IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⵘ Impact Factor 7.047 ⵘ Vol. 10, Issue 7, July 2022

DOI: 10.17148/IJIREEICE.2022.10711

# A Patient Health Information Exchange Using Aws S3 Service

**Madhu Sudhan K N[1], B M Bhavya[2]**

PG Student, Dept of MCA, P.E.S College of Engineering Mandya, Karnataka, India[1]

Assistant professor, Dept of MCA, P.E.S College of Engineering Mandya, Karnataka, India[2]

**Abstract:** Health information exchange (HIE) has several impressive advantages for patient care, including raising the standard of medical treatment and accelerating the coordination of care. In order to transfer data ownership from providers to patients, the Office of the National Coordinator (ONC) for Health Information Technology is looking for patient-focused HIE ideas. The existing system faces several obstacles to patient-centric HIE, including worries about security and privacy, inconsistent data, and slow access to the appropriate information across numerous healthcare facilities. This project uses the distinctive property of the AWS technology, which is regarded as "extremely secure," to propose a workable solution to these problems. We created an AWS architecture to safeguard patient privacy and data security, maintain data provenance, and give patients complete access to their medical information. This design achieves patient-centric HIE by personalizing data segmentation and creating an "allowed list" for clinicians to access their data. This patient-centered HIE method assessed the model's viability, stability, security, and robustness statistically.

**Keywords:** AWS S3 Service, Electronic Health records (EHR), Health Information Exchange (HIE), Medical Prescription.

## I. INTRODUCTION

In the USA, non-federal acute care hospitals have adopted electronic health record (EHR) systems at more than 96 percent. Health information exchange (HIE) that is timely between healthcare systems has a significant positive impact on lowering costs, enhancing care, and bolstering disease in charge. [4] Billions of dollars have been spent by the Office of the National Coordinator (ONC) for Health Information Technology to promote the creation of HIE systems and achieve meaningful use of EHR. One collaborative association of business entities, such as state-wide hospital systems, has successfully achieved HIE. However, different types of HIE present issues with data security, patient privacy, and patient engagement. Additionally, there are recent indications that interoperability will now be patient-centered. [5] Although one of the three currently available HIE forms, consumer-mediated exchange, enables patients to access and manage their health information online, in order to achieve a truly patient-centric HIE, patients must have full control over their data, including authorizing healthcare facilities' access, determining what information is shareable, acknowledging the use of the data, and approving the life cycle of the shared data.

## II. LITERATURE SURVEY

This article used a blockchain-based infrastructure to exchange primary patient health information. The system has two modules: linkage and request. The linkage module connects the EHR databases with the blockchain by creating touchpoints to index the records in the future. The request module enables patients to grant clinicians permission to access their data through the blockchain and to request records by selecting touchpoints through the blockchain [1]. A blockchain-based health chain proof-of-concept with a patient-centric focus was presented in this study, The suggested architecture encourages patient involvement and makes it easier for patients and clinicians to communicate secure, mediated information, In order to reduce resource usage, eliminate data fragmentation, and enable in-place editing, redactable patient blocks using chameleon hashing were created the information sharing approach is based on PRE, smart contracts, HL7 FHIR, our 2PD PRE scheme, and signature techniques [2]. In this work, a novel zero-fee, zero-miner, and zero-block paradigm for the secure and private storage and exchange of health-related data was developed. In this study, the main Tangle transaction performance metrics and the MAM Protocol (masked authenticated messaging) were assessed and explained [3].

## III. PROBLEM STATEMENT

In a typical patient's health information that has paper-based documents, we may encounter numerous issues such as document loss and security breaches, and data can be readily manipulated by an unknown individual. A centralized health information exchange framework will address these issues. These issues weaken patient treatment records and lead to loss of patient treatment information.

## IV. PROPOSED SYSTEM

The system must create a programme that will aid in protecting the patient's medical information. The patient details and treatment information are saved in the database, and this programme enables healthcare practitioners to view and share patient medical record data securely. Treatment information for each patient is encrypted using the AES Rijndael and Shamir algorithms. The database is then filled with the encrypted data that was written to the QRCode picture and uploaded to the AWS S3 service. In order to view the specifics of the patient's therapy, a doctor or nurse request and access key are created. The doctor or nurse can examine the treatment specifics via decrypted data when AWS S3 retrieves the encrypted data from the QRCode image and verifies the access key.

