DSAS: A Secure Data Sharing and Authorized Searchable Framework for e-Healthcare System

Appasamy M*[1], Madhu S[1], Deepa R[1]
[1]Department of Artificial Intelligence and Machine Learning, St. Joseph's College of Engineering, Chennai, India,
appasamyarun715@gmail.com

Abstract. Electronic Health Record (EHR) systems including cloud-based versions are scalable and allow access remotely, however, there is a high risk posed by the lack of data confidentiality, unauthorized access, and regulatory non-compliance. Available solutions like the Attribute-Based Encryption (ABE), Proxy Re-Encryption (PRE), and blockchain-based audit systems are very secure but not applicable in real-life situations in a hospital setting as they are computationally expensive, their key management is complicated, and their performance is sluggish. To overcome these shortcomings, we introduce DSAS, a lightweight Data Sharing and Authorized Search architecture that integrates AES-256-GCM encryption, RSA-2048 key wrapping, the role access control and the searchable Symmetric Encryption (SSE). DSAS provides patients with the ability to store encrypted EHRs in the cloud, open and close access to healthcare professionals dynamically, and conduct privacy-preserving keyword search without making plaintext available to the cloud. Also, the HMACchained audit log can be tamper-resistant, making the accountability and HIPAA and GDPR standards compliance. Through experimental analysis, it is proven that DSAS can provide lowlatency encryption, encrypted search at speed, and re-keying overhead, and thus is a viable, deployable, and secure solution to next-generation e-Healthcare systems.

Index Terms: Electronic Health Records, Cloud Computing,Searchable Encryption, Access Control, Healthcare Security,DSAS

1 Introduction

Healthcare is the sector where electronic revolution has led to an increase in the rate at which Electronic Health Records (EHRs) are stored and managed using cloud-based solutions. The use of cloud systems is rapidly increasing in hospitals, clinics and telemedicine providers to enable remote consultation, collaboration of various hospitals and to continuously monitor the patient [1]. Even though cloud-based systems of the EHR can be expected to be more cost-effective and scaled, they also pose significant risks of confidentiality violations, unauthorized dissemination, insider abuse, and failure to comply with high-protective privacy regulations, such as HIPAA and GDPR. Patient sensitive data should not be revealed to even the storage provider due to lack of trust in cloud servers.

S. P. Vijayaragavan et al. (eds.), *Proceedings of the Global Conference on Sustainable Energy Systems, Smart Electronics and Intelligent Computing (GCSESEIC 2025)*, Advances in Engineering Research 297,
https://doi.org/10.2991/978-94-6239-654-8_44

To remove these risks, several research studies have proposed cryptography models, which are founded on the basis of Attribute-Based Encryption (ABE) [3], Proxy Re-Encryption (PRE) [4], secret sharing or blockchain [5]. Even successful theoretically, these approaches are not so successful in practice:

- High cost of computation unsuitable in big hospitals,
- Complex procedures in the key management and revocation ,
- This is due to poor usability by clinicians and, Inability to engage in an efficient search (real time encrypted), which is essential when dealing with an emergency case and during diagnoses [6],[7].

As a result, most of the solutions offered are not suitable to be incorporated in the current healthcare systems. The significance of lightweight, fast and simple models of security is even further elaborated in studies based on clinicians. In line with high privacy protection, healthcare workers must be able to access instant information about patients, have good predictability, and see access controls.

We address these gaps by suggesting a lightweight Data Sharing and Authorized Search system, DSAS, which is created with a specific focus on the real-life cloud-based healthcare environment. DSAS integrates potent cryptographic techniques to participate file encryption (AES-256-GCM), secure key exchange (RSA-2048), encrypted key word search (Searchable Symmetric Encryption — SSE) [1], [2], and role-based access control (RBAC) on the basis of Spring Security, and HMACchained audit logging into one and executable architecture. Compared to ABE/PRE-based models [3], [4], DSAS is fast, straightforward, and deployable, which ensures that patients gain complete access management and doctors can search without putting the data at risk of being compromised to the cloud service.

