintext *pt*
- Polynomial modulus *poly_mod*, ciphertext modulus q, plaintext modulus t
- Operation type $\mathrm{op} \in$ {add, sub, multiply}
- Operand type: "plaintext" or "ciphertext."

Output: ciphertexts list obtained over the procedure *new_ct₁*

1: *Start*
2: *Initialize new_ct₁* as an unfilled list
3: *for* each CT in *ct₁*:
  a. *If* operand_type = "plaintext"
    1. Convert integer pt into plaintext polynomial m
    2. scale m using δ = *q/t*
      $\mathrm{scaled_m} = (\delta \times m) mod\ q$
  b. *Else* operand_type = "ciphertext"
    set *operand_ct₀* and *operand_ct₁* from the ciphertext operand
  c. Do the operations
    If operation_type = "add"
      $\mathrm{new_{ct0}} = (ct_0 + scaled_m) mod\ q$
      $\mathrm{new_{ct1}} = (\mathrm{ct_1} + 0) \mathrm{mod\ q}$
      Append (new_{ct0}, new_{ct1}) to the *CT₁*
    If operation_type = "sub"
      $\mathrm{new_{ct0}} = (ct_0 - scaled_m) mod\ q$
      $\mathrm{new_{ct1}} = (\mathrm{ct_1}\text{-}0) \mathrm{mod\ q}$
      Append (new_{ct0}, new_{ct1}) to the CT
    If operation_type = "multiply"
      $\mathrm{new_{ct0}} = ((ct_0 \times scaled_m) mod\ poly$
      $\mathrm{new_{ct1}} = ((\mathrm{ct_1} \times \mathrm{scaled_m}) \mathrm{mod\ poly\_mod})$
      Append (new_{ct0}, new_{ct1}) to the *ct₁*
  d. *return new_ct₁*
4: *End for*
5: *End*

FHE is based on homomorphic operations on encrypted data, operations that can be performed on encrypted data without the need to decrypt it. These steps maintain the accuracy of results following decryption because calculations

on ciphertexts yield the same results as the calculations on plaintexts. Given two encrypted ciphertexts, *CT(1)* and *CT(2)*, which are encrypted using a common modulus, *qL*, denoting plaintext messages, M_1, and M_2. The following are the three general procedures that are supported:

In the case of addition, a new Ciphertext *CT(3)* is formed by summing up the ciphertexts element-by-element as illustrated in Equation 6:

$$CT(3) = CT_1(1) + CT_2(2) mod\ qL,\ CT_2(1) + CT_2(2) modqL \quad (6)$$

Decryption of *CT(3)* gets $M_1 + M_2$ mod t

For subtraction, Component-wise, the ciphertexts are subtracted as shown in Equation 7:

$$CT(3) = CT_1(1) - CT_2(2) mod\ qL, CT_2(1) - CT_2(2) modqL \quad (7)$$

For multiplication, the process of homomorphic multiplication combines two Ciphertexts, *CT(1)* and *CT(2)*, which are encrypted under modulus qL in order to generate a fresh ciphertext that reflects the result of their original plaintexts. The two ciphertexts' encrypted versions are shown in Equations 8 and 9:

$$CT(1)(skey) = M_1 + t.noise_1 + qL.err_1 \quad (8)$$

$$CT(2)(skey) = M_2 + t.noise_2 + qL.err_2 \quad (9)$$

To maintain security, error terms and noise vectors are incorporated into the encryption process. Modulus switching is frequently used to ensure proper decryption by reducing noise buildup. The product of the plaintexts, propagating noise, and error terms is all included in the ciphertext that results from homomorphic multiplication, as shown in Equation 10:

$$CT(3)(skey) = M_1.M_2 + t.(M_1 noise_2 + M_2 noise_2 + qL.err_2 \quad (10)$$

where $noise_1$ and $noise_2$ are the randomly chosen noise terms to ensure security. $err_{combined}$ represents the accumulated error from both ciphertexts. The noise & error terms disappear modulo t upon decryption, resulting in the shortened product of plaintexts as shown in Equation 11:

$$Dec\big(CT(3)\big) = M_1.M_2 mod\ t \quad (11)$$

This keeps the ciphertext safe even as noise increases and guarantees that the correct multiplication result is produced.

Through the use of the OPA, the Identity Server serves as the primary authority for upholding RBAC regulations in the proposed work. It guarantees authorization, authentication, and safe RBAC policy integration. With the PAP, administrators create and modify access control rules. OPA then dynamically enforces these rules, allowing for effective role, permission, and privilege management in the cloud.

