ISSN : 2455-135X © 2025 International Scientific and Academic Research (ISAR) Publisher

HSMRSA-Optimized 3D-AES Cryptosystem for Healthcare Data Security

Prasanna Guduru[1] , K.Vijaya Lakshmi[2]
[1,2]*Department of Computer Science, Sir Venkateswara University, Tirupati, Andhra Pradesh, India*
[1]*prasanna.guduru@gmail.com*

Received: 17/12/2025 Revised: 18/12/2025 Accepted: 19/12/2025 Published: 20/12/2025

Abstract - *The increasing volume of medical health data has created a need for security solutions that can identify and fix errors, as well as prevent abuses of current encryption methods, especially in key generation and access provisioning on an as-needed basis. Using the three-dimensional version of dynamic key optimization, this research advances a new framework: the Advanced Encryption Standard (3D-AES) block cipher, and the Hybrid Slime Mould Reptile Search Algorithm (HSMRSA), which is used to bind the two. The HSMRSA framework further strengthens the efficiency of the search process and has been refined to operate in conjunction with the complementary method. When integrated, the two approaches enable the generation of secure keys through established cryptographic procedures. The 3D-AES cipher, like the original AES, is extended through a three-dimensional transformation model and spatial permutation, thereby improving diffusion and overall data security. A Self-Destructive Data Object (SDO) is used for access control, and a distributed hash table (DHT) is applied to split cipher text to provide a realistic time-based security model. To ensure confidentiality, integrity, and authentication of health data, we propose an enhanced architecture.*

.

Keywords: *optimization, security, healthcare data, cryptography, HSMRSA, 3D-AES,key generation, access control, distributed hash able, self-destructive data object*

1. Introduction

In order to provide confidentiality, integrity, and authentication for health data, we propose an enhanced architecture. Due to a significant increase in virtual technologies, such as remote patient monitoring, telehealth systems have been widely adopted by healthcare professionals to interact for patient care. Not only clinical services organizations [1,2], the data collected from systems such as Electronic Health Records (EHR), Diagnostic results, and large biological datasets, are prominent resources for ongoing research.

It has become easier to study the disease pattern more thoroughly and efficiently, and strengthen the idea of remote patient monitoring systems that using digital technologies. With digital technology, it has become more advantageous for healthcare research professionals to provide high-level services at their convenience. Unfortunately, this type of data storage is dangerous as it leads to cyber-attacks, as data stored in these repositories is centralised[2] which leads to cyber attacks.

Confidentiality, integrity, and data authentication play a vital role in providing patient security, confidence, and adherence to regulations [4]. Cryptography is the right way to secure the information in these vulnerable and unstable environments. In these volatile and threatened environments, the management of critical healthcare data resources has been enabled by means of cryptography as a protective measure to safeguard the data. The AES (Advanced Encryption Standard) is one of the symmetric encryption algorithms that can be considered as an effective and secure encryption solution [5,6]. However, the standard approach to using the existing public key and private key encryption algorithms will differ greatly due to the challenges of implementing these algorithms in the new health care applications. The authors identified a major critical flaw in the methods of generating encryption keys; specifically, the reliance upon the integrity of keys provided at the system level will have limited protective capabilities based on the quality of the execution and keys generated using those methodologies [7]. Present-day designs key generation processes that yield very little entropy in the final, random key outputs, and even in some cases produce fixed-key outputs that may be vulnerable to cryptanalytic techniques used to discover the original key used to generate the randomized key [8] as output.

Additionally, AES has some constraints that are not beneficial for the traditional AES on medical datasets, specifically with the UCI Heart Disease Dataset that we used to perform our analysis. The standard pattern for encrypting patient data, as referenced, likely describes the AES algorithm's use of a 4x4 byte matrix (the "state") as an "analog of traditional AES (two-dimensional)" data structure. Different combinations of clinical features can be combined into a private health record (PHR) or a confidential folder regarding health. The physical structure of this design may reduce the diffusion of the health parameters associated with each patient; hence, potential statistical associations could be inferred or made accessible through attacks if they are encrypted[8]. There are two important limitations of AES, they are the trade-off between general cryptographic strength versus computational strength, and the possible implications of that on the above-mentioned issues. The need to enhance security at the expense of unacceptable computational latency often makes several solutions not feasible for real-time medical applications of remote surgery [9], or continuous patient monitoring, and in these cases, it is not feasible for the application.

