International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V73I5P115

Volume 73 Issue 5, 161-172, May 2025

Original Article

Block-Chain Based Multi-Layer Encryption Method for the Secure Storage of Healthcare Data in Cloud Environments

Suresh. R[1], T. R. Nisha Dayana[2]

[1,2]*Department of Computer Science, Vel's Institute of Science, Technology and Advanced Studies Chennai, Tamil Nadu, India.*

[1]*Corresponding Author : sureshphd2023@gmail.com*

Received: 14 February 2025 Revised: 02 May 2025 Accepted: 19 May 2025 Published: 31 May 2025

Abstract *- Remote diagnostic processes in electronic healthcare require patient health information. Hospitals, healthcare facilities, and insurance companies must all securely exchange this medical data as the demand to fully utilize them develops. However, maintaining the ownership and integrity of this data is difficult due to third-party access and potential manipulation. Standard cryptography systems may handle a variety of issues, including multi-tenant setups, secure key management, and real-time data processing. By providing a multi-layered, machine learning (ML) based dynamic encryption solution intended for cloud environments, this contemporary cryptographic technique aims to overcome these issues. The proposed approach combines homomorphic encryption (HE), attribute-based encryption (ABE), and blockchain-assisted key management (BAKM) into a single framework. The framework utilizes digital contrasts to facilitate access control, and data can only be accessed and interpreted by authorized entities. The recommended method provides a complete solution for protecting medical data in cloud environments while permitting safe and easy data sharing between patients, researchers, and hospitals, notwithstanding the privacy trade-off. A secure and reliable digital healthcare environment is made possible by this strategy.*

Keywords *- Advanced Encryption Standard (AES), Attribute based Encryption (ABE), Blockchain (BC), Homomorphic Encryption (HE), Machine Learning (ML), Multi-layer Encryption.*

1. Introduction

Modern technology is currently playing a major role in the transition from traditional healthcare to smart healthcare systems. Data confidentiality is a crucial requirement for contemporary security systems everywhere. Significant vulnerabilities continue to exist because of insecure and insufficient access management, even after different access-control rules have been put in place to improve system security [1]. Managing enormous amounts of data, such as individual reports and photos, increases the demand for human labour and poses security threats. There are several challenges in preventing cybercriminals from accessing sensitive and private patient data.

Thus, it is critical to consider security when storing, retrieving, and sharing patient medical data in the cloud to prevent data compromise by authorized users of e-healthcare systems. So far, a number of cryptographic methods have been developed to enable safe medical data access, exchange, and storage in cloud service providers [2]. Distributed networking and cloud computing have revolutionized the IT sector, and when combined with intelligent systems, they will completely change the healthcare sector. Given the volume of data created by gadgets, it would be wiser to store it in a cloud environment. Healthcare data security and privacy are still issues, though, because medical information is very private and must be protected. The BC is a potential distributed network that guarantees data security and authenticates and transparently conducts transactions [3].

1.1. Cryptography and Blockchain in Cloud

Cloud computing's growing popularity in the healthcare sector has many advantages, including scalability, cost-effectiveness, and easy access to patient data. It has, therefore, also created significant obstacles to protecting the confidentiality and security of private medical data. Financial loss, harm to one's reputation, and eroded patient trust are just a few of the dire outcomes that can result from breaches or illegal access to sensitive data. Encryption technologies and BC are essential in this regard for resolving these issues. Encryption is being used to avoid unauthorized access to and review of health information. During this procedure, the data being transferred gets modified, requiring the use of a decryption key to unlock it. Thanks to encryption, a wide range of organizations, including hospitals, health care insurers, and schools and universities, may send data securely

while maintaining anonymity. Since encrypted data cannot be recovered without the right keys, the danger of data theft is decreased even in the event that the data gets accessed [4].

BC resolves the need for any authoritative figure by decentralizing the information across various nodes. This elevates data resilience by removing a single point of failure. The blocks forming the chains are linked to each other, and they are cryptographically secured. Medical records are unchangeable; thus, their legitimacy is guaranteed. BC enables each transaction to have a distinct, accessible, and verifiable record.

This skill is required to guarantee responsibility and openness in the handling of medical data. According to Habib et al. (2022), smart contracts are agreements that automatically execute themselves when a certain set of conditions is met, and these play a great role in facilitating secure sharing of information among systems in the healthcare sector [5].

Despite advances in blockchain-embedded cryptographic methods for securing health data, much of the work focuses on independent methods without assessing interoperability, scalability, and real-time application in multi-stakeholder environments. There is still little consensus, standardized frameworks that can manage performance, privacy, and security across diverse health data sources and platforms. BC and encryption technologies work together to create a strong foundation for the security of healthcare data. While encryption protects data's secrecy, blockchain guarantees data's integrity and traceability. This combination offers a complete solution for safe cloud storage of healthcare data by successfully reducing common dangers like insider attacks, data manipulation, and unauthorized access [6]. The major contributions of the study are as follows.

- Construct a multi-layer encryption system using HE, AES and ABE that protects private health information while it is being stored and transferred in cloud settings.
- Deploy blockchain technology to ensure data integrity and reliability by keeping an unchangeable, tamper-proof record of all healthcare data exchanges.
- Implement a decentralized, blockchain-based authentication system that ensures that only authorized individuals can access specific encrypted data layers and blocks unauthorized access.

The paper's remaining content is arranged as follows: The relevant research is reviewed and discussed in Section 2. The proposed method and the technologies utilized are demonstrated in Section 3, which also offers a thorough overview of the multi-layered approach and BC technology. Section 4, the outcomes are discussed, and thorough performance assessments demonstrate the efficacy of the methodology. The conclusion summarises the knowledge acquired and suggests areas for further study in Section 5.

2. Literature Review

A review of research suggestions that pertain to this contribution, emphasizing their approaches and conclusions, is presented below. Ferik et al. (2024) present a unique method for shielding medical images from unwanted third-party interventions by combining BC technology, digital watermarking, symmetric encryption, and compression [7]. Kanagasabapathi et al. (2024) propose an innovative strategy to access rights management that works with the Ethereum BC technology.

An extended salp swarm optimization (ISSO) is employed to provide the required critical random number for secret key generation to increase security. The research also uses two additional algorithms, the Paillier Federated Multi-Layer Perceptron (PF-MLP) model and the HE Standard (HES), to further mitigate the risk of privacy exposure from the original health tweet dataset. The research suggests the need to develop and assess the framework with the most robust limitations and security effectiveness to help protect the health tweets dataset in such a way that they remain completely confidential [8].