The significant contributions of the work are:

- Lightweight and efficient EHR security system that has
  AES, RSA, SSE, RBAC and auditability;
- An algorithm to re-encrypt keys such that a patient can be flexible in granting/revoking keys;
- The fast encrypted search engine founded on SSE that requires less than 75 ms to fulfill the search on massive data sets [6], [7], [14].
- Resistant audit chain through the use of tampering that has accountability and regulatory compliance [12].
- Spring Boot, React, MySQL and AWS S3 are fully cloudfriendly and were verified through experimental testing.

2 Related Work

Significant effort has been put on maintaining the safety of Electronic Health Records (EHRs) in insecure clouds. The available literature is mainly based on either Attribute-Based Encryption (ABE), Proxy Re-Encryption (PRE), searchable encryption, secret sharing, or blockchain-based auditing [15]. Although both strategies offer theoretical security assurances, their practical constraints make them not applicable in a largescale implementation in real healthcare.

The initial Personal Health Records (PHR) access-control systems were based very much on the Ciphertext-Policy Attribute Based Encryption (CP-ABE) as a means to define fine-grained policies about encrypted data [3]. These models provide multi-user access but have the disadvantage of slow encryption/decryption, high ciphertext size, and are computationally expensive, and are not suitable in real-time clinical processes. Superior extensions like HASBE enhance scalability at the expense of expanding key management and user revocation [3].

In order to enable delegated access, a variety of studies present Proxy Re-Encryption (PRE)-based models, in which semi trusted proxies re-encrypt ciphertext on behalf of new customers [4]. Even though PRE lessens the load on the data owners, it creates the risk of key leaking out, costly re-encryption processes and dependency on trusted transformation servers that dilutes the overall threat model. The PRE systems also do not have efficient keyword searching of encrypted EHRs.

Searchable Symmetric Encryption (SSE) and public-key searchable encryption has been considered to encrypt search in healthcare [2]. These solutions offer a simple search capability using keywords, but they do not have all the critical characteristics of healthcare including patient-controlled access, audit trail, secure key sharing, and dynamic revocation of permissions. The majority of SSE works do not consider search as a component of a full EHR security architecture but as a separate function.

New blockchain-derived EHR systems can provide auditability with no tampering, but have high latency, high storage, and lack scalability at high volume transaction rates [5]. Such limitations render blockchain inappropriate in emergency care or even in a massive hospital where the sub-second response time is essential.

Alternative schemes like secret sharing, homomorphic encryption, and multi-party computation assume high levels of theoretical privacy but are computationally infeasible to regular healthcare practice, particularly in the high-frequency access to EHR and constant updates [12].

Unlike the categories mentioned above, DSAS has a realworld and lightweight deployment strategy. Instead of using extensive cryptography tools, DSAS incorporates AES-256GCM, RSA-2048 key wrapping, SSE-based encrypted search, RBAC authorization, and HMAC-chained auditing into a single system. Fig.1 design provides a high degree of security without compromising high usability, low latency, and of easy deployment in modern cloud settings, a significant gap that has existed in previous studies.

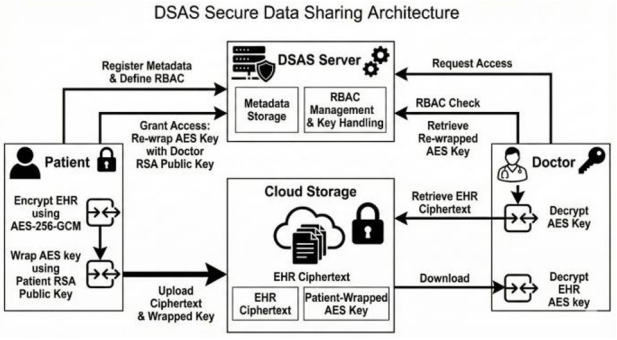

Fig. 1. DSAS System Model Overview

3 Methodology

This section describes the architecture of below Fig.2, cryptographic primitives, and operational workflows of DSAS. We present the design objectives, key algorithms (encrypt/upload, grant/revoke, search, audit), and practical implementation notes.