This led the researchers to work on nature-inspired optimization algorithm for secret key optimization in addition to cryptography. TSO[11] and WOA[16] were proposed for encryption key generation. These algorithms are better in security, but the overall computational overhead cost and latency in key generation. Other schemes, including TLBO-based ones, will always take a long time to process large volumes of data [12]. The intensive introduction of digital instruments, including remote monitoring tools, teleconsultation tools, and networked systems, is fundamentally transforming present-day health care data, introducing new healthcare and medical research [1, 2].

This shift to a digital ecosystem will enable us to consolidate and process a vast amount of data, whether it is electronic health records and diagnostic images, genomic data, etc., and this will spur advancement in the field of personalized medicine and remote monitoring of patients. This large amount of data is subject to significant risk from cyber attacks based on how easily they can be accessed due to the extreme scalability and accessibility of this framework. For example, centralised storage of patient data is being easily exploited by hackers[3]. Therefore, Patient Data’s integrity and confidentiality, and patient data's availability are of great concern, with implications on regulatory compliance issues and the continuance of trust and safety in the patient community[4]. Cryptography will plays an vital role in these systems because they are susceptible to attack. AES has proven to be very effective in securing data[5,6]. Mainstream cryptographic solutions continue to experience difficulties in addressing the rapidly changing healthcare provision domain. Therefore, we initially contend with a primary weakness of key generation: insufficient quality and management of encryption keys[7]. Numerous existing approaches generate keys that lack sufficient entropy, or persistently store keys in one place so they can easily be attacked with various forms of cryptanalysis[8]. In addition, traditional AES can only be used in connection with regulated "organized" medical datasets (for instance, the existing UCI Heart Disease Dataset). In addition, it is generally thought that state matrices (within the standard AES framework) have only two dimensions, when concealing the specific interactions and associations between different clinical variables in the tabulated patient health records is cumbersome, particularly since they are interrelated and generate significant health data.. AES offers robust encryption capabilities. However, several research have demonstrated that due to the rigidness of AES's structure, in some specific cases, inadequate to achieve sufficient obfuscation of correlated data given in the form of its statistical relationships. Thus, providing an attack vector for more advanced inference attacks that could be successful in extracting information from such encrypted health-related data because of correlations remaining in the data after encryption using AES.[8]. Additionally, an ongoing tension between cryptographic strength and computational capacity adds further complexities to this issue.

Security improvements are usually accompanied by unacceptable temporal computation, resulting in some of the proposed solutions being inappropriate for real-time healthcare applications (remote surgery, continuous patient monitoring, etc.) [9, 10]. Hence, the authors have started work on the applications of nature-based optimisation algorithms in cryptography, Such as TSO and WOA [11], TSO for key selection and WOA for encryption, respectively. Although they add to security, they only impose the overhead of computation and the time of generating a key. That is, TLBO systems should perform over huge amounts of raw data, which results in always higher processing time [12]. One gap we find in all of these works - and a common finding across other efforts on the topic - is the absence of a natural means for dynamic, time-based governance of data. In the absence of this, data is forever at the mercy of manual management to regulate the life cycle. To overcome some of these limitations, we introduce a new built-in cryptographic model. Our main contributions are:

- First, we developed HSMRSA, a Hybrid Slime Mould Reptile Search Algorithm, to create symmetric cryptographic keys that are robust and fast computationally.
- To circumvent the issues, we create a 3D AES block cipher as a block cipher can improve confusions and diffusions using its three-dimensional state and a new form of spatial transformation, 3D-SliceRotate, that hides sensitive correlations in health care data, without encumbering the computationally intensive process.
- We built a distributed and time-aware security system such that we have integrated a distributed hash table (DHT) for secure cipher text partition with a self-destructive data object (SDO) to implement time-bound access control.
- Finally, we fused all of these elements to form a full methodological framework based on our attributes: an attribute-based system initialized by a Trusted Authority, HSMRSA-based key generation, and 3D-AES encryption, resulting in a solution that fully encapsulates end-to-end security for cloud-based health infrastructure.

