International Journal of Inventive Engineering and Sciences (IJIES)
ISSN: 2319-9598 (Online), Volume-12 Issue-1, January 2025

Enhancing Privacy in the Management of Library Resources: A Novel Approach Utilizing FRS and ABE-CP Algorithm for Improved Protection

Muhammad Aliyu, Lele Mohammed

Abstract: ***Libraries contain sensitive resources that users can access. The vital challenges faced include unauthorized access, privacy violations, malicious attacks, etc. Researchers have explored several ways to curb these challenges including encryption, policies, anti-malware etc. An improved design to secure library resources based on Fragmentation Redundancy Scattering (FRS) and Attribute-Based Encryption Ciphertext-Policy (ABE- CP) was presented. The core idea of FRS is to fragment confidential information to produce insignificant fragments and then scatter the pieces across the distributed system. As such, the scheme provides a new technique to manage network components and exchange encryption keys. By comparing the proposed scheme with other schemes, the proposed scheme prevents unauthorized access to users' data and protects user's privacy. The scheme in addition becomes resistant to common attacks like replay and man-in-the-middle. According to our findings, the scheme is safe, practicable, and comprehensive with high security for both encrypting and decrypting users' data within the least possible time.***

Keywords: ***Fine-grained Access Control, Library, Privacy, ABE-CP, FRS, Resources.***

I. INTRODUCTION

Due to the fast evolution of Information and Communication Technology (ICT), network capabilities now have a greater impact on people's lives since they facilitate communication between individuals at any time and from any location. One of the key challenges faced by IT in the modern days Smart Library Management Systems (SLMS) is privacy protection [1]. A multitude of chances for diverse economic, social, and cultural research studies might arise from the dissemination of SLMS. In addition to these advantages, is the compromise of user data. When registering on SLMS, users need to make sure they provide their personal information accurately. Users must have complete control over this information and any other information they choose to share. Privacy is the right of individuals to control the classification and removal of their personal information at will and to the extent of controlled disclosure of personal information [2]. In this paper, privacy is the protection of users' private data and their shared data against any unauthorized access by any natural or legal person. The lives of SLMS actors may suffer negative or damaging effects if private user information is disclosed [3]. To receive hitch-free services on the current SLMS, users must consent to the privacy settings of the SLMS. Giving the service provider ownership of all content is one of the usual terms of service that users must accept [4]. Stated differently, the ownership agreement grants the service providers complete permission to distribute and utilize user content in any way they see fit [5].

As stated also in [6], some relevant and important requirements about the concept of privacy and services needed by users that should be observed include auditing, fairness, integrity, availability, access control, communication privacy, and validation. A categorization to date was conducted in [7] where privacy protection techniques to reduce threats include anonymity, Information security, fine-grained access control and privacy settings, change in users' behavior and user awareness, and encryption and decentralization. The authors concluded that encryption is the best tool for maintaining the confidentiality of users' shared data on the SLMS network. Encryption does not only confidentially store users' data on the SLMS networks' storage server, but it also silences the reception or transmission of user data on the network. All the encryption techniques studied in the research only considered a bulk type that could be easily read by SLMS network servers. Therefore, in this research, end-to-end encryption will be considered using [8] FRS and ABE- CP [9] techniques for improving security and privacy in SLMS networks.

The rest of the paper is organized as follows. Section 2 talks about a review of past literature in areas of security techniques in SLMS networks by other authors section 3 describes the proposed scheme and section 4 analyzes the proposed scheme in terms of security and privacy of users' data. Section 5 assessed the efficiency of the proposed scheme and finally, section 6 concludes the entire work shading more light on future work.

II. REVIEW OF PAST LITERATURE

In recent years, several encryption techniques have been proposed to protect privacy or online data sharing. For example, a Two-Layer Encryption (TLE) method for data loaded in the cloud environment was seen in [10]. This TLE technique uses an access

Manuscript received on 19 September 2024 | First Revised Manuscript received on 12 December 2024 | Second Revised Manuscript received on 31 December 2024 | Manuscript Accepted on 15 January 2025 | Manuscript published on 30 January 2025.