Patients receive prescription information from doctors. These prescription specifics are entered into the database and written to PDF. For the purpose of seeing patient prescription information, a pharmacist request and access key are generated. The pharmacist can download the prescription PDF saved in the database after validating the access key. The pharmacist can sell the medications to the appropriate patients based on the patient's ID after downloading the prescription. To view the specifics of the patient's prescription and treatment, a patient request and access key are made. Patients can examine the treatment details and download the prescription after the access key has been verified.

### A. System Architecture

The application manager or admin will include significant actors in this architecture, such as hospitals with their respective hospital names, pharmacies with their respective pharmacy names, departments, and the medical tests. The actors will receive their user IDs and passwords through email once the application manager has added them to the application. These passwords and IDs will be generated programmatically and delivered to the actors. They will be able to change their passwords, which will be recorded in the database, and we will use the SMTP protocol for mail reasons.

The hospital staff adds the doctor's and patient's information, and the doctor can access or search the database to view the patient's medical history. Before recommending medications, the doctor consults with the patients. Prescriptions are saved in the AWS S3 service as PDF files. the results of the patient's medical tests are saved in the AWS S3 service by the hospital personnel. The patient can then use their Patient ID to view the prescription, download it, and purchase their medications directly from the pharmacy.

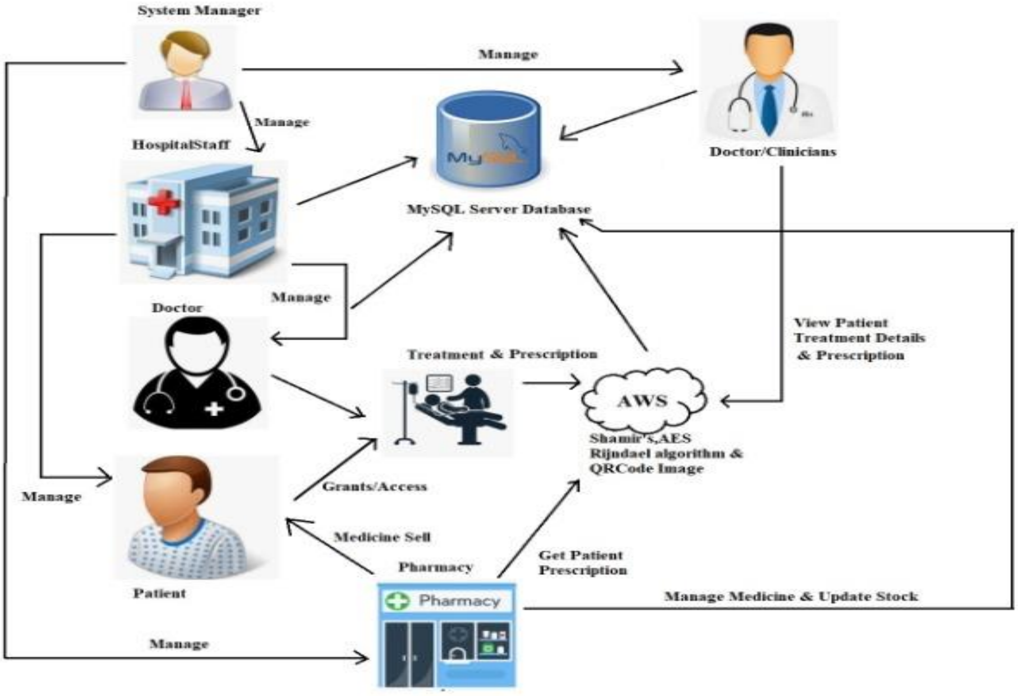

Figure 1: System Architecture

DOI: 10.17148/IJIREEICE.2022.10711

## V. IMPLEMENTATION

The programme will be updated with all of the patient's medical tests and prescriptions, and a report will be produced. These patient treatment data will be saved in the application in hash format. The Shamir's Method will be used to divide the data into several pieces, and the hash will then be sent to the AES encryption algorithm as well as a few other entities. Ten rounds of the AES encryption method will be completed before a 128-bit key is produced. The AWS S3 Services will keep this key on file.