The remainder of the manuscript is structured as follows: Section 2 examines the pertinent literature. We then present our proposed methodology and architecture in Section 3. Last, Section 5 concludes.

2. Literature Review

Hence, the fast growth of healthcare systems necessitates security architectures and technologies in these systems that protect the underlying database from unauthorized access. Against that background, this review critically examines the development of proposed security models, strategies for key management, and cryptographic primitives for this goal, highlighting remaining challenges and deficiencies in the literature already reported. Data security architectures for health data have drastically changed, moving from centralized to more distributed approaches. Early frameworks made use of Centralized Trust Architectures, where only one authority had control over all cryptographic keys and access policies. In order to illustrate this point, Li et al. [13] introduced such an architecture via a Ciphertext-Policy Attribute-Based Encryption (CP-ABE) scheme. Even though CP-ABE allows fine-grained access control, it imposes a single point of failure and significantly costs computational and physical resources, and thus presents great difficulties for large health networks when it comes to scaling up. To deal with the complexity of contemporary IoMT ecosystems, the researchers introduced the Layered Security Architectures. Zhang & Li [20], for example, proposed the perception-network-cloud three-tier model, wherein security controls are tuned in each tier. While this approach optimizes resources of all kinds, it frequently creates security gaps between layers and doesn't solve the basic problem of ensuring secure key distribution throughout an entire architecture. The apparently limited scope of centralized control has led to a more recent movement towards the idea of Decentralized architectures. Sharma et al. [14] proposed a blockchain model in which data hashes and access logs are stored on a distributed ledger, and encrypted data is stored in the cloud. Although this solution provides outstanding auditability and integrity, it is not suitable appropriately for healthcare applications whose latency is an underlying concern due to performance bottlenecks introduced as a result of consensus mechanisms built into blockchain. To further define the approach and methodology, earlier, in recent years, data-centric architectures have evolved that are focused on securing data and splitting data apart. The fundamental principle is that data should be spread throughout multiple sites to avoid a single point of compromise. Though Aldabbagh et al. [15] have shown high-level cryptography for exchanging data, yet they did so in centralized storage; they did not use the concept of distributed design and have not yet exploited the security benefits of distributed data sharing. This is a blatant gap when true principles of data centrism are applied to a healthcare facility. Other than use of architecture, the relevance of keys given by cryptography makes researchers examine sophisticated key generation algorithms. An example of such application is by Nguyen et al. [11] who used Tunicate Swarm Optimization (TSO) algorithm in cryptographic key picking and obtained improved security. However, regrettably their method also came at a high price in terms of computation costs in the generation of key that accumulated to a sizeable latency, so their method was also vulnerable to latency, which might be unfavorable to applications that needed stratagem. Similarly, nature-inspired optimization by Reddy and Reddy [16] uses the Whale Optimization Algorithm as an addition approach to Elliptic Curve Cryptography. They obtained a very good security with a very large computational overhead, which poses system latency as a bigger risk. More recently, similar findings were obtained by Khalifa and Al-Masri [12]: in their 2021 Teaching-Learning Based Optimization using Homomorphic Encryption, the processing time scale with the number of data sets was found to be increasing. And, indeed, such trend is observed in a recent careful study by Goyal et al. [17], with metaheuristic strategies actually enhancing cryptographic security, however, at the performance cost that is usually undesirable and unacceptable in healthcare resource-constrained systems. One such emerging trend is a clear demand of effective optimization mechanisms that do not cause a trade-off between security and performance. However, despite these key generation challenges, because of its proven and efficient security, AES is the currently prevailing base choice for encryption of healthcare data [18]. But the standardized nature of it encouraged researchers to boost variants. For instance, Thabit et al. [10] proposed an encryption scheme that uses a genetic algorithm. The problem with these proprietary algorithms is that they cannot be subjected to advanced, peer-reviewed cryptanalysis, as do widely accepted standards such as AES. Simultaneously, dimensional enhancement is very worthwhile as it furthers security. For instance, Kanna et al. [19] proposed a 3D chaotic map-based scheme for encrypting medical images and produced better diffusion in their work. One of the disadvantages that we observe with these schemes is that they often continue to prioritize chaotic systems over coherent improvements in well-known algorithms such as AES. Second, they are usually independent without thinking about a more general scheme for key management. Through our literature review, we have noted that there are three egregious flaws of the current technology to protect healthcare data: 1) Architectural Limitations: The current architecture is illdesigned, and thus all the conventional models do not give consideration to security and performance. Architectures that are centralized have bottlenecks; those that are decentralized have excessive latency. These distributed security functions require a desperate need of architectures capable of executing those functions simultaneously without being inefficient. 2) Key Generation Inefficiency: Cryptographic security optimization techniques, suggested lately, have proven to be very expensive in terms of computing cost. 3) Disjointed Security Components: A great many cryptographic innovations and key management solutions were independent works of the authors. We want more than one set of solutions, thus this is a huge potential for more research. We need tools in order to make it possible to employ modern encryption techniques with an intelligent approach to optimize key recognition. Our work is directly motivated by these three deficiencies. We aim to establish a complete security model that unites architectural patterns, effective key management, and cryptographic advancements into one integrated solution.