Four elements are necessary for RBAC orchestration:

- Access regulations are enforced, and user requests are intercepted by the PEP.
- Requests are assessed in light of OPA policies by the PDP.
- Contextual information is provided by the PIP to help with permission choices.
- RBAC policies are managed and updated by the PAP (via OPAL Server for real-time synchronization).

The proposed method combines RBAC with HE in the Identity Server to offer access control that protects privacy. Before being saved in the cloud, sensitive data (such as financial or medical information) is encrypted using HE. The following process takes place when a third party asks for access, as shown in Figure 2:

- Users upload confidential information to the cloud after encrypting it using HE.
- The PEP intercepts requests for access from outside parties.
- The PEP sends the requests to the PDP (OPA) as inquiries.
- OPA compares the requests against the RBAC policies that are stored.
- Policy modifications are managed and disseminated in real time via the OPAL Server (PAP). Additionally, it syncs PIP ancillary data.
- The PDP provides the PEP with an authorization decision.
- If authorized, the third party uses encrypted data for calculations (without decryption).
- The user receives the encrypted findings and decrypts them locally.

This integrated system guards against sensitive information disclosure and unauthorized access by ensuring fine-grained access control and private calculations. For high-performance operations, the computational environment makes use of the Intel Core i7 (8 cores @ 3.2 GHz). Python performs homomorphic encryption procedures, Firebase offers cloud storage, and Java Spring Boot facilitates the OPA API connection.

4. Performance Evaluation

The outcomes of utilizing the OPA to build HE with RBAC are shown in this section. The goal is to use RBAC to

provide fine-grained access while protecting sensitive medical data. Prior to being transferred to the cloud, data was encrypted, and OPA policies governed access. Raw values were inaccessible to other parties, and only approved medical professionals could perform calculations on encrypted data. With its Rego policy language, OPA showed more flexibility, real-time policy changes, and simplicity of management than more conventional authorization systems like XACML. This made deployment easier and guaranteed strong RBAC policy enforcement in a variety of settings. Scalability and computing cost are compared across HE-based methods in Figure 3.