***** Correspondence Author(s)

Muhammad Aliyu*, Department of Computer Science Engineering, School of Science and Technology, The Federal Polytechnic Bauchi, (Bauchi), Nigeria. Email ID: maliyudeba@gmail.com, ORCID ID: 0000-0001-6155-3133

Lele Mohammed, Department of Computer Science Engineering, School of Science and Technology, The Federal Polytechnic Bauchi, (Bauchi), Nigeria. Email ID: leleringhanny@gmail.com

control policy. Also, a scheme for multi-authority attribute-based access control in the cloud was purported [11]. The scheme's architecture comprises a Certificate Authority (CA) and other issuing authorities. A proposed lightweight secure data-sharing scheme for cloud processing was purported in [12]. The technique made use of ABE-CP to control access by transferring the bulk of the computational load in ABE-CP to external proxy servers from mobile devices. A different approach using access control based on a combination of Role-Based Access Control (RBAC) with ABE-CP was seen in [13]. This special access control model has an effective key update function in a data outsourcing environment.

Another method that controls encrypted access using ABE was seen in [14]. This method, called Persona, allows users to make categorization into different groups, which sets a different access level for each group. Their technique protects the privacy of users of online community libraries. A modified persona that seeks to create the possibility of revocation by modifying ABE called EASiER was seen in [15]. The method uses a third party to decrypt end-to-end connectivity by providing dynamic control access. A scheme that outsourced decryption and post-decryption recovery algorithm using ABE with the Outsourcing decryption and Match-then-decrypt ABE-CP algorithm (OM-ABE-CP) was purported in [16]. The proposed scheme was deployed on a cloud-based encrypted data-sharing scheme on SLMS. The scheme is based on the Decisional Bilinear Diffie-Hellman (DBDH) assumption and proves to be a secure structure against Replayable Chosen-Ciphertext Attacks (RCCA).

A method that uses public key RSA encryption as well as AES symmetric key encryption algorithm was proposed in [17]. The developed framework uses an end-to-end encryption method. Hide In The Crowd (HITC) is another flexible encryption-based access control technique that provides privacy and security for users' names which allows users to decrypt library resources that they borrow. The technique in [18] is based on access control with an appropriate granularity level. HITC can be connected to any existing SLMS platform which was developed as a browser plug-in and without the need for a third-party server. This method uses the RSA encryption method. A purported scheme to increase privacy and security of SLMS networking systems was also seen in [19]. The purported techniques' security scheme is based on AES and Elliptic Curve Cryptosystem (ECC). The scheme supports replaying of attack denial and forgery attack denial using the Elliptic Curve Digital Signature Algorithm (ECDSA) and timestamps. An offline key agreement process between users under the Computational Diffie-Hellman (CDH) assumption was designed by short-term periodic key updates. Another proposed method for SLMS in mobile devices was seen in [20] which are easy to implement with low complexity. The technique provides a fresh take on symmetric encryption for mobile devices' SLMS. This approach encrypts data using a secret key and a series of prime integers that are derived from a bi-dimensional matrix.

A privacy strategy for mobile SLMS networks was developed in [21] founded on ABE-CP feature-based encryption and access control. The authors used secure proxy decryption to handle a publish-subscribe system, which not only offers a safe access control mechanism but also ensures users' privacy about their shared material and trustworthiness. Furthermore, sharing the content with this system uses less energy. Additionally, ABE-CP and a hierarchical blockchain-based solution were created for SLMS privacy [22]. This technique can safeguard users' data on semi-honest online networking servers and offer a selection of fine-grained features. The blockchain structure reduces storage use in their plan. According to [23], a blockchain-based approach answers user requests in a fair, accurate, and secure manner and offers a safe keyword search algorithm on online networks based on a hybrid blockchain and public-key encryption. Furthermore, this method's keyword search technique is secure, dependable, and doesn't require any kind of validation. [24] Looked at the concerns surrounding online SLMS network security and privacy from the standpoint of potential threats. They categorized the different kinds of attacks and looked at how to carry them out, existing defenses, and related issues. The study aimed to highlight areas of overlap and draw a distinction between security and privacy. In addition, they looked for vulnerabilities in services like location-based services that may be abused in various SLMS.