A BC-based Internet of Things (IoT) may save costs and improve patient care by better utilizing medical resources, as shown by Rastogi et al. (2024) [9]. The information will be sensed and encrypted using Diffie-Hellman Galois–Elliptic-curve cryptography (DHG-ECC). The Sand Cat Optimization Algorithm (PCC-SCOA), which is based on the Pearson Correlation Coefficient, will then be used to select the best features from the resulting attributes. The selected optimal features will then be combined and hashed using the Digit Folding–Streebog Hashing approach. A Smart Contract will be made out of this hash code.

According to Siyal et. al. (2024) [1], multi-participant environments that require BC Systems can securely exchange information with the integration of IABHE+DSNN, which utilizes an Identity and Attribute Based Honey Encryption (IABHE) algorithm with Deep Spiking Neural Networks (DSNN). With this approach, the security of data is guaranteed by integrating multiple features and entities. The approach combines a multitude of security features and identities to protect data, using the MapReduce framework alongside IABHE for encryption and DSNN for key generation.

Kaur et al. (2024) propose a new framework that allows sensor equipment to produce data that is saved in the cloud and allows BC to protect transactions. The framework encrypts the block hashes using DNA cryptography. While ensuring the privacy and confidentiality of health data, the proposed framework would improve the authenticity and integrity of interactions between patients, doctors, and insurance companies [3]. A blockchain technology-based decentralized identity management system is developed and integrated in the SAP systems. Rath et al (2024) [10] come up

with a cutting-edge solution for a SAP-based healthcare system data security and authentication by combining blockchain and encryption technologies. Every member of the healthcare organization's staff, including administrators, physicians, and patients, is given a unique digital identity. These identities are protected with base 64 activities using DocuSign security features found in the SAP platform. A security study that examines the accessibility of data, confidentiality, and integrity, all of which are mentioned above, was used to assess the efficacy of the suggested solution.

A new BC technology for cloud-based health data security is presented in the Verma et al. (2024) paper. It helps to ensure authentication and offers medical records with integrity. Here, a blockchain with optimal encryption is deployed using an improved Blowfish model that guarantees authentication characteristics. The best keys are also generated using a novel technique called Elephant Herding Optimization with Opposition-based Learning (EHO-OBL). As a result, the developed method maintains the data's integrity, and ultimately, numerous measurements show how much better the offered method is [11].

According to Chirra et al. (2024), a flexible access control framework that respects the highest standards of patient data privacy and data sharing interoperability permits stakeholders to exchange sensitive information [12]. Zhang et al. (2022) recommend a BC-based e-health system for patient privacy and EHR security. Pairing-based cryptographic techniques have been used in the technology to create secure EHR data and allow patients to participate in transactions that BC distributes. This can enable traceability and prevent unwanted changes to the patients' electronic health records. Further, BC-based smart contracts are used to provide safe payment systems that let patients and the hospital consistently pay for storage and diagnostic services, respectively [13]. Singh et al. (2024) propose a unique approach that separates data into sensitive and non-sensitive parts, adds noise to sensitive data, and then utilizes ML, k-anonymization, and differential privacy to accomplish classification tasks. It allows various owners to share their data in the cloud environment for a range of purposes. The model defines the process for communication between the many untrusted parties that handle the data of the owners. The suggested model offers a reliable system that protects real data [14]. Table 1 provides a list of existing studies related to the present study.

Table 1. Existing related studies in security mechanisms in cloud environment

Author(s) & Year	Method(s)	Strengths	Limitations
Ferik et al., 2024	Blockchain, Digital Watermarking, Symmetric Encryption, Compression	Secures medical images from third-party access; layered protection mechanisms	May increase computational overhead and processing time
Kanagasabapathi et al., 2024	Ethereum Blockchain, ISSO, PF-MLP, HE Standard	Strong access management, enhanced key generation, ML-enhanced privacy	Framework complexity and performance on large datasets are not fully validated
Rastogi et al., 2024	DHG-ECC, PCC-SCOA, Digit Folding–Streebog Hashing, Smart Contract	Optimized feature selection, secure IoT-cloud integration for healthcare	Real-time application feasibility and performance under stress are not discussed
Siyal et al., 2024	IABHE + DSNN, MapReduce Framework	Secure multi-user environment; advanced encryption and dynamic key generation	High system complexity, potential challenges in DSNN deployment
Rath et al., 2024	Blockchain, Decentralized Identity System, DocuSign, SAP Integration	Strong identity authentication and data security in enterprise SAP systems	Limited generalizability outside SAP or proprietary platforms
Verma et al., 2024	Blockchain, Enhanced Blowfish, EHO-OBL	High encryption performance, key optimization, and strong authentication	Complexity of optimization method; limited info on real-world testing
Kaur et al., 2024	Blockchain, DNA Cryptography	Ensures integrity and privacy in sensor-based healthcare data	DNA cryptography is still experimental and lacks standardized implementation
Chirra et al., 2024	Blockchain-Based Flexible Access Control	Supports secure and interoperable data sharing among stakeholders	Specific cryptographic mechanisms and performance metrics are not described
Zhang et al., 2024	Blockchain, Pairing-Based Cryptography, Smart Contracts	Guarantees EHR security, enables traceability, and secure payments	Pairing-based crypto can be computationally expensive
Singh et al., 2024	ML, k-Anonymization, Differential Privacy	Protects sensitive data, supports classification and sharing among multiple parties.	Potential trade-off between data utility and privacy due to noise addition

Emerging studies review the integration of blockchain with encryption methods for healthcare data privacy. Methods such as DNA cryptography, digital watermarking, and symmetric encryption maintain patients' information confidentiality and protect images from infringement. Robust high-level access management, identification processes, and differential privacy-supporting privacy-perventing features employ k-anonymization and smart contracts to enhance the sharing of information and verification of users. Key generation and feature selection optimization algorithms such as EHO-OBL and ISSO are optimized. In addition, pairing-based cryptography and federated learning are used for secure distributed data. These systems work to ensure the confidentiality, integrity, and accessibility of the data. Still, they are mostly tested in standalone environments, and they leave scope for improvement in interoperability and system-wide standardization.

3. Materials and Methods

The suggested BC-Based Multi-Layer Encryption Method with ML for Dynamic Adaptation aims to safely store healthcare data in cloud environments while identifying and adjusting to real-time data sensitivity. Figure 1 illustrates the general procedure of the recommended approach to patient data security.