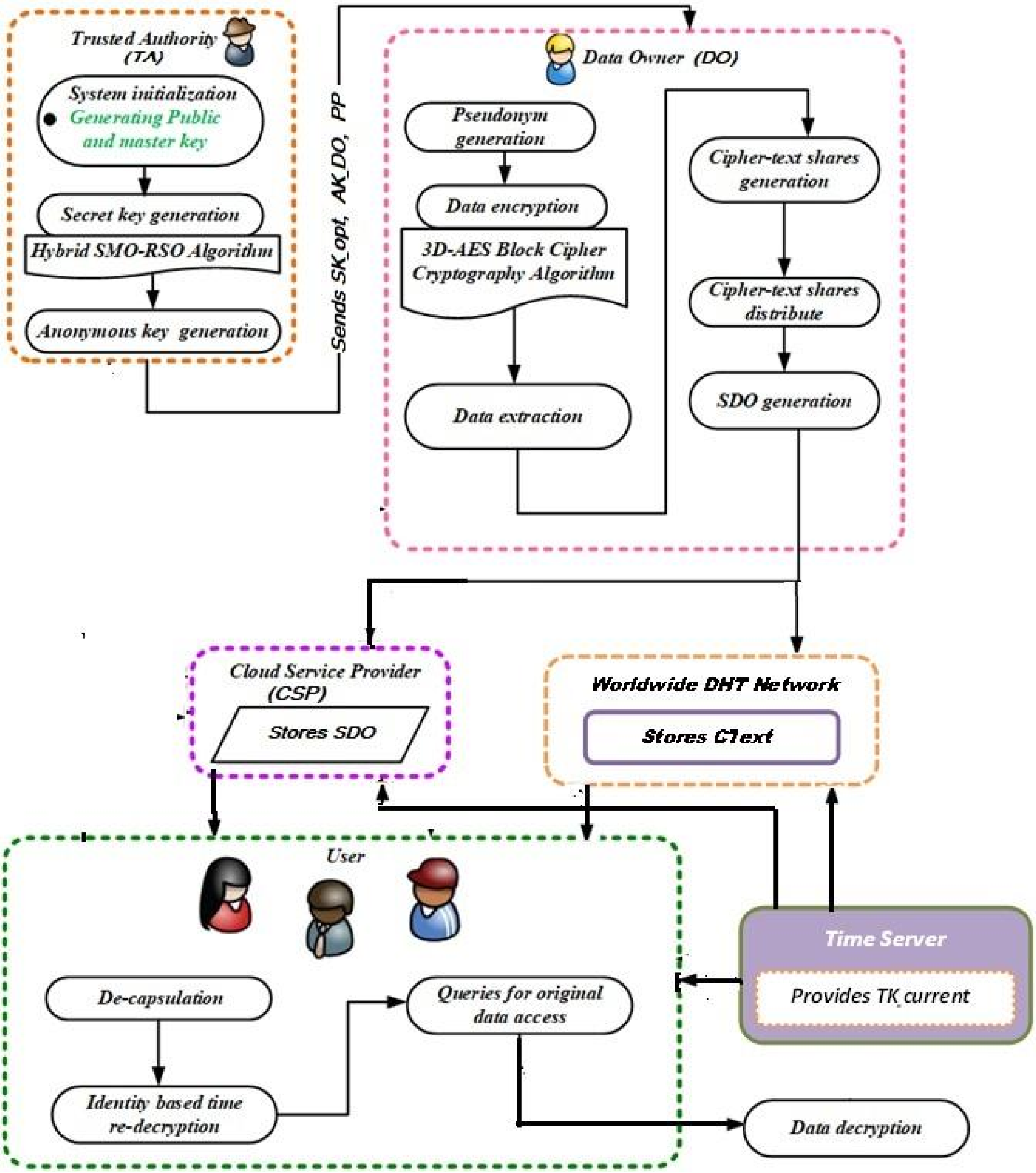

Fig 1: Proposed HSMRSA Optimized 3D AES architecture

3. Proposed Architecture

Based on the concept of four basic entities, we propose an architecture. Each plays an important part in meeting the overall system's security requirements. To set the system's initial security level, the trusted authority (*TA*) creates a master key pair *(PK, MK)* and the global public parameters (*PP)*. The TA also generates optimal attribute- based secret keys (SK_{opt}) by conducting the HSMRSA. To preserve the privacy of the data owner (AK_{DO}), anonymous keys are generated. The DO (such as a hospital or clinic) is the primary generator of the health record *M*; the DO encrypts the plaintext using the 3D-AES block cipher and optimizes the key (SK_{opt}) that is given by the TA. The ciphertext (*CT*) is subsequently divided into an encapsulated portion (CT_{enc}) and an extracted section (CT_{ext}) is generated, which are distributed on the DHT network with the generated access keys (*ak*). The DO then generates a Self-Destructive Data Object (SDO) which contains CT_{enc}, access keys, and an expiration time – T_{expire} – and makes a data transfer to the CSP for storage. CSP provides a highly flexible storage architecture that is responsible for both the maintenance of the SDOs received from the Data Owners, as well as sending back SDOs to authorized users immediately. The DHT Network is a peer-to-peer overlay that enables decentralized storage, which maintains different CT_{ext} shares and indexes them using the access keys *ak*. Because every DHT node holds only part of the shares, no one person has enough information to rebuild the data.

3.1. Proposed Architectural Workflow