Although encryption has its limits, encryption-based techniques are appropriate for safeguarding users' privacy. [25] illustrate the limitations of encryption. Tools and techniques like Reliable Computing, Hardware Resistance, Modules Trusted Platform, Trusted Hardware, and Software Logging Tools were invented to safeguard and increase privacy to get beyond these restrictions and not solely employ encryption.

Offered users, particularly those utilizing SLMS networks for distant learning, intelligent privacy protection [26]. The location privacy and user trajectory privacy in SLMS networks are more vulnerable in these kinds of applications. In this study, users' location data is removed, and their privacy is preserved by offering a list of anonymous locales. According to their methodology, SLMS network privacy levels are first gathered using anonymous sets and then split according to weighted sensitivity. The tool that determines the quantity of lost information and the standard function to evaluate it is then used to design the best anonymous. Lastly, borrowers are given access to the anonymous trajectory collecting for privacy through the use of differential privacy algorithms. Their method's flaw was that it neglected to take the online network's dynamic character into account.

A trajectory privacy protection technique based on spatial-temporal limitations was presented in [27] for SLMS. By creating and sharing this data, users can contribute to road condition assessments, route design, and even user behavior. An excellent opportunity for a link attack is created by this. Therefore, privacy disclosure may be avoided by protecting public route data. Their approach stops data from leaking when publishing trajectories. To create a dummy trajectory, they first stop using sensitive places as pivots in their method.

Stated differently, the trajectory's overall movement direction and pivot selection are restricted to guarantee alignment with the trajectory's overall movement direction. To make sure that the generated dummy trajectory has the same movement pattern as the real trajectory, the algorithm is then created based on the spatial-time constraints algorithm, initial dummy locations, and trajectory algorithm. They demonstrated how their approach strikes a compromise between trajectory privacy and data availability.

Described a technique that protects users' privacy by making it impossible to be tracked. During data release, it ensures unlinkability on networks [28]. They made use of unlinkability in the publishing of weighted network data. Two privacy models were provided by the method: node unlinkability and edge weight unlinkability. They also altered a graph's structure using edge randomization to thwart structure attacks. A significant benefit of their approach is that it reduces information loss by minimizing the change in the data structure without removing the crucial edges.

To identify suspicious linkages within the user communities, [29] offered a classification approach based on mutual clustering coefficient and user profile data. They used a variety of commonalities, including employment, schooling, hometown, and favorites, to contrast and identify questionable connections. Through the number of features based on the mutual cluster coefficient and user profile data, they were able to derive a formula that allowed them to identify questionable personalities. Additionally, their suggested approach can alert librarians to dubious users. An SLMS network service provider can use this methodology to identify phony users.

III. THE PROPOSED SCHEME

There are six phases in the proposed design, three of which are the primary phases. These three stages consist of data encryption, data decryption, and key exchange. The remaining three involve fragmenting the encrypted keys, creating redundancies, and scattering the key across the network. The following will provide a detailed description of the proposed strategy.

End-to-end encryption is achieved in this approach using two encryption phases. The first encryption stage is of the ABE type with ABE-CP [9] and the second stage is FRS type [8]. This scheme uses the first stage to prevent the SLMS network server from knowing what is being exchanged, and the second stage to implement fine-grained access control. Additionally, this scheme has a unit known as the CA that is independent of the online network and is not under its control. This CA does not have access to the users' primary key, but it is in charge of producing and distributing certain cryptographic keys. The task for this unit will be fully explained in the sections that follow.