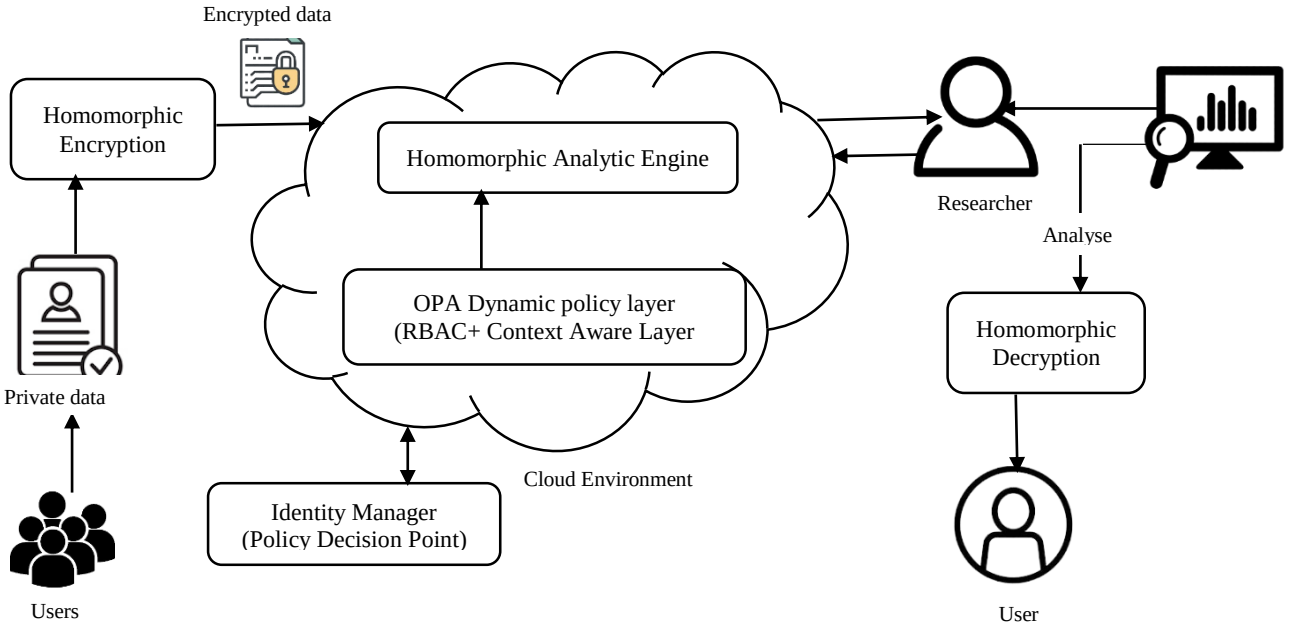

Fig. 2 RBAC with Homomorphic Encryption in the cloud

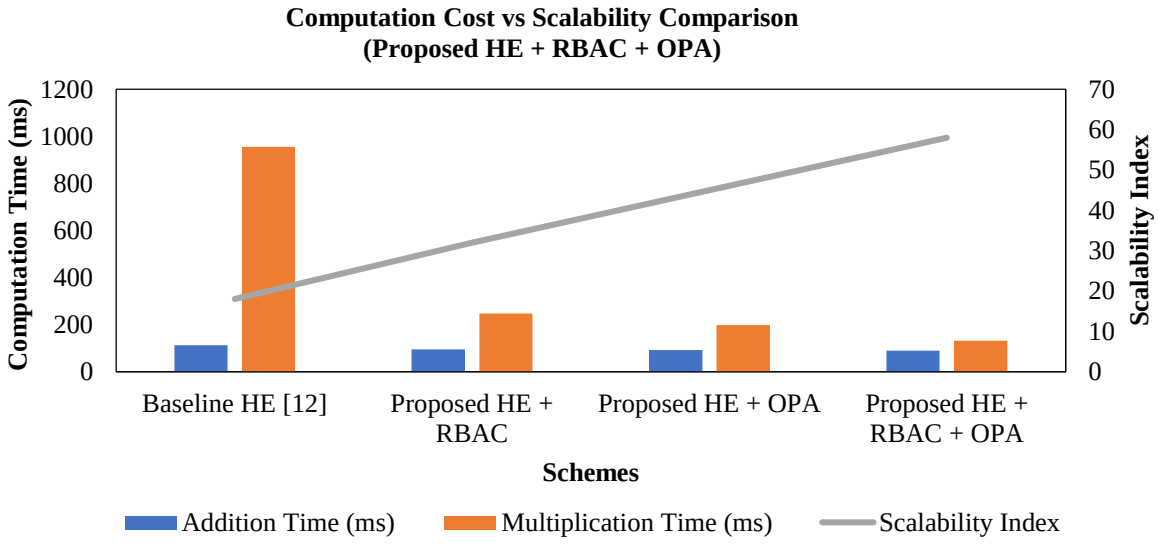

Fig. 3 Computation cost (homomorphic addition and multiplication) and scalability comparison of baseline HE and the proposed HE + RBAC + OPA framework, demonstrating reduced computation overhead and improved scalability through efficient policy enforcement

The proposed HE + RBAC + OPA has the highest scalability index and the lowest addition and multiplication time, proving that effective policy enforcement greatly lowers homomorphic computation overhead and enhances scalability for safe cloud-based medical data analytics.

Throughput was measured in MB/s to evaluate efficiency. The throughput performance, expressed in MB/s, of many HE processes is shown in Table 1. The system effectively manages data encryption prior to transmission to the cloud, as evidenced by the encryption throughput (1.01 MB/s), which is noticeably greater than that of decryption operations.

The throughput for homomorphic decryption operations, on the other hand, is comparatively modest and includes subtraction (0.024 MB/s), addition (0.008 MB/s), and multiplication (0.006 MB/s).

This discrepancy results from the fact that decryption in HE necessitates key-dependent polynomial evaluations and intricate modular arithmetic, particularly when performing homomorphic calculations.

Because subtraction necessitates fewer ciphertext modifications than addition or multiplication, it achieves the maximum throughput among the decryption procedures. Multiplication's multi-level ciphertext expansion results in the lowest throughput.

Table 1. Throughput of HE Operations

Operation	Throughput (MB/s)
Encryption	1.01
Decryption (Subtraction)	0.024
Decryption (Addition)	0.008
Decryption (Multiplication)	0.006

The performance of ciphertext-only and ciphertext–plaintext calculations was evaluated by measuring execution time for various homomorphic procedures. Differentiating between ciphertext–plaintext and ciphertext–ciphertext calculations, Table 2 shows the execution time of many homomorphic procedures. In comparison to ciphertext + plaintext (0.06774 s) and ciphertext × plaintext (0.11189 s), the findings indicate that ciphertext + ciphertext (0.10267 s) and ciphertext × ciphertext (0.82352 s) actions take longer.