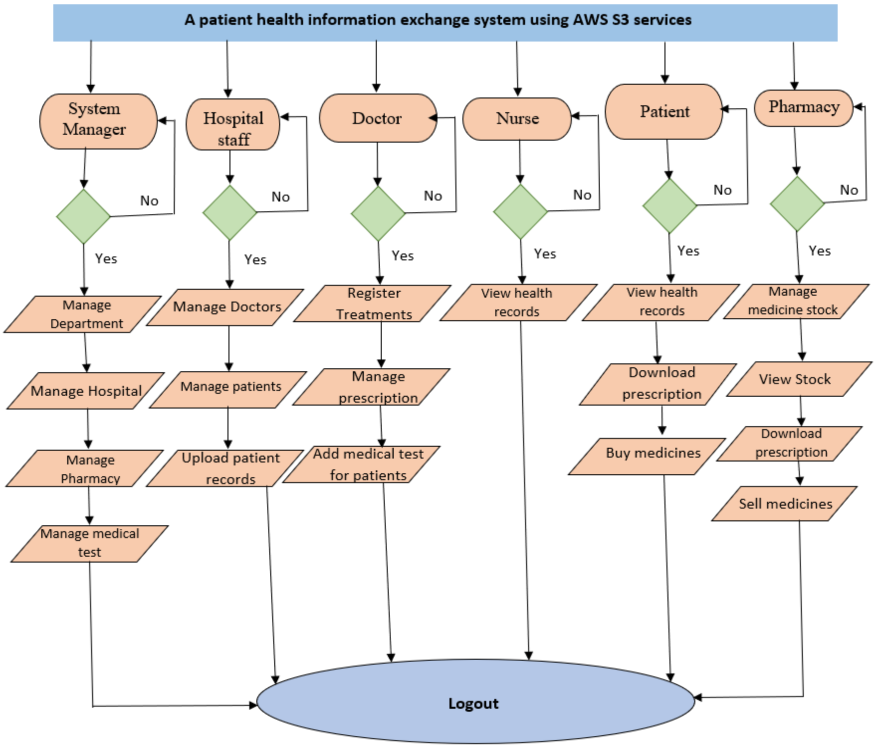

Figure 2: Workflow diagram

### A. AES Algorithm

The symmetric key cryptography method AES was released by MST. Rijndael came up with the algorithm. The Rijndael encryption algorithm is another name for it. AES serves as DES's substitute. AES makes use of the block cypher method. Both cypher text and plain text must have the same size. An input key's size is equivalent to that of plain text.
Three alternative key lengths—128, 192, and 256 bits—as well as data lengths of 128, 192, and 25 bits are supported by AES. AES uses multiple rounds of processing to work with various key bits, such as encryption that is carried out 10 times using a 128-bit key. 12 encryption rounds are carried out by the 192-bit key. Data is encrypted for 14 cycles using the 256-hit key.

**AES Algorithm steps:**

1. By looking up a fixed table, the 16 input bytes are substituted (sub bytes) (S-box). A matrix with four rows and four columns represents the outcome.
2. Rows are shifted: The four rows of the matrix are all moved to the left.
3. Mix Columns: A unique mathematical formula is now used to alter each four-byte column. The four bytes of one column are entered by this function, which returns four entirely new bytes that replace the original column. The outcome is another new matrix with 16 more bytes. It should be noted that the final round does not include this phase.
4. Add round key: The 128 bits of the round key are XORed with the 16 bytes of the matrix, which are now regarded as 128 bits. The output is the ciphertext if this is the final round.

Encryption phase
During encryption, the entire block is rotated by 0°, 90°, 180°, or 270° clockwise, depending on the parity bit value derived from the round key, which is 00, 01, 10, and 11 accordingly (Table 1).

| Key bit value | Rotation (Clockwise direction) |
|---|---|
| 00 | 0° |
| 01 | 90° |
| 10 | 180° |
| 11 | 270° |

Table 1: Rotation Policy while Encryption

The operation modifies the 2D array's 2D array's information's relative location. The updated step can be carried out by applying the mathematical operations covered below:

Think of the initial array of bytes as a 2D array M of size, let's say nxn, where n is the array's number of rows and columns. On such an array, a 90° rotation can be considered a two-step mechanism.