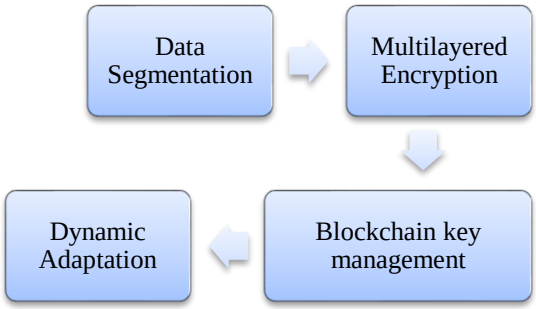

Fig. 1 Overall workflow of multi–layered encryption

3.1. Data Segmentation

Dividing the dataset into pertinent sections according to criteria such as sensitivity, utilization, or access needs is known as data segmentation for patient data. Next, give each segment a label according to the data's sensitivity [15].

3.1.1. Identify Data Attributes

Examine the patient data to better understand its form and characteristics. Typical characteristics include:

- Personal Identification data like names, addresses, and social security numbers.
- Medical information includes diagnoses, treatments, and medical history.
- Billing information includes insurance details and payment records.
- Operational data includes appointment calendars and provider notes.
- Segment data: Segment data according to sensitivity and administrative requirements:
- Highly sensitive data includes personally identifiers such as Social Security numbers.
- Critical medical history (e.g., genetic information and mental health data).
- Moderately sensitive data includes general medical records such as non-critical test findings, and billing and insurance information.
- Low sensitivity data includes operational data, such as appointment scheduling or anonymised records.

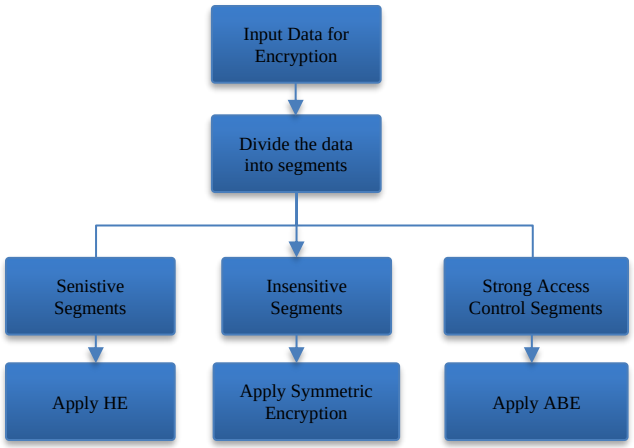

Fig. 2 Data segmentation in multilayered encryption

3.1.2. Design Segmentation Criteria

Select parameters depending on the organizational objectives and regulatory needs. Table 2 shows the segmentation for a healthcare system.

Table 2. Data segmentation for the patient data

Data Segment	Features	Level of Encryption	Access Rights
Personal data	Name, DOB, SSN	High	Admin, Doctors
Medical Records	Diagnosis, Treatments, Scan reports, Prescriptions	High	Doctors and Nurses
Billing Details	Insurance and Payment details	Medium	Billing Staff
Operational data	Appointment schedules, contact details	Low	All staff

- Data Type: Sort by personal identification data, medical, billing or operational information.
- Access requirements: Create segments depending on responsibilities (for example, physicians and billing staff).

- Usage: Data segmentation for analytics, research, and patient care.
- Regulatory Boundaries: Separate data managed by separate geographical legislation (for example, GDPR vs. HIPAA).

3.1.3. Implement Data Segmentation

Database partitioning involves splitting data into various tables or databases, such as one for Personal information and one for medical history. Then use metadata to tag documents based on sensitivity and purpose [16].

3.2. Multi-Layered Encryption

Firstly, employ HE on sensitive components that require processing without decryption. HE is a sophisticated cryptographic technique that allows calculations on encrypted information without decryption. HE provides a strong framework to protect sensitive patient data while allowing activities such as analytics, ML, and statistical computations. FHE is appropriate for different computational needs with patient data. The first approach to provide arbitrary calculations on encrypted data was Gentry's Fully Homomorphic Encryption (FHE) scheme [17,18]. Technologies like secured cloud computing, privacy-preserving ML, and secured queries for databases are made possible by this fundamental advancement in cryptography. The important elements of Gentry's scheme include;

3.2.1. Lattice-Based Cryptography

Gentry's approach uses lattice-based cryptography to solve difficulties such as the Short Integer Solution (SIS) problem and the Learning with Errors (LWE) problem. The technique is highly secure since these issues are computationally difficult to resolve, even with quantum computers [19,20].

3.2.2. Ciphertexts and Noise

There is some noise in the ciphertext, or encrypted data, according to Gentry's approach. Every calculation adds to the noise, which eventually reaches a threshold that makes decryption impossible.

3.2.3. Bootstrappigng

Bootstrapping is used to reset a ciphertext's noise so that more calculations may be performed. An encrypted copy of the secret key is used to homomorphically decrypt the ciphertext. This results in a ciphertext that is "fresh" and has less noise. Because of bootstrapping, the technique is "fully homomorphic," allowing for an infinite number of calculations.

3.2.4. Structures Based on Rings

Gentry reduced costs by introducing structures such as polynomial rings for displaying keys and ciphertexts in order to increase performance. The steps in Gentry' Scheme are as follows.

3.2.5. Generation of Keys

Generate a private key and a public key. The public key is used to encrypt data. The private key is not only used for bootstrapping but also for data decryption.

3.2.6. Encryption

M in plaintext should be encrypted with the public key pk. The ciphertext c with an embedded noise is as follows

$$c = E(pk, m) \quad (1)$$

In Homomorphic computation, the Ciphertexts are directly subjected to operations like as addition and multiplication.

$$c_1 = \mathrm{E}\ (\mathrm{pk}, m_1) \quad (2)$$

$$c_2 = \mathrm{E}\ (\mathrm{pk}, m_2) \quad (3)$$

$$c_{sum} = c_1 \mathrel{+}= c_2 = E(p_k, m_1 + m_2) \quad (4)$$

Then the decryption of ciphertext c using the private key s_k is as follows

$$m = D(s_k, c) \quad (5)$$

Bootstrapping

Apply bootstrapping if the ciphertext's noise increases considerably. To perform bootstrapping, use the public key pk to encrypt the private key sk and decrypt c homomorphically using the encrypted sk. As a result, the ciphertext is "refreshed" and has less noise.

Secondly, use ABE for segments with strong access control specifications. ABC provides restricted access over things that have been encrypted. Based on qualities, it encrypts data, which can only be decrypted by individuals who possess the same characteristics. Because of this, ABE is perfect for situations requiring strict access restriction, such as patient data segregation.

Select the Key policy (KP-ABE), and the data is encrypted using characteristics, and the decryption keys contain the access policies. For instance, a user can decode data encrypted with the characteristics "Doctor AND Cardiology" by using a key that specifies those qualities.