3.1 Design Objectives

The DSAS architecture is developed by having a series of lightweight but strong security goals suitable to the practical implementation of healthcare on clouds. Its main aim is to make sure that all Electronic Health Records (EHRs) are confidential even in case of storing on non-trustworthy cloud infrastructure. DSAS also allows patient-centric access control

model where a patient reserves the ultimate right to provide and withdraw access to eliminate unauthorized access to medical information. The system also provides secure and low latency search of encrypted records, which means that medical care professionals will be able to effectively use it to access the necessary information without exposing sensitive data to the cloud. In order to address real-time clinical needs, it is based on computationally-efficient cryptographic primitives and thus much more deployable than heavy ABE- or PREbased systems. Lastly, DSAS integrates an audit mechanism that is tamperproof to ensure traceability and compliance with standards like the HIPAA and GDPR.

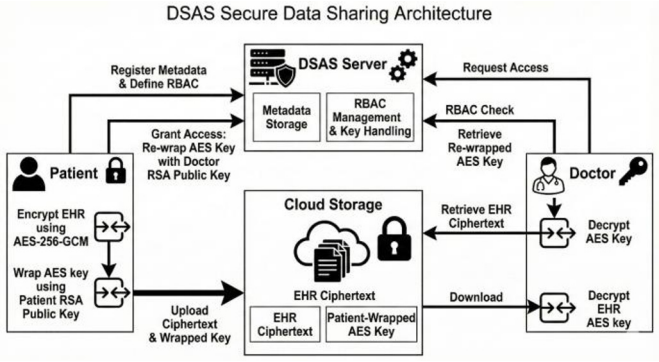

Fig. 2. DSAS Secure Data Sharing Architecture

3.2 Cryptographic Foundations

The DSAS framework is based on a collection of highperformance, industry standard, cryptographic primitives to provide confidentiality, integrity, authentication, and secure data sharing. AES-256-GCM is an encrypted EHR file that offers authenticated encryption with minimal overhead, so it is suitable in large file sizes that often occur in medical imaging and diagnostics. The AES keys are per-file wrapped and distributed safely with RSA-2048 to the authorized parties so that they can be flexibly controlled without the use of complex policies based on attributes. Searchable Symmetric Encryption (SSE) is included by use of HMAC-SHA256derived tokens which enables the search with encrypted keys and the search process expresses no plaintext to the cloud. BCrypt hashing is used to protect user credentials, and TLS is used to secure all communication between servers and clients. These are lightweight primitives and enable DSAS to offer high security assurances without reducing the performance.

3.3 Secure Upload and Storage

When a patient posts an EHR to DSAS, the platform will launch a multi-step procedure of ensuring confidentiality and secure cloud storage. An individual file encryption key based on AES-256 is created on a per-record basis and the plaintext EHR is encrypted locally and do not leave the users device. The ciphertext that comes out is resistant to manipulation and unauthorized alteration as a result of the integrity protection of GCM. AES key is then encrypted with the RSA key of the patient in order to make sure that only the patient will be able to decode the file using the key. DSAS identifies metadata keywords in the EHR and converts them into deterministic search tokens based on HMAC and encrypted in an index stored in an encrypted format in the server. The encrypted EHR is stored in cloud storage, and metadata, such as wrapped keys and token mappings, are stored in the backend of the DSAS. All the activities in this workflow are documented in an audit log that is resistant to tampering.

3.4 Access Granting

DSAS has a patient-centered access granting model where only the patient is allowed to grant other users- e.g. doctorsaccess to the specific EHRs. The patient decrypts the key of AES file with personal RSA key and re-encrypts it with the key of the doctor. This creates a fresh wrapped key, which when decrypted can only be deciphered by the doctor. Since the server is only presented with wrapped keys, and does not know anything about the plaintext AES keys, the system provides a high level of confidentiality. The key which is new and is wrapped is then appended to the DSAS metadata and the key can then be accessed by authorized doctors who can then use it to decrypt the EHR of the corresponding key. The given process does not require trusted proxies or costly cryptographic transformations and offers an efficient alternative to multi-authority ABE and PRE-based delegation, simple and efficient at the same time.