Rotation of 90° = rCOOM (M′)

where the procedure reverseColumnsOrderOfMatrix() merely reverses the order of the matrix's columns. Consider, for instance, a 4x4 matrix M. In this case, the 90° rotation can be thought of as follows:

$$\begin{array}{c|cccc} & \multicolumn{4}{c}{\text{Column}} \\ \text{Row} & 1 & 2 & 3 & 4 \\ \hline 1 & A & B & C & D \\ 2 & E & F & G & H \\ 3 & I & J & K & L \\ 4 & M & N & O & P \end{array}$$

$$\text{rCOOM (M′)} = \begin{array}{c|cccc} & \multicolumn{4}{c}{\text{Column}} \\ \text{Row} & 1 & 2 & 3 & 4 \\ \hline 1 & A & B & C & D \\ 2 & E & F & G & H \\ 3 & I & J & K & L \\ 4 & M & N & O & P \end{array}$$

The transposed matrix M's row order can also be reversed to achieve a rotation of 270 degrees. It can be represented mathematically by the operations covered below.

Rotation of 270° = rROOM (M′)

The two basic types of operations—transpose and reversal of row/column order—are all that are needed to perform the procedures described above in MATLAB or any other programming language.

$$M= \begin{array}{c|cccc} & \multicolumn{4}{c}{\text{Column}} \\ \text{Row} & 1 & 2 & 3 & 4 \\ \hline 1 & A & B & C & D \\ 2 & E & F & G & H \\ 3 & I & J & K & L \\ 4 & M & N & O & P \end{array}$$

$$1\ \text{rROOM (M′)} = \begin{array}{c|cccc} & \multicolumn{4}{c}{\text{Column}} \\ \text{Row} & 1 & 2 & 3 & 4 \\ \hline 1 & D & H & L & P \\ 2 & C & G & K & O \\ 3 & B & F & J & N \\ 4 & A & E & I & M \end{array}$$

**Decryption phase**

There are two ways to carry out operations during the decrypting stage. The first method allows for the use of identical bits with the elements rotated counterclockwise at the angle used for encryption. During the decryption process, the key value bits can be changed to directly represent the angle of rotation for clockwise rotation. By taking the 2's complement of the key bit values for rotation and adding 1 to it (ignore carry), one can convert key bit value to obtain key value that represents angle of rotation. The specifics of clockwise rotations utilising computed bits are provided in Table 3 for the decryption procedure.

| **Key value** | **Rotation (anti-Clockwise direction)** | **OR** | **Rotation (Clockwise direction)** |
|---|---|---|---|
| 00 | 0° | | 360°-0°=360°=0° |
| 01 | 90° | | 360°-90°=270° |
| 10 | 180° | | 360°-180°=180° |
| 11 | 270° | | 360°-270°=90° |

Table 2: Rotation Policy while Decryption (anti-clockwise OR Clockwise

| **Key value** | **2's Complement+1 (Ignore carry)** | **Rotation (Clockwise direction)** |
|---|---|---|
| 00 | 11+1=00 | 0° |
| 01 | 10+1=11 | 270° |
| 10 | 01+1=10 | 180° |
| 11 | 00+1=01 | 90° |

Table 3: Rotation Policy while Decryption (clockwise rotation)