Describe the Use Case, such as Attributes, access policies and Segments. Attributes determine which qualities (such as role, department, and region) are relevant to access control.

The access policies indicate which policies (such as "Role: Doctor AND Department: Cardiology") govern who has access to the data and define the data segments to apply policies (e.g., billing information, sensitive medical data).

The steps to implement ABe are as follows.

- Configure the master secret key and the public key.
- Specify the encryption's global settings.
- Define the attributes to provide user qualities (e.g., "Doctor" or "Cardiology").
- Create decryption keys for users according to their characteristics.
- Use an access policy (such as "Doctor AND Cardiology") to encrypt data.
- The data can be decrypted by users whose qualities meet the criteria.

Lastly, the insensitive segments can use standard symmetric encryption to improve speed. Standard symmetric encryption is an efficient means to increase processing speed while preserving a minimal level of security for insensitive patient data segments. Operational logs, publicly available information, and anonymized statistics are examples of less sensitive data that symmetric encryption is perfect for safeguarding due to its computational efficiency. Advanced Encryption Standard, or AES, is a widely used and highly secure scheme that supports a key length of 256 bits. Create a distinct symmetric key for every batch of data or data segment. Encrypt the data using the produced key with the AES algorithm. Use the same key to decrypt data as necessary.

3.3. Blockchain-Assisted Key Management (BAKM)

Blockchain-Assisted Key Management (BAKM) employs BC technology to enhance encryption key management's efficiency, security, and transparency. This method handles issues with standard key management systems, such as decentralized control, tamper resistance, and trust [21]. The steps for implementing BAKM are as follows

3.3.1. Workflow for Key Management

The following phases are involved in key management:

- Key Generation: Produce safe cryptographic keys.
- Distribution of Keys: Make sure that the keys are only given to authorized parties.
- Key Storage: Safely keep keys or key references.
- Key Revocation and Rotation: Revocation of compromised keys and regular key rotation are important.
- Audit Trail: For accountability, keep track of all important linked acts.

Blockchain Technology

Adopt a BC platform that satisfies the specifications. For instance, open BCs such as Ethereum and Polygon. Applications requiring worldwide transparency can use it. For enterprise usage with regulated participation, private BCs like as Hyperledger Fabric and Corda are perfect. Quorum and other consortium BCs strike a compromise between access restriction and decentralization.

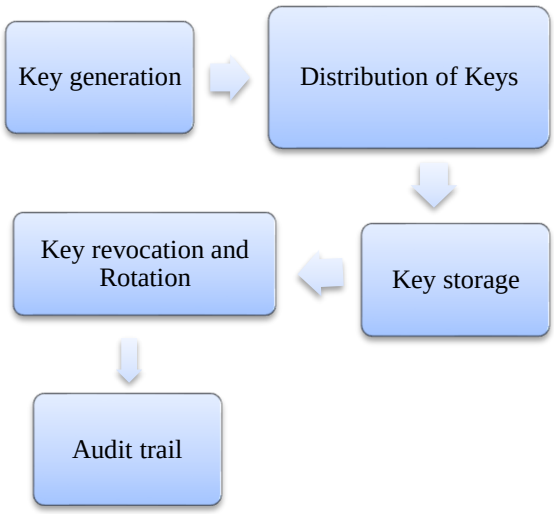

Fig. 3 Workflow for key management

3.3.2. Key Management with Smart Contracts

Key lifecycle management may be automated with smart contracts. Initially, register a new key and link it to metadata, such as the owner, characteristics, and expiration date. Establish who has access to keys and under what circumstances, with access control policies. Then employ routines to rotate keys on a regular basis and include features to alert dependent systems and revoke keys [22].

3.3.3. Protect Keys

Refrain from storing raw keys on the blockchain itself. Off-Chain Storage is used to safely store keys off-chain, such as in a key management service or a hardware security module. On-chain storage should only be used for references (hashes, encrypted keys, or metadata). Before connecting important information to the blockchain, encrypt it. The example workflow for the encryption keys of patient data managed by BC is shown in Figure 4.

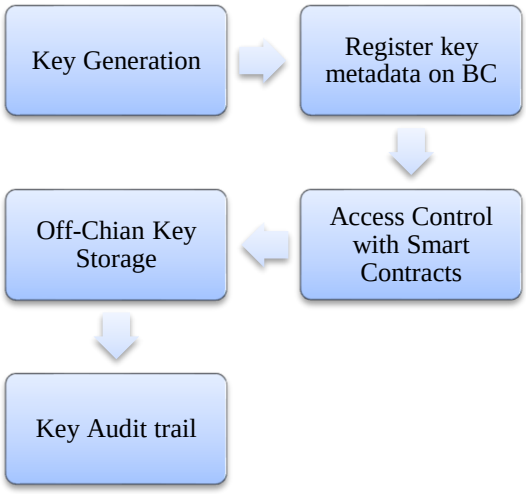

Fig. 4 Workflow of patient data encryption keys managed through BC

3.3.4. Dynamic Adaptation

Apply ML to track cloud usage habits and dynamically alter encryption algorithms for maximum performance and security. The adaptation algorithm chooses the encryption method based on the key measures like throughput, latency and resource utilization. The data were collected based on the cloud usage measures like data size, type, sensitivity, frequency of encryption algorithms, access patterns, etc.. The system usage details, like CPU, memory and bandwidth usage, were collected. Then, the algorithm strength (AES, RSA) and historical records of security attacks were also gathered for designing an ML model. Then the model is trained with the collected data using a suitable ML model like Random Forest (RF), Gradient Boosting and Neural Networks. The dynamic adaptation system works as follows.

- Track Cloud Usage: Gather data continuously and in real time.
- Forecast the Need for Encryption:
- Predict the optimal encryption technique and parameters for the given situation using the learned model.
- Implement the algorithm: Changing encryption methods dynamically (e.g., RSA for key exchange, AES for tiny data) is known as switching.
- Fallback Mechanisms: In the event that predictions do not work, make sure the default encryption scheme is applied.

Then measure the throughput, system resource consumption, and encryption and decryption times and adjust the encryption parameters and ML model in basis on the evaluation's findings [23].

4. Results and Discussion

The experimental setup for executing the multilayered scheme includes initializing the environment by installing Palisade, PyCryptodome, ABEpy, and Hyperledger Fabric on the computer. Pallisade is used to create encrypted data for sensitive data using FHE(Gentry; scheme) and PyCryptodome for encrypting the insensitive data. ABEpy is used to control access access based on the user's attributes. Finally, set up a simple Hyperledger Fabric network (using Docker and the Fabric tools) to securely store and manage encryption keys in a decentralized way.