The operational workflow of our architecture unfolds through a sequence of cryptographically secure phases. In the following subsections, we detail these phases and introduce the mathematical models that drive the HSMRSA key optimization, 3D-AES encryption, and data partitioning processes.

Phase 1: System Initialization and Key Issuance

In the first phase, the Trusted Authority (*TA*) initializes the entire system. Its initial responsibility is to generate the public parameters (*PP*) and the master keys. Then, for a single user (the Data Owner) with a set of attributes S, the TA generates a unique, optimized secret key, SK_{opt}, using our Hybrid Slime Mould Reptile Search Algorithm (HSMRSA). Our suggested HSMRSA algorithm optimizes the key generation process through a structured, iterative technique. Initializing a population of possible key solutions, W_i , as the first step. Over T iterations, it refines these candidates. The core of HSMRSA is a hybrid position update that combines the strengths of the Slime Mould Algorithm (SMA) and the Reptile Search Algorithm (RSA).

The Slime Mould Algorithm (SMA) component is responsible for a global, adaptive search of the entire key space. It performs an oscillatory exploration to discover highly random and non-deterministic key candidates. We can represent this using the following below equation:

$$\vec{W}(\mathrm{t}+1) = \begin{cases} \vec{W}_{best}(t) + \vec{vb}\,.\left(\vec{W}(t)\,.\,\vec{W_A}(t) - \vec{W_B}(t)\right) \\ \quad \text{if } r < P \\ \vec{vc}\,.\,\vec{W}(t) \end{cases}$$

This approach ensures the discovery of highly random and non-deterministic key candidates. Together, the adaptive weight $\vec{W}$ and the oscillating parameter $\vec{vb}$ Work to maximize the entropy and unpredictability of the potential keys. This establishes a strong cryptographic foundation by forcing the algorithm to explore the entire solution space thoroughly and avoid getting stuck in local optima.