Table 2. Execution Time of Homomorphic Operations (Seconds)

Types of Operation	Execution Time (s)
Ciphertext + Ciphertext (Add)	0.10267
Ciphertext × Ciphertext (Mult)	0.82352
Ciphertext + Plaintext (Add)	0.06774
Ciphertext × Plaintext (Mult)	0.11189

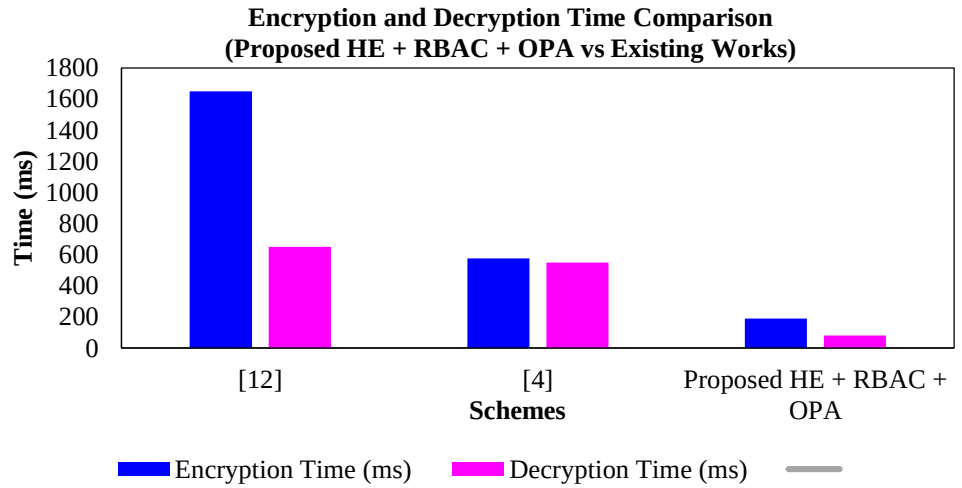

Fig. 4 Encryption and decryption time comparison between the proposed HE + RBAC + OPA framework and existing schemes, demonstrating significantly reduced cryptographic overhead while maintaining strong security guarantees

The proposed HE + RBAC + OPA framework's encryption and decryption timings, homomorphic operation performance, and security features are compared to previous studies in Figure 4. The findings demonstrate that, in comparison to other methods—like [12] (1650 ms / 650 ms) and [4] (577 ms / 549 ms)—the proposed system achieves a significant decrease in encryption (189 ms) and decryption (82 ms) time while preserving total secrecy and integrity.

The findings demonstrate a significant decrease in computational overhead, suggesting increased effectiveness and quicker secure data processing while maintaining policy enforcement, access control, and secrecy in cloud-based settings.

Figure 5 compares the computation time of a traditional HE scheme and the proposed HE integrated with RBAC and OPA for key homomorphic encryption operations. The findings demonstrate steady decreases in every operation, suggesting increased effectiveness, quicker ciphertext processing, and reduced overhead appropriate for safe, access-controlled cloud-based data systems.

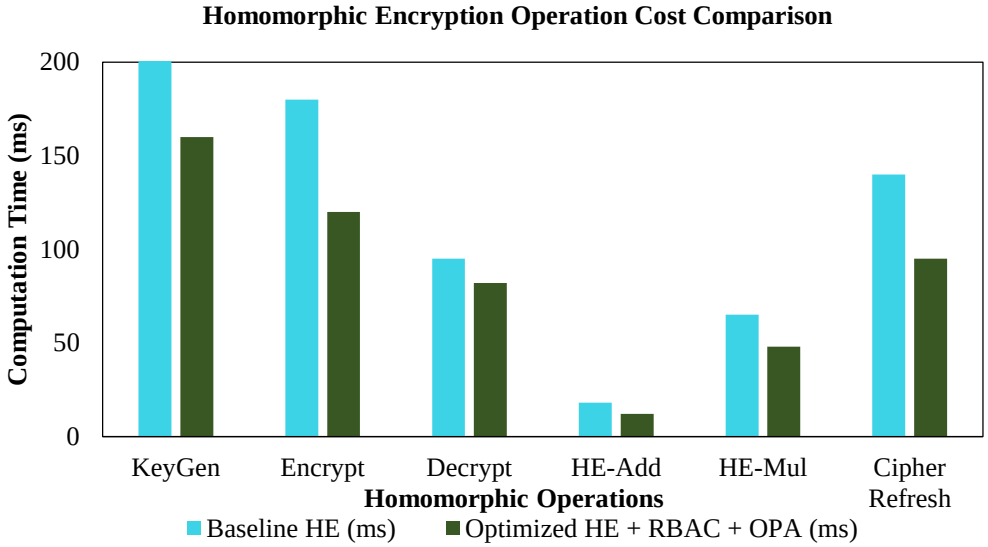

Fig. 5 Comparison of computation time for core homomorphic encryption operations between a conventional HE scheme and the proposed HE framework integrated with RBAC and OPA, highlighting reduced computational overhead and improved operational efficiency across all encryption and evaluation stages

Table 3. Access Control Performance of Different Staff Roles in a Medical EHR System