4.1. Quantitative Assessment

A quantitative assessment of the suggested encryption technique Gentry's Scheme + AES+ABE with Blockchain and the contrasted methods such as ECC+ AES without Blockchain and DES without Blockchain are given in Table 3.

The processing cost associated with permitting FHE operations and blockchain integration causes Gentry's Scheme to have noticeably longer encryption and decryption times[24]. ECC with AES performs rather well; the first configuration's timings are marginally faster than the second's. The quickest is DES, which has far faster encryption and decryption speeds than the others, but the security is compromised in the process.

Table 3 Quantitative Performance Analysis of Proposed Methods

Measure	Gentrys Scheme +AES+ABE+BC	ECC+ AES (No BC)	DES (No BC)
Encryption time(ms)	1050	45	15
Decryption time(ms)	1115	50	25
Security	10	8	6
Throughput	150	175	350
Resource Usage	80%	45%	15%
Scalability	9	8	6

The Multilayered approach (FHE(Gentry's)+ AES+ ABE with BC) provides the highest level of security. Although it is marginally less secure than the Gentry combo, ECC plus AES offers robust security. DES has the lowest security score, which is indicative of its vulnerability to contemporary assaults and obsolescence.

The maximum throughput (350 MB/s) is attained by DES, which sacrifices security in favor of ease of use and speed. ECC with AES is effective for real-world use scenarios due to its high throughput (175–180 MB/s). Because Gentry's Scheme uses intricate encryption procedures, it has the lowest throughput (150 MB/s).

The high computational cost of FHE and blockchain is shown in Gentry's Scheme, which consumes the greatest resources (80%). Performance and resource efficiency are balanced by ECC with AES's moderate resource utilization (35–45%). DES has the least amount of resources (15%), which is consistent with its straightforward architecture.

Both ECC with AES and Gentry's Scheme receive high ratings for their capacity to adapt to a variety of challenging situations. The ECC setup requires more resources; it receives a somewhat lower score. DES is less appropriate for contemporary, large-scale applications because of its poor scalability.

From Figure 5, it is found that, with HE (enabling encrypted data operations), AES encryption, ABE for access control, and blockchain for distributed storage and auditability, this combination of Gentry's Scheme + AES + ABE + BC offers the best security. However, this has a very high cost in terms of encryption time (3050 ms), which might not be feasible for environments that need to process data quickly and securely for real-time applications.

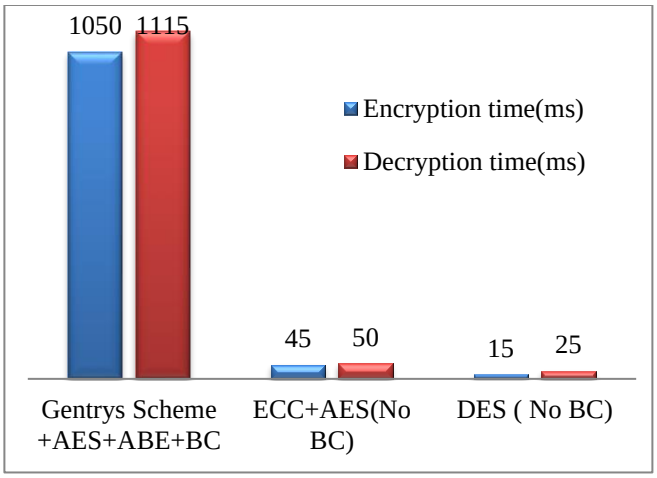

Fig. 5 Methods vs Encryption/Decryption times

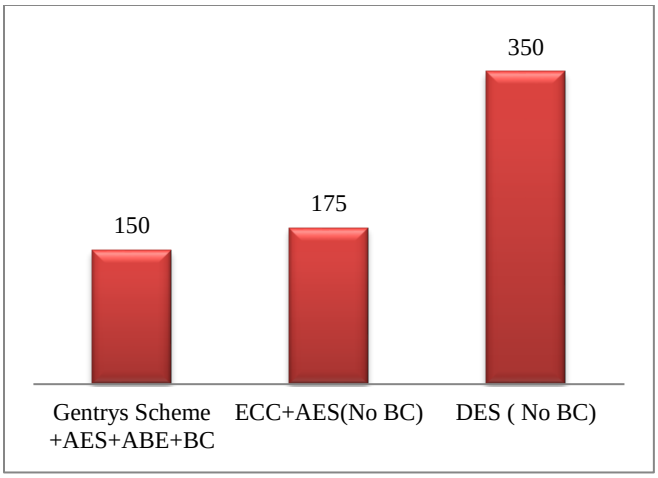

Fig. 7 Methods vs Throughput

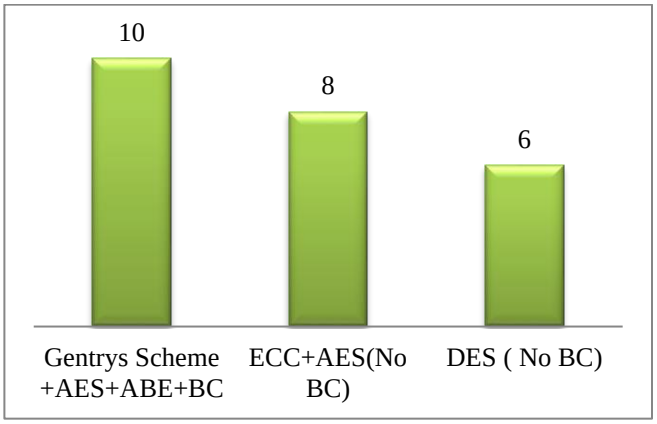

Fig. 6 Methods vs security

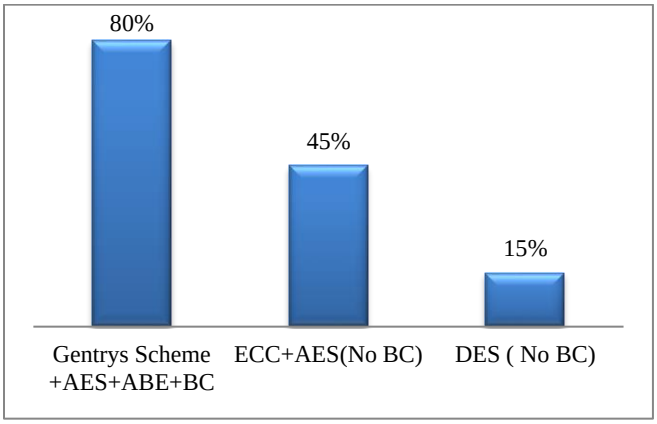

Fig. 8 Methods Vs Resource Usage

From Figure 6, it is clear that the combination of powerful cryptographic algorithms such as homomorphic encryption, AES, and ABE with blockchain ensures data integrity and auditability; this combination offers the maximum level of security (score of 10). Because ECC and AES encryption are used together, this method's security (score of 8) is still quite good. It has a somewhat lower security score than Gentry's plan, though, because it lacks the extra protection and audibility that BC offers. Due to the differences in encryption or setups, this approach could not be as safe as the preceding one, even if it makes use of powerful cryptographic techniques like ECC and AES. It may have little flaws or restrictions that cause it to score lower than the original ECC + AES technique.