Simultaneously, the Reptile Search Algorithm (RSA) component homes in on the most promising key candidates identified by the SMO, performing a targeted, hunting-inspired refinement. We can define this strategic exploitation with the formula:

$\vec{W}(t+1) =$

$$\vec{W}(t) + \vec{W}_{best}(t) \,.\, \eta \vec{H}(t) \,.\, (\vec{W}_{best}(t) - \vec{W}(t) \,.\, \mu)$$

Rapidly converges the population toward the optimal key weight W_{opt} . The hunting parameter η and evolutionary function $\vec{H}$ Enable precise, step-wise improvements, which significantly reduce computational latency and ensure efficient convergence without sacrificing security. The algorithm terminates after T iterations, outputting the optimized weight vector W_{opt} . The final optimized secret key for the user is structured as:

SK_{opt} = (D = g_2^{α · W_{opt}},
∀y ∈ S: D_y = g_1^{W_{opt}} · H_1(y)^{r_y},
$D_{y'}$ = g_1^{r_y})

Where $r_y = H_2(W_{opt} \parallel y) \bmod p$. The TA also issues an Anonymous key $AK_{DO} = H_1(ID_{DO})^{\beta}$ Where:

- $W_{opt} \in Z_p^*$ is the optimized weight from HSMRSA
- $r_y = H_2(W_{opt} \parallel y) \bmod p$ are deterministically derived random values
- S is the set of user attributes
- $\alpha \in Z_p^*$ is the master secret exponent from key generation
- H_1: {0,1}* → G_1 hashes attributes to group elements
- H_2: {0,1}* → Z_p^* generates random exponents,
- r is derived from the HSMRSA optimization process.

Phase 2: Encrypting and pre-processing data. The DO encrypts a plaintext health record M using the 3D-AES block cipher along with the optimized key SK_{opt}. Here are some essential steps for this process:

1) *3D-AES Encryption:* The 512-bit plaintext M is first arranged in a three-dimensional State Cube: A[x,y,z] 4×4×4. The encryption process then follows a multi-round structure. It starts with an initial AddRoundKey operation. This is followed by 9 main rounds of transformation. The process concludes with a final round (Round 10). Each of the 9 main rounds is composed of the following operations, which are performed on the state cube:

- *SubBytes:* We substitute each byte in the cube non-linearly using a standard S-box.
- *ShiftRows***:** We cyclically shift the rows within each two-dimensional slice of the cube.
- *3D-SliceRotate:* This is our novel spatial permutation, where we rotate entire slices of the cube along the z-axis by angles of 0°, 90°, 180°, or 270°.
- *MixColumns:* We apply a linear transformation to mix the columns, using the following matrix operation:

$$\begin{bmatrix} s'_{0,z} \\ s'_{1,z} \\ s'_{2,z} \\ s'_{3,z} \end{bmatrix} . = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \bullet \begin{bmatrix} s_{0,z} \\ s_{1,z} \\ s_{2,z} \\ s_{3,z} \end{bmatrix}$$

- *AddRoundKey*: XOR with the round key K_{round} : $S'[x,y,z] = S[x,y,z] \oplus K_{round}$ $[x,y,z]$. The final round of 3D-AES omits the MixColumns step. The output of this process is the final 512-bit cipher text, *CT*.

2) Data Partitioning: Next, the ciphertext CT is split into n segments to enable distributed storage. In this design, the first m segments constitute the Extracted Cipher text (CT_{ext}) used for DHT networks, and the second n−m segments represent the Encapsulated Cipher text (CT_{enc}) to the cloud. The partitioning in this way can be formalized by

$CT = \{CT_1, CT_2, \ldots, CT_n\}$, *where* $CT_{text} = \{CT_1, \ldots, CT_m\}$ *and* $CT_{enc} = \{CT_{m+1}, \ldots, CT_n\}$. You build the SDO: the Data Owner fabricates a self-destructing data object (SDO), assembling all of the necessary building blocks to achieve secured, timely access:

$SDO = \{CT_{enc}, \{ak_1, ak_2, ., ak_m\}, T_{expire}, PS, PP\}$. The main components of an SDO are the encrypted ciphertext encapsulating the data, the Access Key Set in order to retrieve the shares from DHT, the expiration timestamp, a pseudonym for the information to stay anonymous, and a public system parameter.