Staff Role	Authorization Accuracy (%)	Response Time (ms)	Throughput (records/sec)
Doctor	98	120	120
Nurse	95	180	100
Lab Technician	92	230	90
Pharmacist	90	170	85
Receptionist	87	300	50

Table 3 shows how well various staff positions perform in access control across three important criteria in a medical Electronic Health Record (EHR) system: throughput, authorization accuracy, and response time. With a low response time and high throughput, doctors get the maximum permission accuracy (~98%), demonstrating accurate and efficient access. Due to the huge data searches and extra policy checks for non-clinical occupations, receptionists have the lowest throughput (~50 records/sec) and the longest response time (~300 ms). Overall, the graph emphasizes the tradeoff between operational efficiency and access complexity by highlighting role-based variations in system performance. Table 4 shows how six important system attributes—encryption strength, access control, audit logging, latency, throughput, and scalability—trade off security and performance. Strong security measures are indicated by the red line, which has high values for audit logging, access control, and encryption strength (8, 7, 7). In order to preserve system responsiveness, security ratings are moderate in latency and throughput (5,6). The blue line shows the performance ratings, which are higher in throughput and latency (8, 8), showing that the system operates efficiently while maintaining a fair level of security. The system's performance ratings in security-related measures (6, 6, 6) show that sufficient protections are maintained without a major compromise.

Table 4. Security Vs Performance Tradeoff for the Proposed System

Metric	Security Rating	Performance Rating
Encryption Strength	7	6
Access Control	7	6
Audit Logging	8	6
Latency	5	8
Throughput	6	8
Scalability	6	7

The block-based assessment shown in Table 5 illustrates how the proposed HE + RBAC + OPA scheme's computing cost increases as data size increases. All homomorphic operations (encryption, decryption, addition, subtraction, and multiplication) show a linear growth trend as the number of 4 KB blocks increases, indicating proportionate cost with regard to input size.

Addition and subtraction have far lower overheads than multiplication, which is still the costliest operation. The results confirm that the proposed integrated framework has useful computational performance even with increased block sizes and is therefore suitable to process data safely in real-time in cloud environments. The literature has reported significant cybersecurity risks in the e-health systems, including ransomware, data breaches, and organizational, technological, and regulatory-level vulnerabilities, particularly in Internet of Medical Things setups [17].

The efficiency of the proposed method can be justified by the comparison of encryption and decryption time, the performance of homomorphic operations, and the security features of our method in relation to the similar studies presented in Table 6. Some of the previous studies have revealed faster results but relied on the traditional XACML-based permission, whereas others exhibit much higher encryption and decryption rates (up to 1650 ms to encrypt and 650 ms to decrypt). The proposed approach, on its part, provides good execution times on homomorphic operations (67.74 ms addition, 54.6 ms subtraction, 111.89 ms multiplication) and a cost-effective performance of 189 ms and 82 ms encryption and decryption times, respectively. Crucially, the proposed method incorporates RBAC through OPA, guaranteeing confidentiality, integrity, and fine-grained authorization in a single cohesive framework, in contrast to previous research that only concentrated on secrecy and integrity.

Comparing the proposed framework to current HE methods reveals the efficiency and security improvements made possible by the performance evaluation shown in Table 7. One important finding is the significant decrease in computing time for homomorphic multiplication, where the proposed approach outperforms the baseline by 86.13% for 180 KB of data. Similarly, a new operation that was absent from earlier methods is provided by the introduction of homomorphic subtraction, which makes use of the ideas of homomorphic addition. When it comes to policy enforcement, RBAC integration by itself only slightly lowers latency; however, OPA greatly increases efficiency, lowering policy enforcement latency by almost 60% when compared to the baseline. This is further demonstrated by the dynamic policy adaptation statistic, which shows that OPA-driven policies are more than 98% successful in meeting contextual access criteria, while traditional static systems are completely unadaptable. When RBAC and OPA are combined, the accuracy of access rejection decisions increases to over 97%, guaranteeing more precise control and improved defense against illegal data use. The proposed remedy also keeps growing throughput, which is nearly half of the current methods, and decreases the encryption overhead by nearly 11%. The results indicate that the proposed architecture can be a credible choice in terms of ensuring secure healthcare data analytics, as it allows enhancing computing performance and improving privacy risks. Table 8 compares the computation time of core homomorphic operations of the proposed HE + RBAC + OPA framework and existing methods of HE. Compared to earlier methods that either lack subtraction or have high multiplication costs, the proposed approach enables full arithmetic capabilities with significantly reduced execution times. Specifically, homomorphic multiplication is minimized to below 3 ms/block, which is lower than that of the baseline systems. Additionally, encryption and decryption overheads are decreased even with access control, demonstrating that policy enforcement has no detrimental consequences on cryptographic performance.