The system is made up of a peer-to-peer network of linked devices, where each node can only communicate with other nodes that are within its communication range. With the CA, therefore, every node has the same standing. In return for being able to save and retrieve their data within the system, nodes are expected to contribute a certain amount of storage capacity to the network. Any node that provides storage space to users will be referred to as a storage node and any node that wants to save or receive a file from the users will be referred to as a client node. The information that must be stored in the system must be encrypted, fragmented, and dispersed, all of which are the responsibility of the client.

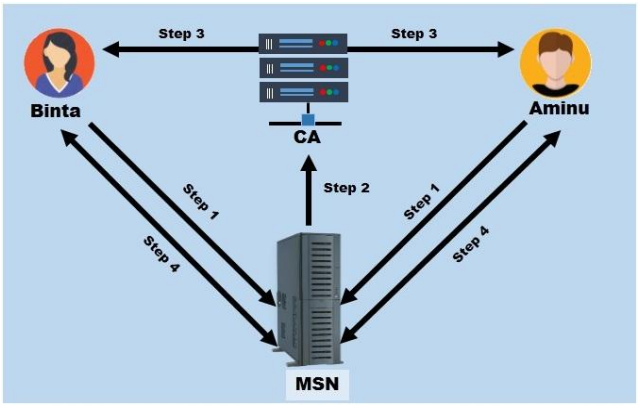
The ABE-CP encrypts files in their entirety. Since the data is meant to be private and readable by the publisher(s) alone, ABE-CP is required. The ciphertext output of the encryption procedure is used in the fragmentation process. The ciphertext is divided into pieces that are all the same size. Every one of these pieces has a distinct identity attached to it. The fragment's name and its contents have no discernible relationship. In this instance, the client is the only one who checks integrity. Before storing, storage nodes must hash each fragment's content and compare it to its name; if the two do not match, the fragment should be rejected. The fragment name cannot be communicated to system storage nodes that are operating correctly if a fragment has been maliciously or unintentionally corrupted. Once the data is in the system, the client has very little influence over it. The overall number of fragments and performance overhead will rise with the number of files kept in the system over time. Each fragment might have a lifetime value attached to it to prevent this issue. Although storage nodes may have an upper limit on this value, the client can specify this lifespan value. The information saved in the system must also be retrieved, assembled, and decrypted by the client.

Gathering the pieces from the network is the initial stage of the retrieval procedure. For a given record, the client views the list of fragment names that were saved during the fragmentation process. By comparing the content hash to the fragment name, the client can confirm the integrity of the fragment data. If any fragments are corrupted, they should be collected again. After obtaining all required fragments, the fragment data needs to be reassembled using the ordering information that was preserved throughout the fragmentation process, and in the correct sequence. With the same encryption key, the reassembled data can be decoded and should match the ciphertext used throughout the encryption process.

A. Registration of Users

Authenticating a cell phone number is a prerequisite for individuals to register on the SLMS, such as Aminu by Binta. The user receives an authentication code after registering his/her mobile number with the SLMS network, which he/she can use to verify the legitimacy of his/her number. The mobile phone number of the user is sent to the key generation center (CA) and is entered into its database once it has been verified. Because it is unique, each user's mobile phone number sets them apart from one another.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)



[Fig.1: The Proposed Scheme Architecture]

The architecture of the suggested scheme is depicted in Fig. 1. Step 1: sends user requests to MSN SLMS for registration and communication; Step 2: sends user requests to CA for ABE-CP key creation; Step 3: CA creates ABE-CP keys and sends them to the user; Step 4: exchanges user main keys and facilitates communication and data sharing.