3.5 Access Revocation

In DSAS, access revocation is implemented via secure rekeying, which does not allow former authorized users to still decrypt the updated or future versions of an EHR. In case a patient withdraws access on behalf of a user, the system will create a new AES file key and re-encrypt the EHR file. All other authorized users then have their keys regenerated with the revoked user being removed in the process. As the user who has been revoked no longer holds a valid wrapped key to the new AES key, they are cryptographically denied access to the file even in case they still have old copies of metadata. The model of revocation provides

a high degree of forward secrecy and unauthorized post-revocation access, but at a low level of computation than complete reconfiguration of ABE policies.

3.6 Authorized Retrieval

First, in order to gain access to an EHR, an authenticated doctor is placed in the role-based authorization of the Spring security access control layer of DSAS. In case of valid permissions, the system returns the AES key wrapped in a Doctor-specific key of the file demanded. The doctor uses his or her private RSA key to un wrap this wrapped key to reclaim the AES file key. The encrypted file of EHR stored in the cloud can be decrypted on the local level. This will guarantee that the plaintext data will never be disclosed to the DSAS server or cloud provider. Since the processes of both permission checking and decryption take place on the client-side, DSAS will not allow unauthorized users, such as privileged insiders, to bypass access controls or access sensitive medical information.

3.7 Searchable Encryption.

DSAS facilitates safe search of encrypted EHRs with the help of a lightweight mechanism of Searchable Symmetric Encryption (SSE). Metadata keywords during the upload stage are turned into deterministic HMAC-based search tokens, and these are stored in a search index encrypted. However, upon a keyword search by the doctor, the client of the doctor creates the corresponding token by the same HMAC function and sends it to the DSAS server. Matches This token is compared on the server against the encrypted index, without knowing the underlying keyword or accessing the plaintext dataset. DSAS then comprehensively filters matching document IDs via RBAC and only results that can be accessed by the doctor are executed. Since only the encrypted tokens are processed by the cloud, there is no meaningful information in the search operation which as well has low latency even when working with large datasets Fig.3.

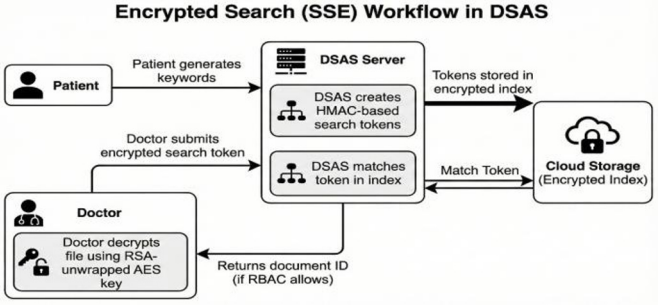

Fig. 3. DSAS Encrypted Search (SSE) Workflow

3.8 Audit Logging

In the DSAS, auditability is provided by using a chained logging system where each computer entry in the logs is cryptographically linked to the previous one using an HMAC function. Each operation including upload, search, download, grant, and revoke is logged in terms of user identifiers, timestamps, and the type and outcome of the operation. The chained design such that any changes made to a historical record invalidate all the entries is such that the changes made becomes immediately noticeable. This will offer powerful forensic assurances and promote regulatory adherence to HIPAA and GDPR that mandates comprehensive and immutable logs of any interactions with patient information.

4 Implementation Details

DSAS architecture is introduced in the form of the modular cloud-ready system with a strong consideration of the broadly used and popular backend, front and storage technologies to secure the simplicity of deployment, scaling and performance. The backend is created based on Spring Boot that gives a solid platform to apply the RESTful APIs, authentication modules, and access control mechanisms. DSAS implements Spring security which provides role based access control (RBAC) so that only authenticated and authorized users can start the search or retrieval operation. The frontend is developed with React.js, which provides a user-friendly system that allows patients to operate their medical records and doctors to carry out authorized searches and access available EHRs. All API communications are carry out through HTTPS to ensure security and integrity of transferred data.

The files are encrypted EHRs that are stored in AWS S3 which is highly durable and available although viewed as untrusted regarding confidentiality; thus, the files are encrypted throughout their length. In a bid to provide secure authentication, user passwords in DSAS are stored using BCrypt that reduces brute-force and rainbow-table attacks. The RSA key pairs of all the users are created locally in the client-side and are stored safely to ensure the unauthorized access.