From Figure 7, it is found that the proposed approach has a relatively low throughput (150), which means it processes fewer data units per second despite the excellent security it offers. This is to be expected, given the substantial computational complexity and latency added by BC and homomorphic encryption. Because this approach (175) employs ECC and AES without the additional complexity of BC, its throughput is higher than Gentry's system. The system is a little less secure than Gentry's plan, but it can analyze more data in less time.

The throughput increase (from 175 to 180) could be the result of minor variations in processing circumstances, system optimizations, or system implementation. In any case, compared to the Gentry-based approach, this strategy is still quite effective and processes data faster. DES processes the most data per second, as seen by its highest throughput of 350. However, security suffers as a result of this high throughput. Although DES has the fastest encryption time, it is not advised for applications needing robust data security because it is deemed insecure by current standards. Although DES is generally the fastest if high throughput is the goal, its security flaws make it inappropriate for sensitive data. Performance and security are better balanced with ECC + AES. Although it has a slower throughput, Gentry's scheme is ideal for users who need the highest level of protection [25].

From Figure 8, it is known that the suggested approach uses 80% of the system's resources, which is a significant amount. A resource-demanding solution is produced when HE, AES, ABE, and BC are combined. However, the robust security offered by this method justifies this resource demand. Comparing this encryption approach to Gentry's scheme, it uses a substantial amount of system resources (45%). By offering robust encryption without the disproportionate

resource requirements of homomorphic and BC encryption, it achieves a balance between security and resource utilization. Lower resource consumption is the result of the lack of BC and maybe more effective implementation.

Although it offers the highest level of security, Gentry's Scheme + AES + ABE + BC uses the most resources (80%). This makes it appropriate for settings where resource availability is not an issue, but security is a top priority. ECC + AES techniques offer robust security at a reasonable cost, consuming either 45% or 35% of system resources. These techniques are more effective and appropriate for settings where security and performance are crucial. DES utilizes only 15% of the resources; however, because of its poor security, it is not advised.

From Figure 9, it is clear that the proposed method receives a scalability score of nine despite its high computational complexity. This is due to the fact that the techniques employed can effectively manage growth in the volume of data or the number of users. Particularly in distributed systems, the combination of blockchain, AES, ABE, and homomorphic encryption offers a high degree of scalability and flexibility.

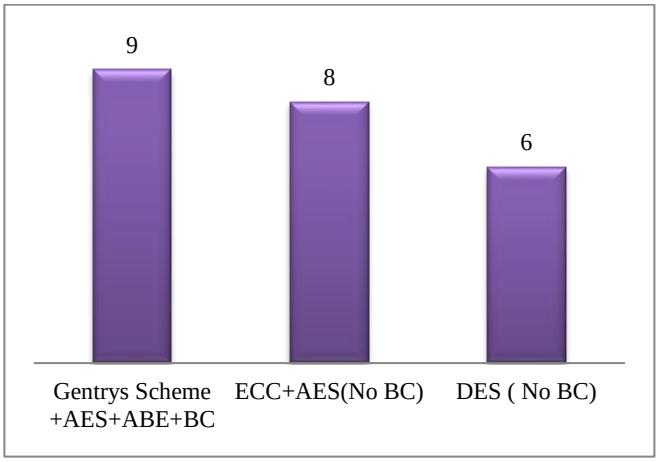

Fig. 9 Methods Vs Scalability

Additionally, this encryption technique receives a scalability score of 9. Since ECC + AES are both appropriate to efficiently manage massive volumes of data or an increasing number of users, the lack of BC has little effect on scalability. Although it still works effectively, the little drop in scalability (from 9 to 8) may suggest that this ECC + AES variation may have some minor drawbacks when handling extremely high scalability requirements. DES receives the lowest scalability score (6) because of its outdated nature and inability to effectively manage distributed or large-scale systems. Because scalability was not considered in its design, it is not appropriate for contemporary systems that must grow with the amount of data or users. Overall, DES is a bad choice for scalable systems; the best options are Gentry's Scheme + AES + ABE + BC, and ECC + AES approaches if scalability is a crucial necessity.

The strongest security is provided by the proposed approach, although the computational overhead is larger. However, techniques like ECC + AES are more performance-efficient and offer strong security, but they lack some of the other safeguards (like BC) offered by more intricate schemes. Because of its fragility, DES is seen as insufficient for current standards. The multi-layered approach (Gentry's Scheme+ AES + ABE + BC) is ideal for applications like healthcare that demand the highest level of security and compliance. It is less suitable for latency-sensitive operations because of its high resource consumption and lengthy processing times.

End-to-End Security guarantees that data is safeguarded throughout storage, transport, and processing. Fine-grained access control enables accurate and flexible sharing of data in multi-tenant cloud systems. Tamper-proof key management reduces the danger of a single point of failure in BC technology. Performance optimization is achieved by balancing encryption quality and processing efficiency while responding to data sensitivity levels.

4.2. Qualitative Assessment

Table 4 provides the qualitative performance analysis of the proposed Methods, which is an all-encompassing, comparative assessment of the cryptographic methods Gentry's Scheme + AES + ABE + BC, ECC + AES (No BC) and DES (No BC) in ten important qualitative parameters. All parameters indicate distinct operational and security aspects worthy of consideration for use in cloud computing or secure data domains such as e-health systems.

4.2.1. Key Management Complexity

Gentry's Scheme combined with AES, Attribute-Based Encryption (ABE), and BC will have a high key management complexity because of the multilayered crypto design. Managing keys across homomorphic and attribute-based schemes in a distributed ledger system increases the complexity. On the other hand, ECC + AES combinations are of medium complexity because ECC allows for efficient key generation along with shorter key sizes at the cost of somewhat weaker security. DES, being a legacy symmetric key system, is of low complexity in key management, and hence, it is easier to implement in less secure or legacy environments.

4.2.2. Energy Consumption

Energy requirements are greatest in the Gentry-based hybrid because homomorphic encryption and BC consensus processes are computationally demanding. ECC + AES systems require moderate energy, splitting between lightweight ECC calculations and symmetric AES computation. DES requires the least energy, a reflection of its ancient but simple design, which requires minimal computational effort.