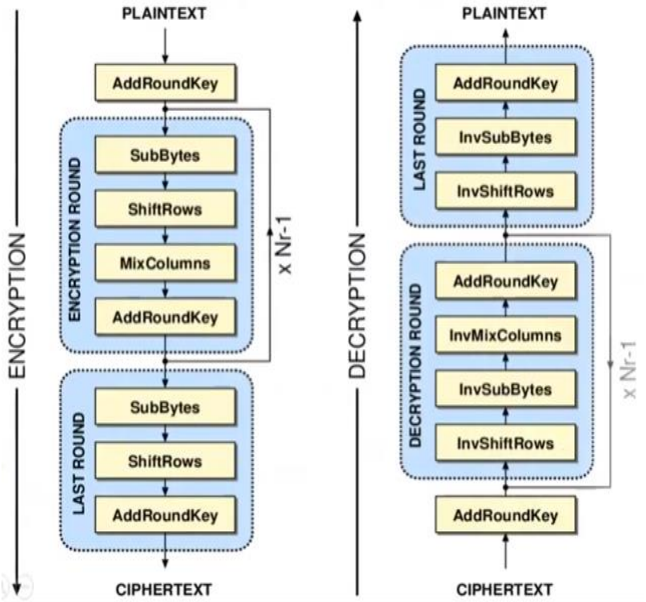

Figure 3: AES encryption and decryption

**B. Shamir's Algorithm**

Adi Shamir developed the cryptographic algorithm known as Shamir's Secret Sharing. This algorithm's primary goal is to separate the secret into multiple distinct pieces so that it can be encrypted.

- Say S represents the secret that needs to be encoded.
- There are N pieces to it: S1, S2, S3,..., Sn.

DOI: 10.17148/IJIREEICE.2022.10711

- The user then selects a number K to decrypt the pieces and reveal the original secret once it has been divided.
- The secret S cannot be reconstructed with (K - 1) parts or fewer, hence it is chosen in a way that prevents us from discovering it if we only know K parts.
- If we are aware of K or more components from S1, S2, S3,..., Sn, we may easily compute or reconstruct our secret code S. This is sometimes referred to as a (K, N) threshold scheme.

We can discover the linear polynomial axe + by = c for the two points (x1, y1) and (x2, y2) that are supplied. Similar to that, we can discover the quadratic polynomial for the three locations provided.

ax2 + bx + cy = d

The goal is to construct a polynomial of degree (K - 1) in which the constant term is the secret code and the other numbers are random, and in which the constant term can be determined by utilising any K of the N points that can be produced from this polynomial using Lagrange's Basis Polynomial.

For instance: Let S = 65, N = 4, and K = 2 be the secret code.

1. Initially, we construct a polynomial of degree (K - 1) in order to encrypt the secret code.
2. Let y = a + bx be the polynomial as a result. The constant portion 'a' in this case is our secret message.
3. Any random number, like b = 15, will do as b.
4. We therefore produce N = 4 points from this polynomial, y = 65 + 15x.
5. These four points should be (1, 80), (2, 95), and (3, 110). (4, 125). It is obvious that we may create the beginning polynomial from any two of these four locations, and the constant term an in the resulting polynomial serves as the necessary secret code.

## CONCLUSION

The study has suggested a novel technique for maintaining confidentiality and authentication while keeping patient medical records (Patient Treatment Data) secret and only accessible by medical professionals with the patient's permission. The method is helpful for medical case studies and for training new nurses and doctors. The main goal in this case is to encrypt patient treatment data using Shamir's, Rijndael algorithm, QRCode, and AWS S3 Service.

## REFERENCES

[1] Zhuang, Yan, et al. "A patient-centric health information exchange framework using blockchain technology." IEEE journal of biomedical and health informatics 24.8 (2020): 2169-2176.

[2] Soni, Mukesh, and Dileep Kumar Singh. "Blockchain-based security & privacy for biomedical and healthcare information exchange systems." Materials Today: Proceedings (2021).

[3] Abdullah, S., Arshad, J., Khan, M.M. et al. PRISED tangle: a privacy-aware framework for smart healthcare data sharing using IOTA tangle. Complex Intell. Syst. (2022).

[4] Hylock, Ray Hales, and Xiaoming Zeng. "A blockchain framework for patient-centered health records and exchange (HealthChain): evaluation and proof-of-concept study." Journal of medical Internet research 21.8 (2019): e13592.

[5] Jabarulla, Mohamed Yaseen, and Heung-No Lee. "A blockchain and artificial intelligence-based, patient-centric healthcare system for combating the COVID-19 pandemic: Opportunities and applications." Healthcare. Vol. 9. No. 8. MDPI, 2021.