Fig.4, Fig.5 DSAS is packaged as Docker, which allows it to be deployed in a variety of cloud environments (e.g. AWS EC2) or within on-premise hospital servers. The backend service, search engine and audit subsystem can be scaled independently ondemand, based on the load of the system by the modular architecture. MySQL is configured in a replicated mode to aid reliability, failover, ensure that metadata and audit logs are not lost due to system failures.

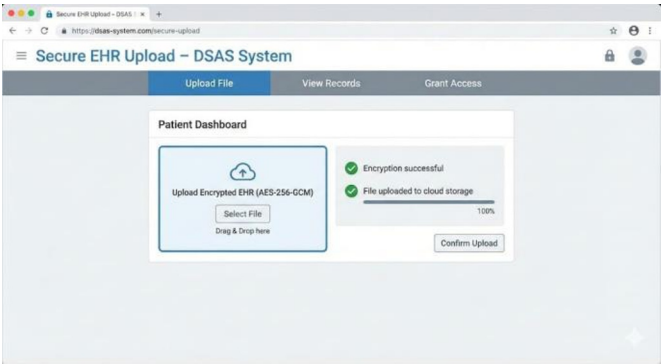

Fig. 4. Secure EHR Upload Interface in the DSAS Patient Dashboard

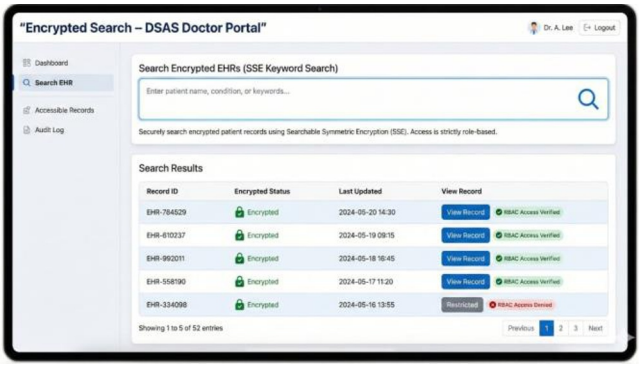

Fig. 5. Encrypted Search Interface in the DSAS Doctor Portal

The key implementation components contained in DSAS are as follows:

Implementation Highlights:

Backend: Spring boot (REST APIs, RBAC(Role Based Acesss, metadata manage)

Frontend: React.js (interface for patient and doctor interface)

Database: MySQL (to store encrypted metadata, index search)

Storage: AWS S3 (encrypted EHR file) Security Components:
AES-256-GCM (file encryption)

RSA-2048 (key wrapping)

HMAC- SHA256 (search tokens audit chaining)

BCrypt (password storage)

This implementation will make sure that not only is DSAS theoretically secure, but it is also practical and can be deployed in real clinical settings. The technology stack used reduces operational costs, provides high security assurances, and ensures high response times needed in healthcare processes.

5 Security Analysis

ASDS will ensure confidentiality, fine grained access control, secure key word searching, and tamper evident auditing with the combination of AES-256-GCM, RSA key wrapping, SSE, RBAC and HMAC log chaining. All encryption and key processing is done on the client side, i.e. plaintext EHRs and unwrapped keys are never revealed to the DSAS server or cloud provider.

- Data Confidentiality: All EHR files are uploaded encryptedwith the use of AES-256-GCM ensuring that the cloud or backend does not access plaintext information. RSA-2048 keys are used to wrap AES keys, therefore, only authorized users holding the matching private key can decrypt them.
- The fine-grained control: Access is cryptographically imple-mented, but not policy-based. Patients provide access through rewrapping AES keys to the particular users, and the RBAC means that only the authenticated and role-permitted users could request decryption or search operations.
- Secure Revocation: On revocation, DSAS recreates a newAES key, recrypts the file, and rewrapping of keys is done only to the other authorized users. Revoked users are unable to decrypt subsequent EHR updates, even in the event they have old ciphertext or metadata.
- Search Privacy: Searchable Symmetric Encryption (SSE) isan algorithm that allows looking up without exposing plaintext keywords. Only authorized document id is returned by the server as the id matches HMAC-derived tokens. Even plaintext privacy of key words is maintained although search and access patterns might leak.
- Auditability and Integrity: Authentication is done usingAES-GCM to identify any alteration of encrypted EHRs and HMAC-chained audit logs provide tamper evidence of all activities. Any alteration of the logs discredits the chain, which favors the forensic validation and the regulatory conformity.