Table 4. Qualitative Performance Analysis of Proposed Methods

Qualitative Metric	Gentry's Scheme + AES + ABE + BC	ECC + AES (No BC)	DES (No BC)
Key Management Complexity	High	Medium	Low
Energy Consumption	Very High	Medium	Low
Latency	Very High	Low	Very Low
Resistance to Attacks	Very High (incl. Quantum)	High	Medium
Interoperability	Medium	High	Medium
Data Integrity Assurance	Very High (via BC)	Medium	Low
Cost of Implementation	Very High	Medium	Low
Auditability & Transparency	Very High	Low	Low
Storage Overhead	High	Medium	Low
Adaptability/Flexibility	High	High	Low

4.2.3. Latency

Latency is a significant disadvantage of the Gentry-based approach. With complex calculations and BC verifications, this system has extremely high latency, rendering it less effective for real-time applications. ECC + AES systems work well with low latency, allowing for quicker encryption/decryption. DES, with its straightforward algorithmic design, provides the lowest latency of all but at the cost of current security standards.

4.2.4. Resistance to Attacks

The Gentry-based hybrid is most resistant, even to future threats like quantum attacks, because of its multilayered cryptography and BC authentication. ECC + AES is highly resistant under today's threat models but could be at risk in the post-quantum world. DES has low resistance and is vulnerable to brute-force and differential attacks because it has a short key length and an old design.

4.2.5. Interoperability

Gentry's approach has medium interoperability, as its complexity and special cryptographic components may create integration difficulties with regular systems. ECC + AES has high interoperability based on widespread use and support by current systems. DES has medium interoperability, being commonly used in the past, although currently less compatible with secure protocols.

4.2.6. Data Integrity Assurance

BC integration in the Gentry-based approach provides extremely high data integrity via immutability and consensus. In ECC + AES, data integrity has to be externally provided with separate mechanisms like digital signatures, leading to medium assurance. DES provides low integrity assurance with no native mechanisms to detect or discourage data tampering.

4.2.7. Cost of Implementation

The cost involved with Gentry-based cryptography is extremely high since the integration of several layers of cryptography and BC infrastructure is quite complex. ECC + AES-based solutions are relatively cost-effective with a good security-performance tradeoff. DES has low implementation costs, but due to its deficiency in contemporary use cases, this is a shortcoming.

4.2.8. Auditability & Transparency

The Gentry-based system leads in auditability and transparency due to BC's distributed ledger that offers verifiable records of transactions. ECC + AES and DES do not have inherent auditing functionality, which contributes to reduced transparency and challenges with traceability.

4.2.9. Storage Overhead

Because of the large ciphertext sizes during homomorphic encryption and other BC storage, the Gentry-based approach has high storage overhead. ECC + AES approaches are more efficient, and therefore, moderate storage is required. DES uses the least storage but with very poor security at the cost.

4.2.10. Improved Security Strength

The proposed model combining Gentry's Fully Homomorphic Encryption (FHE) with AES, Attribute-Based Encryption (ABE), and BC is more secure than traditional as well as hybrid models. In comparison to hybrid models such as ECC + AES or DES, our system presents multilayer protection against illegal use, like FHE's provision of quantum-resilient security. ABE integration gives high-fine-grained access control, while BC gives immutable audit trails, with good resistance to attacks such as key exposure, brute force, and man-in-the-middle. High-end models (e.g., ECC+AES, RSA hybrids) are reasonably secure against classical attacks but highly unlikely to offer quantum resistance, ease of access, and immutable authentication. By integrating cryptographic primitives with BC's distributed ledger, our framework not only guards against unauthorized data tampering but also includes verifiability and traceability—both of which are largely absent in most symmetric or asymmetric-only frameworks.

4.2.11. Enhanced Data Integrity and Transparency

The use of BC technology on our platform ensures that every interaction with data or transaction is irreversibly recorded, greatly boosting data transparency and integrity. Compared to traditional systems supported by central storage,

non-tamper-resistance of data and non-repudiation assurance, as well as distributed trust, are provided by cryptography supported by BC. Traditionally (e.g., ECC + AES, DES), integrity assurance and auditability have either been left untouched by external logging solutions or have been absent. The approach adopted here internalizes these capabilities through BC smart contracts, thereby making data access traceable, changes auditable, and misuse detectable in real-time.

4.2.12. Flexible Key Management and Access Control

Adding ABE introduces attribute-based data access controls. It thus renders key management adaptive and context-sensitive, as opposed to pre-configured key-pair techniques used in RSA, ECC, or DES. Hence, different groups of stakeholders (e.g., administrators, doctors, patients) are able to see only that section of the data they are privileged to, maximizing confidentiality without undermining usability. Earlier work by Rani et al. (2024) or Hema et al. (2024) utilizes hybrid algorithms like AES + RSA or RSA + Fernet, which are still dependent on pre-arranged public-private key pairs with some constraints in flexibility. Our solution uses a policy-based access mechanism that is more scalable in the cloud and healthcare ecosystem.

4.2.13. Auditability, Traceability, and Compliance

With BC and smart contracts, our platform is on the same level of compliance requirements (i.e., GDPR, HIPAA) through having the ability to produce transparent audit trails, origin data, and secure access. They are priceless in highly sensitive areas like healthcare and finance. Other designs, such as the ECC + AES implementations discussed in previous work, contain no audit capability and need other third-party repairs or services for compliance. That introduces attack surface and administrative burden to the system.

4.2.14. Tradeoff Optimization

Although our model is stronger in encryption and decryption latency compared to lightweight cryptography like DES or ECC-AES, it is highly balanced as far as security, scalability, and auditability are concerned, which are more crucial in real-world deployments—specifically in cloud computing and healthcare. Moreover, performance-impacting techniques like homomorphic operation offloading and key optimization by techniques like Elephant Herding Optimization (EHO-OBL) are also mitigated.

The use of BC technology in the model proposed guarantees that every transaction or data exchange is stored immutably, substituting the traditionally susceptible to-tampering data with much more secure data. As opposed to traditional centralized storage-based systems susceptible to data tampering, BC-based cryptography provides non-repudiation and decentralization of trust. In the prior work (e.g., ECC + AES, DES), integrity and auditability promise was either provided through external logging or did not exist at all. The answer reverses these attributes by bringing them inside through BC smart contracts such that access to data is traceable, changes are auditable, and theft is detectable in real-time.