B. Proposed Scheme for Key Exchange between Users

Key exchange is done only to exchange users' main key, used for symmetric encryption. This step is carried out in the following two ways:

1. Two users, each of the other's community lists that want to exchange the main key with each other are key actors in this phase. It should be noted that this phase may be one-way, meaning that one of the users does not want the other user to know about his or her data and that there is only one-way communication between the two users.
2. New key dissemination: A friendly bilateral relationship is assumed in the description of key exchange in the following steps:

Step 1

The MSN server receives the key exchange request from Binta and Aminu. The two users do not have to be online at the same time at this point. Additionally, each user has the option to request a separate transmission of their main key to the MSN server. Assume that the key exchange request is sent simultaneously by two users who are online.

Step 2

Next, the MSN server requests that the CA server deliver each user their private ABE-CP encryption key. It does this by sending this request to the CA server.

Step 3

The ABE-CP private and public keys are then generated by the CA server and sent to users, say; Binta and Aminu. The CA obtains these keys from users directly through SLMS networks and independently of each other just once while the users are present on the network. By CA, Binta and Aminu exchange their public keys while keeping their private keys hidden from their network provider. Aminu receives his ABE-CP private key and Binta's public key from CA, and Binta provides Aminu with her ABE-CP private key. In actuality, the parties have agreed on communication via exchanging users' public keys, and they are now prepared to disclose their primary keys.

Step 4

Binta now generates her main key (KEY_A), encrypts it for symmetric encryption, and sends it to Aminu over the MSN server. She does this by utilizing Aminu's public FRS key, which was supplied to her by the CA server in Step 3.

Step 5

Aminu obtains Binta's primary key, KEY_A, by using his own ABE-CP private key to decrypt the encrypted key that Binta sent, which was produced by the CA server. Aminu utilizes Binta's primary key, which he has stored on his device, to perform symmetric decryption during data exchange.

Step 6

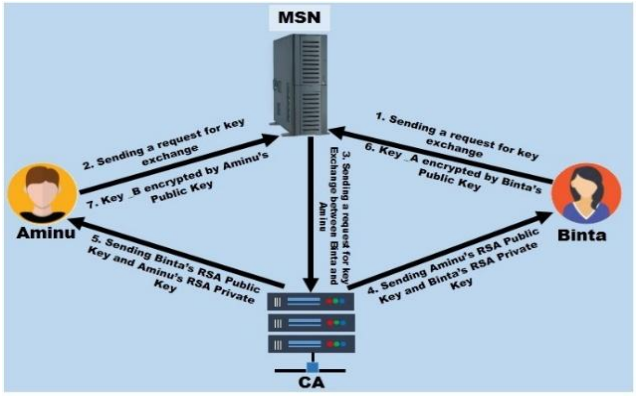
Aminu now encrypts his primary key (KEY_B), which he generates independently using Binta's public FRS key that the CA server gave to him in Step 4 and transmits to Binta over the MSN server. Aminu uses KEY_B for symmetric encryption.

Step 7

Aminu's encrypted key, KEY_B, is obtained when Binta decrypts it using her own ABE-CP private key that was produced by the CA server. Binta utilizes Aminu's primary key, which she has on her device, to perform symmetric decryption during data exchange.

Following the aforementioned procedures, both users now possess the other's primary key. They exchange data using one user's key for symmetric encryption and the other user's key for symmetric decryption. The user's mobile device stores these keys, prohibiting access to the users' primary keys by the MSN and CA servers. It should be mentioned that the CA server knows the FRS keys and will be notified of the contents of the key if the main keys encrypted with FRS are sent through it. This is the reason for transmitting the user's main key via the MSN server using FRS encryption.

However, because it is unaware of the FRS keys, the MSN server is unable to access the user's primary key. The most crucial aspect of consumers' privacy is this system, which safeguards their private keys.



[Fig.2: Users Exchange of the Main Key]

Algorithm 1. Exchange Primary Keys.

1: Begin

2: Binta submits a "Key Exchange" request to MSN.