Phase 3: Secure Data Storage and Distribution

This stage turns the data-centric security model into a concrete manifestation. The CT_{ext} shares are distributed over the DHT network, where each share CT_i is located at a point defined by its access key ak_i. At the same time, the CTenc can be safely loaded into the SDO on the CSP. Thereby ensures that an attack against the DHT only or cloud storage alone will not expose the whole dataset.

Phase 4: Controlled Data Access and Decryption Data access begins on request by an authorized user by retrieving the SDO from the CSP, and then accessing keys stored within it to retrieve all CT_{ext} shares from the DHT. All segments are concatenated into the whole ciphertext: $CT_{comb} = CT_1 \mid\mid CT_2 \mid\mid \ldots \mid\mid CT_m$. Upon the same operation, the order of the decryption process reverses the encryption; the Inverse Final Round begins with the implementation of inverse operations on the process (InvAddRoundKey for removal of the last round key, Inv3D-SliceRotate for a correction of cube slices) and InvShiftRows for rotation of the rows, with InvSubBytes performing inverse byte changes via the inverse S-Box. For the Inverse main Rounds (which run from round 9 down to 1), we repeat the cycle with all five basic steps: InvAddRoundKey; InvMixColumns (precomputed inverse matrix reverse column mixing); Inv3D-SliceRotate, InvShiftRows and finally, InvSubBytes. It ends with another Inverse Initial Round, where the InvAddRoundKey is used to dispose of the original round key. The complete inversion of each of those steps gives us the true plaintext M.

Decryption takes place as well in the SDO under the time-bound policy. It is a function of this that the T_{expire} parameter implements time-based access control design to create the data unavailability, and hence not able to be cryptographically accessed when a set time passes, to establish proactive and automated systems for organizing, storing, and restoring the information.

4. Advantages of Proposed System

There are unique benefits that are obtained in the proposed process over the current models:

- *Improved Privacy:* Combining the optimized key via HSM-RSA and the 3D-AES cipher offers a more robust cryptographic underpinning than AES only or optimizing the key by itself.
- *Resilience through Distribution:* The decentralization of the ciphertext storage solution, by means of DHT, mitigates the centralized cloud storage, thus reducing the risk of data compromise which is another problem of the centralized cloud storage.
- *Dynamic Data Governance*: Integration with SDO features, which have a time-to-live support, provides automatic data life cycle management, which is not found with many available architectures today.
- *Balanced Trust Model*: The system balances manageability and resiliency; initial trust in the system is provided by the centralized Trusted Authority and the data is stored in the network on the decentralized model.

These components together provide the analytical background for both the HSMRSA optimization itself and the 3D-AES cipher, which will be detailed in detail in coming parts of this work.

5. Conclusion

In ensuring the security of the health data processing, this paper suggests a combination of the cloud approach. It directly tackles some of the key flaws of the prevailing system which include ineffective key generation, ineffective dynamic governance and disconnected security system. This solution represents a centralized architecture that has numerous central features. The Hybrid Slime Mould Reptile Search Algorithm (HSMRSA) will be constructed and implemented in such a way that efficient extraction of strong cryptographic keys is achieved. The other breakthrough will be a new and improved 3D-AES block cipher by adding a new three-dimensional data structure and a new spatial permutation to facilitate the most difficult confusion and diffusion of structured data, such as medical records. They will then be enhanced with a Distributed Hash Table (DHT) based storage model and Self-Destructing Data Objects to come up with a data governance model that is resilient and time-sensitive. These elements together provide us with a complete security solution and can be able to match the finest quality of secrecy of medical infrastructure with the sensitive workload demands of health care.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this research work.

Funding Statement

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Acknowledgments

The authors sincerely thank the
reviewers and editors for their valuable comments and suggestions, which helped improve the quality of this manuscript.

Reference

[1] V. Choudary Nuvvula, “Cloud Technologies Revolutionizing Healthcare: Scalable Transaction Systems for Patient Data and Real-Time Diagnostics,” *Int. J. Res. Comput. Appl. Inf. Technol.*, vol. 7, no. 2, pp. 2197–2208, 2024.

[2] A. Haleem, M. Javaid, R. P. Singh, and R. Suman, “Telemedicine for healthcare: Capabilities, features, barriers, and applications,” *Sensors International*, vol. 2, p. 100117, Jul. 2021.