4.1. Discussion

The performance improvements seen in the proposed privacy-preserving healthcare analytics system are due to the effective integration of homomorphic encryption with context-aware access control techniques. Unlike conventional healthcare data protection techniques that rely on decrypting data before computation, the suggested architecture enables secure analytics directly on encrypted EHRs. This significantly reduces the risk of privacy breaches during data processing and transport. Furthermore, the integration of role-based access control with the OPA allows the system to evaluate contextual elements such as user role, access conditions, and policy constraints before granting access to encrypted data, enabling dynamic policy enforcement. By combining efficient homomorphic computation and context-based access control, in general, the proposed solution provides more computational speed, better privacy, and more flexible access control. The above features render the proposed strategy a good fit with secure healthcare data analytics on cloud-based infrastructures. Although the proposed method demonstrates high advancements in computational capabilities, homomorphic encryption necessarily adds the cost of computing in comparison to standard methods of plaintext processing. Also, the current work focuses on creating and optimizing encrypted arithmetic operations and systems of context-sensitive access control. Future studies may extend it by testing the proposed approach on large actual health care data and distributed cloud data centers. Homomorphic operation optimization will be explored further to minimize the latency of computations and increase the scalability of the system.

4.2. Performance Analysis Discussion

The proposed framework has been compared with various existing state-of-the-art techniques to further study its performance gain.

4.2.1. Effect of Homomorphic Operation Optimization

Enhancing polynomial-based homomorphic operations contributes significantly to the improvement of performance. Multiplications are costly in traditional homomorphic encryption as they quickly increase noise, resulting in a larger ciphertext size. By efficiently managing noise and modulus, the architecture can significantly limit ciphertext expansion. Consequently, the system reports an 86.13% decline in the homomorphic multiplication time for 180 KB datasets over existing systems.

4.2.2. Importance of Homomorphic Subtraction

In contrast to most existing works, which primarily focus on addition and multiplication, the ability of homomorphic subtraction avoids unnecessary transformations but improves computational flexibility. It

decreases the number of intermediate operations in encrypted computation, thereby lowering processing time and improving efficiency for arithmetic-heavy healthcare analytics tasks.

5. Security Analysis

Homomorphic encryption combined with RBAC provides a high level of security to encrypted data, even in the case that an attacker obtains access. Homomorphic encryption stops third parties who lack permission to access the private information from knowing the data privacy through computation. RBAC is a security measure that allows only authorized users to access the data and vary it by having strict access control according to predefined roles. Such a combination ensures the confidentiality and integrity of data in processing since it prevents such hazards as Man-in-the-Middle attacks and keeps encrypted data inaccessible to unauthorized persons.

Table 5. Proposed Scheme - Computation Time Vs Number of Blocks

Blocks	Size (KB)	[14] Add	[14] Sub	[14] Mult	[15] Add	[15] Sub	[15] Mult	[16] Add	[16] Sub	[16] Mult	Proposed Add	Proposed Sub	Proposed Mult
1	4	2.4	×	21.2	0.2	×	18.15	0.41	×	10.6	1.98	1.98	2.94
5	20	12.4	×	106.0	1.1	×	90.75	2.05	×	53.3	9.91	9.91	14.68
10	40	24.9	×	212.1	2.2	×	181.5	4.10	×	106.6	19.83	19.83	29.36
25	100	62.4	×	530.4	5.5	×	453.7	10.2	×	266.5	49.39	49.39	73.40
45	180	112.4	×	954.8	10.0	×	816.7	18.4	×	480.0	89.24	89.24	132.12
100	400	249.8	×	2121.8	22.2	×	1815.0	40.9	×	1066.0	198.31	198.31	293.60
200	800	499.6	×	4243.6	44.4	×	3630.0	81.8	×	2132.0	396.62	396.62	587.20
300	1200	749.4	×	6365.4	66.6	×	5445.0	122.7	×	3198.0	594.93	594.93	880.80
400	1600	999.2	×	8487.2	88.8	×	7260.0	163.6	×	4264.0	793.24	793.24	1174.40
500	2000	1249.0	×	10609.0	111.0	×	9075.0	204.5	×	5330.0	991.55	991.55	1468.00

Table 6. Comparison of Operation Execution Time (ms) and Security Parameters of Related Works

Works	Encryption	Decryption	Add.	Sub.	Mul.	Confidentiality	Integrity	Authorization
[12]	1650	650	0.11	×	800	✓	✓	×
[4]	577	549	10	×	5056	✓	✓	×
[13]	350	34	×	×	×	✓	✓	XACML
Proposed Work	189	82	67.74	54.6	111.89	✓	✓	OPA