All in all, DSAS offers a balance between security and efficiency as it does not need heavy cryptographic algorithms like ABE, PRE, and blockchain but it still ensures confidentiality, authorized search, revocation security, and verifiable auditing.

6 Experimental Results

The prototype of the performance of DSAS was tested on the background of Spring Boot, MySQL, and AWS S3. Experiments were conducted on encryption overhead, encrypted search latency, cost of key management and audit log performance. The EHR files that were used to carry out all tests were in the range of 10 KB to 10 MB.

- Encryption Performance: AES- 256- GCM was shown tobe linearly scalable to file size. An EHR 5 MB encryption took around 150 ms, whereas text records (less than 1MB) took 40 ms. This overhead is insignificant in comparison with the cloud upload time and it proves the appropriateness of the symmetric encryption to the clinical workflow which is real-time.
- Search Latency: SSE index was experimented using data of10,000 to 50,000 encrypted records. The detection of keywords in the search always took between 50 and 75 ms, since the lookups are based on deterministic HMAC tokens and indexes of the database. This will guarantee prompt retrieval in the diagnosis and emergency care.
- Key Management : The wrapping of RSA keys took 3-5ms per user in access granting. Revocation was done through AES key rotation and re-encryption and took less than 2 seconds on a 1,000 files, much faster than ABE-based and PRE-based methods, which incur expensive policy review or proxy computation.
- Audit Logging: HMAC chained audit enabled 1 ms/entryoverhead, which allowed events to be logged with high rate without affecting system throughput.
- On balance, it can be said that DSAS provides high security and low computation cost. The encryption, search, and revocation can be performed within less than a second, which proves the feasibility of the DSAS in implementation into cloud-based healthcare.

7 Discussion

The DSAS system proves that cloud-based EHR safety does not entail the intensive use of cryptography tools like ABE, PRE, or blockchain to obtain powerful privacy assurances. DSAS offers lightweight, but efficient security model to be utilized in hospitals and telemedicine systems by integrating AES encryption, RSA key wrapping, search with SSE,

and RBAC authorization. Results of the experiment have also verified that encryption, search, and revocation have sub-second latency, which is much faster than the traditional attributebased or proxy re-encryption methods, which are characterized by high latency and complicated key management.

The major strength of DSAS is that it has patient-centric entry and exit model where the owners of the data have full control of who is allowed to access their medical files. This is in line with the new standards of healthcare privacy which have focused on user privacy and consent-based data sharing. The embedded HMAC-chained audit system also promoted the accountability of the system, allowing tracking of all operations in a system transparently and without tampering.

Although it has its benefits, DSAS also shares the same limitations as its underlying SSE scheme, primarily, search and access patterns leakage. Though this does not reveal plain text keys,adversaries may be able to deduce frequency or other correlation of queries. Reducing such leakages, e.g. by the use of ORAM or query obfuscation, is a promising point to improve. Moreover, big scale deployments can necessitate streamlined key distribution processes to promote multiinstitutional healthcare settings.

On the whole, DSAS offers a sensible balance between security and performance and usability, having answers to numerous practical-world challenges faced by the cloud-based EHR systems.

8 Conclusion And Future Work

The current paper introduced DSAS, a lightweight and useful architecture of secure EHR sharing and authorized encrypted search in a cloud-based healthcare system. DSAS can offer high confidentiality, access control, high search performance, and tamper-evident accountability by applying AES-256-GCM encryption, key wrapping in RSA, key search in SSE, RBAC, and audit logs that are chained to HMAC. In contrast to the current ABE-, PRE-, or blockchain-based solutions, DSAS does not require performing extensive cryptography and complicated infrastructure, which is why it can fit the conditions of a real clinical setting where speed, usability, and scalability are crucial. As it was experimentally found, the framework at hand achieves low encryption cost, fast secure search and efficient revocation, which makes its application to modern healthcare setting legitimate.