[3] M. Al-Hawawreh and E. Sitnikova, “Leveraging deep learning models for ransomware detection in the industrial Internet of Things environment,” in *Proc. Military Communications and Information Systems Conf. (MilCIS)*, Canberra, Australia, Dec. 2019, pp. 1–7

[4] R. Sivan and Z. A. Zukarnain, “Security and privacy in cloud-based e-health system,” *Symmetry*, vol. 13, no. 5, p. 742, Apr. 2021.

[5] D. Liu, Z. Yan, W. Ding, and M. Atiquzzaman, “A survey on secure data analytics in edge computing,” *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4946–4967, Jun. 2019.

[6] G. Xu, F. Wang, M. Zhang, and J. Peng, “Efficient and provably secure anonymous user authentication scheme for patient monitoring using wireless medical sensor networks,” *IEEE Access*, vol. 8, pp. 47282–47296, Mar. 2020.

[7] M. Rana, Q. Mamun, and R. Islam, “Enhancing IoT Security: An Innovative Key Management System for Lightweight Block Ciphers,” *Sensors*, vol. 23, no.18, Art. 7678, Sep. 2023

ISSN : 2455-135X © 2025 International Scientific and Academic Research (ISAR) Publisher

[8] A. Alabdulatif, I. Khalil, and M. S. Rahman, “Security of blockchain and AI-empowered smart healthcare: Application-based analysis,” *Applied Sciences*, vol. 12, no. 21, p. 11039, Oct. 2022,

[9] G. Xu, F. Wang, M. Zhang, and J. Peng, “Efficient and provably secure anonymous user authentication scheme for patient monitoring using wireless medical sensor networks,” *IEEE Access*, vol. 8, pp. 47282–47296, Mar. 2020

[10] Thabit, F., Alhomdy, S., & Jagtap, S. (2021). A new data security algorithm for cloud computing based on genetic techniques and logical-mathematical functions. *International Journal of Intelligent Networks*, 2, 18-33.

[11] Nguyen, G. N., Viet, N. H. L., Joshi, G. P., & Shrestha, B. (2020). Intelligent Tunicate Swarm-Optimization-Algorithm-Based Lightweight Security Mechanism in Internet of Health Things. *Computers, Materials & Continua*, 65(2), 1141-1153.

[12] Khalifa, M. S., & Al-Masri, A. N. (2021). An Optimal Teaching and Learning based Optimization with Multi-Key Homomorphic Encryption for Image Security. *Journal of Cybersecurity and Information Management (JCIM)*, 7(2), 77-84.

[13] Li, M., Yu, S., Ren, K., & Lou, W. (2020). Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings. *IEEE Transactions on Dependable and Secure Computing*, 17(1), 78-91.

[14] Sharma, P., Jindal, R., & Borah, M. D. (2021). Blockchain-based decentralized architecture for Cloud Storage System. *Journal of Information Security and Applications*, 62, 102970.

[15] Aldabbagh, G., Alghazzawi, D. M., Hasan, S. H., Alhaddad, M., Malibari, A., & Cheng, L. (2021). Secure Data Exchange in M-Learning Platform using Adaptive Tunicate Slime-Mould-Based Hybrid Optimal Elliptic Curve Cryptography. *Applied Sciences*, 11(12), 5316.

[16] Reddy, T. V., & Reddy, A. R. (2022). ECC Image Encryption Scheme using Whale Optimization Technique. *International Journal of Multidisciplinary Engineering in Current Research*, 7(5), 1-6.

[17] Goyal, S., Sharma, B. B., & Mauriya, R. (2024). Metaheuristic approaches in cryptography: A systematic review and future directions. *Journal of Information Security and Applications*, 80, 103679.

[18] National Institute of Standards and Technology (NIST). (2023). Advanced Encryption Standard (AES). FIPS PUB 197.

[19] Kanna, G. P., Sriram, V. S., & Kumar, S. S. (2023). A chaos-based medical image encryption scheme using the butterfly optimization algorithm for cloud security. *Journal of Real-Time Image Processing*, 20(1), 3.

[20] Zhang, J., & Li, X. (2021). A lightweight encryption scheme for real-time data in IoT-based healthcare. *IEEE Internet of Things Journal*, 9(2), 1429-1441.