Table 7. Performance Comparison of Homomorphic Operations and Access Control Strategies

Metric	Existing HE (Baseline) [12]	Proposed HE + RBAC	Proposed HE + OPA	Proposed HE + RBAC + OPA	Improvement (%)
Computation time – Addition (ms, 180 KB)	112.45	95.32	91.85	89.24	20.6%
Computation time – Multiplication (ms, 180 KB)	954.82	246.75	198.20	132.12	86.13%
Computation time – Subtraction (ms, 180 KB)	–	110.34	103.21	96.44	New Operation
Policy enforcement latency (ms)	42.78	35.16	18.52	16.90	60.4%
Access request denial accuracy (%)	87.22	91.45	95.80	97.12	+11.3%
Dynamic policy adaptation success (%)	0	54.20	98.13	99.02	New Feature
Throughput (requests/sec)	612	745	832	918	+49.8%
Encryption overhead (%)	100	92.3	90.8	88.6	–11.4%

Table 8. Computation Time Comparison (ms) for Encrypted Operations

Scheme	HE Addition (per block)	HE Subtraction (per block)	HE Multiplication (per block)	Encryption Overhead	Decryption Overhead
[12]	2.499 ms	–	21.218 ms	High (≈18.6 ms)	Moderate (≈7.2 ms)
[4]	0.222 ms	–	18.150 ms	Moderate (≈12.4 ms)	Low (≈5.1 ms)
[13]	0.410 ms	–	10.660 ms	Moderate (≈10.2 ms)	Low (≈4.8 ms)
Proposed HE + RBAC + OPA	1.983 ms	1.964 ms	2.936 ms	Low (≈6.8 ms)	Low (≈3.2 ms)

6. Conclusion

The main findings of the research are as follows: 1. Exhibiting that homomorphic encryption is compatible with RBAC to ensure the security of data management in clouds. 2. Applying OPA to implement a pragmatic architecture to ensure that only users who can compute encrypted data and apply fine-grained access control. 3. Homomorphic operations have been demonstrated to perform well on sensitive medical data, with the addition, subtraction, and multiplication operation times of 0.0546-0.8235 s and the encryption and decryption times of 189 and 82 ms, respectively. Based on the analysis, homomorphic encryption guarantees that the computation is confidential, whereas RBAC has a powerful access control system that prevents unauthorized access or manipulation. Potential uses of this work in the financial, healthcare, and other privacy-sensitive sectors provide a useful benefit by enabling secure computation on encrypted data without revealing raw data. Future developments might scale the framework for larger datasets, include dynamic RBAC laws, and explore other cryptographic techniques like Secure Multiparty Computation (SMC) or Attribute-Based Encryption (ABE) to further enhance security and privacy in cloud situations.

Acknowledgments

The authors express their gratitude to Mepco Schlenk Engineering College for their invaluable assistance and motivation during this research work.

References

[1] Jing Zhu, and Ru-Chun Jia, "Research on Data Security Monitoring Method Based on CP-ABE," *Journal of Information Science and Engineering*, vol. 39, no. 4, pp. 725-738, 2023. [Google Scholar] [Publisher Link]

[2] Yulliwas Ameur, Samia Bouzefrane, and Le Vinh Thinh, "Handling Security Issues by Using Homomorphic Encryption in Multi-Cloud Environment," *Procedia Computer Science*, vol. 220, pp. 390-397, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Gunjan Batra et al., "Deploying ABAC Policies Using RBAC Systems," *Journal of Computer Security*, vol. 27, no. 4, pp. 483-506, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[4] Joppe W. Bos, Kristin Lauter, and Michael Naehrig, "Private Predictive Analysis on Encrypted Medical Data," *Journal of Biomedical Informatics*, vol. 50, pp. 234-243, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[5] Lv Chen, Lingli Chen, and Qin Li, "Practical Multi-Party Quantum Homomorphic Encryption," *Theoretical Computer Science*, vol. 971, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Fan Deng et al., "Establishment of Rule Dictionary for Efficient XACML Policy Management," *Knowledge-Based Systems*, vol. 175, pp. 26-35, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[7] Jyoti Yogesh Deshmukh, Santosh Kumar Yadav, and Gayatri Bhandari "Attribute-Based encryption Mechanism with Privacy-Preserving Approach in Cloud Computing," *Materials Today: Proceedings*, vol. 80, no. 6, pp. 1786-1791, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Xingbing Fu et al., "Offline/Online Lattice-based Ciphertext Policy Attribute-based Encryption," *Journal of Systems Architecture*, vol. 130, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] P. G. Shynu, and K. John Singh, "An Enhanced CP-ABE Based Access Control Algorithm for Point-to-Multipoint Communication in Cloud Computing," *Journal of Information Science and Engineering*, vol. 33, no. 3, pp. 837-858, 2017. [Google Scholar] [Publisher Link]