5. Conclusion

The growing digitization of healthcare systems has made it necessary to manage and store sensitive healthcare data securely. A potential remedy for privacy and security issues in cloud-based settings is the combination of BC technology and multilayer encryption methods. The proposed study uses a multilayer approach to encrypt sensitive data using HE, low-sensitive data using AEs, and strict access control mechanisms using ABE. The suggested approach uses strong encryption levels and the immutable, decentralized nature of BC technology to shield medical records from manipulation and unwanted access. ML algorithms forecast security threats and dynamically modify encryption strength according to threat levels and data sensitivity. This concept highlights how BC, encryption, and ML may be combined to create a safe, flexible, and effective cloud-based healthcare data management system. Future studies should examine how to maximize the computational effectiveness of BC operations and multilayer encryption, particularly for extensive healthcare systems. The suggested architecture may create new opportunities for real-time data processing and decision-making through the combination of artificial intelligence (AI) and IoT devices.

References

[1] Reshma Siyal et al, "Blockchain-Enabled Secure Data Sharing with Honey Encryption and DSNN-Based Key Generation," *Mathematics*, vol. 12, no. 13, pp. 1-25, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] Hemant B. Mahajan et al., RETRACTED ARTICLE: Integration of Healthcare 4.0 and Blockchain into Secure Cloud-Based Electronic Health Records Systems," *Applied Nanoscience*, vol. 13, no. 3, pp. 2329-2342, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Harleen Kaur et al., "Securing and Managing Healthcare Data Generated by Intelligent Blockchain Systems on Cloud Networks through DNA Cryptography," *Journal of Enterprise Information Management*, vol. 36, no. 4, pp. 861-878, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] Hari Krishnan Andi, "Estimating the Role of Blockchain, Deep Learning and Cryptography algorithms in Cloud Security, *Journal of Trends in Computer Science and Smart Technology*, vol. 3, no. 4, pp. 305-313, 2021. [Google Scholar] [Publisher Link]

[5] Gousia Habib et al., "Blockchain Technology: Benefits, Challenges, Applications, and Integration of Blockchain Technology with Cloud

Computing," *Future Internet*, vol. 14, no. 11, pp. 1-22, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] Amna Amanat et al., "Blockchain and Cloud Computing-Based Secure Electronic Healthcare Records Storage and Sharing," *Frontiers in Public Health*, vol. 10, pp. 1-11, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Brahim Ferik et al., "A Multi-Layered Security Framework for Medical Imaging: Integrating Compressed Digital Watermarking and Blockchain," *IEEE Access*, vol. 12, pp. 187604-187622, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] R.S. Kanakasabapathi, and J.E. Judith, "Enhancing Cloud Storage Security through Blockchain-Integrated Access Control and Optimized Cryptographic Techniques," *International Journal of Advanced Technology and Engineering Exploration*, vol. 11, no. 117, pp. 1183-1197, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[9] Parag Rastogi, Devendra Singh, and Sarabjeet Singh Bedi, "An Improved Blockchain Framework for ORAP Verification and Data Security in Healthcare," *Journal of Ambient Intelligence and Humanized Computing*, vol. 15, no. 6, pp. 2853-2868, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] Sonali Shwetapadma Rath, and M.P. Prabhudev Jagadeesh, "Enhancing Data Security in SAP-Enabled Healthcare Systems with Cryptography and Digital Signatures using Blockchain Technology," *International Journal of Systematic Innovation*, vol. 8, no. 1, pp. 36-47, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[11] Garima Verma, "Blockchain-Based Privacy Preservation Framework for Healthcare Data in Cloud Environment," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 36, no. 1, pp. 147-160, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[12] Dinesh Reddy Chirra, "Secure Data Sharing in Multi-Cloud Environments: A Cryptographic Framework for Healthcare Systems," *Journal of Artificial Intelligence in Medicine*, vol. 15, no. 1, pp. 821-843, 2024. [Google Scholar] [Publisher Link]

[13] Guipeng Zhang, Zhenguo Yang, Wenyin Liu, "Blockchain-Based Privacy Preserving E-Health System for Healthcare Data in Cloud," *Computer Networks*, vol. 203, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Ashutosh Kumar Singh, and Rishabh Gupta, "A Privacy-Preserving Model based on Differential Approach for Sensitive Data in Cloud Environment," *Multimedia Tools and Applications*, vol. 81, no. 23, pp. 33127-33150, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] Rohit Ravindra Nikam, and Rekha Shahapurkar, *Data Privacy Preservation and Security Approaches for Sensitive Data in Big Data*, Recent Trends in Intensive Computing, pp. 394-408, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] Man-Jie Yuan, Zheng Zou, and Wei Gao, "Bi-Cryptonets: Leveraging Different-Level Privacy for Encrypted Inference," *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Singapore, pp. 210-222, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Ramalingam Praveen, and Parameswaran Pabitha, "Improved Gentry-Halevi's Fully Homomorphic Encryption-Based Lightweight Privacy Preserving Scheme for Securing Medical Internet of Things," *Transactions on Emerging Telecommunications Technologies*, vol. 34, no. 4, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] Ganesh Kumar Mahato, and Swarnendu Kumar Chakraborty, "A Comparative Review on Homomorphic Encryption for Cloud Security," *IETE Journal of Research*, vol. 69, no. 8, pp. 5124-5133, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[19] Victor Kadykov, Alla Levina, and Alexander Voznesensky, "Homomorphic Encryption within Lattice-Based Encryption System," *Procedia Computer Science*, vol. 186, pp. 309-315, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] Yang Li, Kee Siong Ng, and Michael Purcell, "A Tutorial Introduction to Lattice-based Cryptography and Homomorphic Encryption," *arXiv Preprint*, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] Ibrar Yaqoob et al., "Blockchain for Healthcare Data Management: Opportunities, Challenges, and Future Recommendations," *Neural Computing and Applications*, vol. 34, no. 14, pp. 11475-11490, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[22] Bessem Zaabar et al., "HealthBlock: A Secure Blockchain-Based Healthcare Data Management System," *Computer Networks*, vol. 200, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[23] Muhammad Shahid Iqbal et al., "An Adaptive Ensemble Deep Learning Framework for Reliable Detection of Pandemic Patients," *Computers in Biology and Medicine*, vol. 168, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[24] Anum Farooq et al., "Towards the Design of New Cryptographic Algorithm and Performance Evaluation Measures," *Multimedia Tools and Applications*, vol. 83, no. 4, pp. 9709-9759, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[25] M. Indrasena Reddy et al., "Encryption with Access Policy and Cloud Data Selection for Secure and Energy-Efficient Cloud Computing," *Multimedia Tools and Applications*, vol. 83, no. 6, pp. 15649-15675, 2024. [CrossRef] [Google Scholar] [Publisher Link]