Future directions in this area will involve maximizing leakage reduction in SSE schemes by incorporating ORAM or query obfuscation schemes to enhance further the privacy of searches. Generalizing DSAS to federated multi-hospital setting, adding automated anomaly detection of abnormal access patterns, and adding lightweight homomorphic

analytics of privacy-preserving computation are also ways in which this can be improved to increase its level of security and applicability. In general, DSAS offers a solid background of safe, patient-centric data exchange and will be a promising milestone in the implementation of privacy in cloud-based ehealthcare systems.

References

1. Song, X., Wagner, D., Perrig, A.: Practical techniques for searches on encrypted data. In: Proc. IEEE Symp. Security and Privacy, pp. 44–55 (2000)
2. Curtmola, R., Garay, J. A., Kamara, S., Ostrovsky, R.: Searchable symmetric encryption: Improved definitions and efficient constructions. In: Proc. ACM Conf. on Computer and Communications Security (CCS), pp. 79–88 (2006)
3. Li, M., Yu, S., Zheng, Y., Ren, K., Lou, W.: Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption. In: IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 1, pp. 131–143 (2013)
4. Ateniese, G., Fu, K., Green, M., Hohenberger, S.: Improved proxy re-encryption schemes with applications to secure distributed storage. In: ACM Transactions on Information and System Security, vol. 9, no. 1, pp. 1–30 (2006)
5. Chen, L., Lee, W. K., Chang, C. C., Choo, K. K. R., Zhang, N.: Blockchain-based searchable encryption for electronic health record sharing. In: Future Generation Computer Systems, vol. 95, pp. 420–429 (2019)
6. Yuan, X., Wang, X., Wang, C., Wang, Q.: Enabling efficient and secure keyword search over encrypted data in cloud. In: IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 2, pp. 353–365 (2016)
7. Wang, D., He, D., Shen, J.: Towards efficient and secure cloud-based EHR system using searchable encryption. In: Journal of Medical Systems, vol. 42, no. 12, pp. 1–9 (2018)
8. Zhang, Y., Deng, R. H., Bertino, E., Zheng, D.: Efficient attribute-based access control with authorized search for encrypted cloud data. In: IEEE Transactions on Information Forensics and Security, vol. 12, no. 7, pp. 1595–1608 (2017)
9. Gupta, B. B.: Blockchain-assisted fine-grained searchable encryption for cloud-based healthcare cyber-physical systems. In: IEEE/CAA Journal of Automatica Sinica, vol. 8, no. 12, pp. 1972–1985 (2021)
10. Shen, M., Zhang, H., Zhu, L., Xu, K., Yu, N.: Secure phrase search for intelligent processing of encrypted data in cloud-based IoT. In: IEEE Internet of Things Journal, vol. 5, no. 3, pp. 1949–1959 (2018)
11. Balaji, A., Sathyasri, B., S, V.V.R., Indumathy, D., Krishnan, R., Vanaja, S.: Intruder Alert System in Smart Home based on IoT Technique. (2022). https://doi.org/10.1109/icpects56089.2022.10047243.
12. Wang, C., Wang, Q., Ren, K., Lou, W.: Privacy-preserving public auditing for data storage security in cloud computing. In: Proc. IEEE INFOCOM, pp. 1–9 (2010)

13. Sinthia, P., M, Malathi., T, Sripriya., Krishnan, R., G, Gurumoorthy., Jalaldeen, K.: Monitoring vital parameters of comatose patients using smart sensors integrated with cloud storage. (2024). https://doi.org/10.1109/i-smac61858.2024.10714845.
14. Liu, Y., Zhang, Y., Tian, Y., Ma, J.: Efficient and privacy-preserving keyword search on encrypted cloud data. In: IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 11, pp. 2758–2769 (2014)
15. Yang, Z., Li, M., Lou, W., Hou, Y. T.: Preserving privacy in cloud-assisted healthcare systems. In: Proc. IEEE INFOCOM Workshops, pp. 1–5 (2011)