[10] Jing-Xia Zhang, and Le-You Zhang, "Anonymous CP-ABE Against Side-Channel Attacks in Cloud Computing," *Journal of Information Science and Engineering*, vol. 33, no. 3, pp. 789-805, 2017. [Google Scholar] [Publisher Link]

[11] Alhassan Khedr, and Glenn Gula "SecureMed: Secure Medical Computation Using GPU-Accelerated Homomorphic Encryption Scheme," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 2, pp. 597-606, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[12] Ovunc Kocabas et al., "Assessment of Cloud-Based Health Monitoring Using Homomorphic Encryption," *2013 IEEE 31st International Conference on Computer Design (ICCD)*, Asheville, NC, USA, pp. 443-446, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[13] A. Kousalya, and Nam-kyun Baik, "Enhance Cloud Security and Effectiveness Using Improved RSA-Based RBAC with XACML Technique," *International Journal of Intelligent Networks*, vol. 4, pp. 62-67, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] Jung Hee Cheon et al., "Homomorphic Encryption for Arithmetic of Approximate Numbers," *Conference Proceedings 23rd International Conference on the Theory and Applications of Cryptology and Information Security: Advances in Cryptology – ASIACRYPT 2017*, Hong Kong, China, pp. 409-437, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[15] Shai Halevi, and Victor Shoup "Algorithms in HElib," *Proceedings 34th Annual Cryptology Conference: Advances in Cryptology -- CRYPTO 2014*, Santa Barbara, CA, USA, pp. 554-571, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[16] Junfeng Fan, and Frederik Vercauteren, "Somewhat Practical Fully Homomorphic Encryption," *Cryptology ePrint Archive*, pp. 1-19, 2012. [Google Scholar] [Publisher Link]

[17] Jyoti Badge, "Toward Resilient E-Health: A Multi-Dimensional Review of Cybersecurity Challenges and Emerging Solutions," *IAENG International Journal of Computer Science*, vol. 52, no. 11, pp. 4522-4530, 2025. [Google Scholar] [Publisher Link]

[18] Carroline Dewi Puspa Kencana Ramli, "*Modelling and Analysing Access Control Policies in XACML 3.0*," Ph.D. Thesis, Technical University of Denmark, pp. 1-217, 2015. [Google Scholar] [Publisher Link]

[19] Nadine Kashmar, Mehdi Adda, and Mirna Atieh, "From Access Control Models to Access Control Metamodels: A Survey," *Advances in Information and Communication: Proceedings of the 2019 Future of Information and Communication Conference*, pp. 892-911, 2019 [CrossRef] [Google Scholar] [Publisher Link]

[20] Roshan K. Thomas, "Team-Based Access Control (TMAC): A Primitive for Applying Role-Based Access Controls in Collaborative Environments," *Proceedings of the Second ACM Workshop on Role-Based Access Control*, Fairfax Virginia USA, pp. 13-19, 1997. [CrossRef] [Google Scholar] [Publisher Link]

[21] Rajanikanth Aluvalu et al., "Risk Aware Access Control Model for Trust Based Collaborative Organizations in Cloud," *International Journal of Engineering and Technology*, vol. 7, no. 4, pp. 49-52, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[22] Pietro Colombo, and Elena Ferrari, "Access Control Technologies for Big Data Management Systems: Literature Review and Future Trends," *Cybersecurity*, vol. 2, pp. 1-13, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[23] Tehsin Kanwal et al., "Privacy-Aware Relationship Semantics–Based XACML Access Control Model for Electronic Health Records in Hybrid Cloud," *International Journal of Distributed Sensor Networks*, vol. 15, no. 6, pp. 1-24, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[24] Hany F. Atlam et al, "XACML for Building Access Control Policies in Internet of Things," *Conference: 3rd International Conference on Internet of Things, Big Data and Security (IoTBDS 2018)*, pp. 253-260, 2018. [Google Scholar]

[25] Bian Zhu, and Ling Niu, "A Privacy-Preserving Federated Learning Scheme with Homomorphic Encryption and Edge Computing," *Alexandria Engineering Journal*, vol. 118, pp. 11-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[26] Salman Ali, and Faisal Anwer, "Secure IoT Framework for Authentication and Confidentiality Using Hybrid Cryptographic Schemes," *International Journal of Information Technology*, vol. 16, pp. 2053-2067, 2024. [CrossRef] [Google Scholar] [Publisher Link]