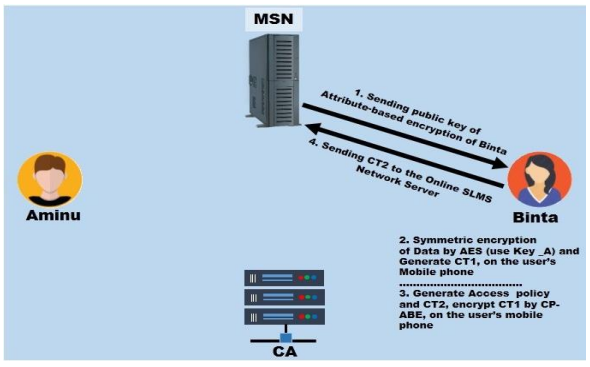
3: Aminu sends a "Key Exchange" request to MSN.
4: MSN verifies Binta and Aminu's user IDs.
Send "IDBinta and IDAminu" to CA if ((IDBinta and IDAminu) = exist);
else
Abandon requests and proceed to Step 10
End if
5: CA creates Binta and Aminu's ABE-CP keys and distributes them to them
6: Binta uses Aminu's public ABE-CP key to encrypt "KEY_A" before sending it to Aminu.
7: Aminu uses Binta's public ABE-CP key to encrypt "KEY_B," which he then sends to Binta.
8: Aminu decrypts and saves "KEY_A."
9: Binta stores "KEY_B" after decryption.
10: End

The suggested scheme's main key exchange between users is seen in Fig. 2. Algorithm 1 also displays the functions each user used to swap keys in the sequence in which they were executed.

C. Encryption of Data

The proposed system performs two-way encryption phases on data a user wishes to share. FRS is utilized in the system for symmetric encryption in the first stage, and ABE-CP is employed in the second. We arrive at end-to-end encryption after these two steps. To shield user data from the network server, (symmetric) encryption is used in the first phase. In addition to performing encrypted access control, the attribute-based second stage of encryption keeps unauthorized users from accessing user data. The data encryption in the suggested design will be explained in full and step-by-step in the sections that follow. The steps for encryption are as follows:

1. Using the primary key, the user encrypts the data or the identical PlainText (PT) using FRS. This step's output is referred to as CT1.
2. ABE-CP encrypts the data that was encrypted in the first phase. The encrypted access policy uses this step, along with the subsequent steps: (a)
3. The user requests the network server for the ABE-CP public key. (b)
4. The user's public key is generated by the network server.
5. The user creates the desired access policy.
6. The user encrypts the encrypted data of the first stage, CT1, using the public key obtained from the network server and the access policy that was created. This step's output is referred to as CT2.
7. CT2 is transmitted to the network server.



[Fig.3: Proposed Encryption Process]

Fig. 3 shows the encryption process used by the proposed scheme. In Algorithm 2, execution of the data encryption phase is shown in the order of execution steps.

Algorithm 2. Encryption of Data
1: Begin
2: FRS encrypts Binta "PT" using main "KEY_B"
then from PT, "CT1″ is taken
3: Request to MSN server by Binta for ABE-CP public key
4: ABE-CP public key is Generated by the MSN server then Send it to Binta
5: Access Policy Generated by Binta
then policy added to "CT1″
6: "CT1″ encrypted by Binta using ABE-CP public key From"CT1″, "CT2″ is taken
8: From the MSN server, Binta Sends "CT2″
9: "CT2″ is stored by MSN server
10: End

D. Data Decryption

To be granted access to shared data, a user must send a request to the SLMS network server. Users share documents, audio files, movies, images, and messages with one another. The methods for requesting and decrypting data are as follows:

1. Aminu initially requested Binta's info from the network server.
2. Aminu has previously allowed the network to access KEY_ATT_B, his ABE-CP private key, which is needed to decrypt the data in the first stage. It should be noted that even while users have access to the KEY_ATT_B key through the network, the user still needs to give the SLMS network their key to make this step clear.
3. In the first phase, the network server uses the ABE-CP key to decrypt Binta's data. The public and private keys for ABE-CP are generated by the network server. Mean ABE-CP, or the first phase of decryption, is handled by the network server. To decode the data from the first steps, the network server uses the ABE-CP private key that the MSN server generated. At this point, if the KEY_ATT_B key satisfies the so-called data access policy, the network can decrypt the data from the initial steps. Aminu now receives CT1, which is not the original raw data as of yet.
4. Aminu eventually completes symmetric decryption and retrieves Binta's original raw data, which is the final PT, using Binta's primary key, KEY_A, which was obtained during the key exchange phase.

Fig. 4 shows the data decryption procedure for the proposed system. Algorithm 3 shows the sequence in which the data decoding step is carried out.

Algorithm 3. Data Decryption.
1: Begin
2: "CT2″ Requested by Aminu from MSN server
3: "KEY_ATT_B" Sent by Aminu to MSN server
4: "KEY_ATT_B" Checked by MSN server
if ("KEY_ATT_B" = Legit) then
"CT2″ is Decrypted by "KEY_ATT_B"
then "CT1″ Extracted from "CT2″

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

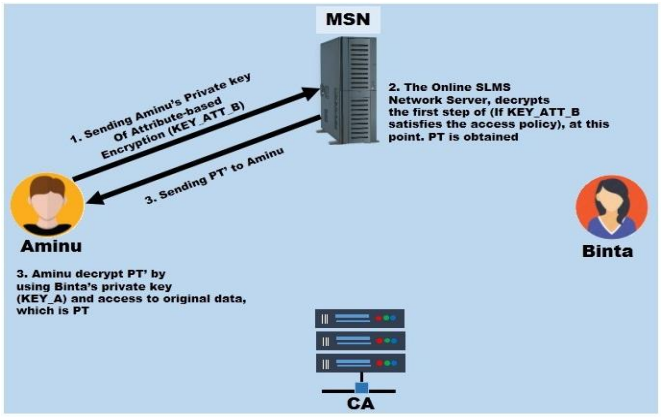
5: “CT1″ Sent to Aminu from MSN server

6: Aminu uses the main key "KEY_B" to decrypt "CT1." then “PT” extracted from “CT1″

7: “PT” Stored by Aminu

8: End

Lastly, our method offers fine-grained access control and end-to-end encryption, shielding user data against misuse and privacy infringement by unapproved users and SLMS network providers.



[Fig.4: Decryption of Data]

IV. PRIVACY AND SECURITY APPRAISAL

This section will assess the suggested scheme's security and privacy from many angles [30]. Among the most crucial problems are how to prepare and share the keys among users and SLMS network components [31], since encryption approaches have been utilized in the past [32]. As a result, the following thoroughly reviews and assesses these difficulties [33]. The schemes published in the literature will also be compared in this section with the proposed scheme; these authors will be referred to as Shalini, Wang, Pozo, Brenna, He, and Halevi, respectively.

A. Key Management Evaluation

Under the suggested plan, keys are transferred securely, and each participant is aware of a certain set of keys that guard against illegal access and preserve user privacy and security. The primary key, the FRS public key, the FRS private key, the ABE-CP private key, and the ABE-CP public key are the five keys that each user owns. As a result, instructions on how to create, utilize, access, and transfer each of these keys will be provided here.

- *FRS Public Key:* This key is one of the five that the CA created for the user. The users' primary key is encrypted using this key. Users who want to send their main key to someone who has a public FRS key are given access to this key. The SLMS network server cannot tamper with the creation, alteration, or transmission of the FRS public key; instead, it is transmitted only to authorized users via the CA. With the help of this transfer scheme, users can't receive phony FRS public keys instead of genuine things, and the SLMS network server is kept in the dark about what is included in the keys that are transferred.

- *FRS Private Key:* This key, which is likewise generated by CA, is one of each user's five keys. Users can decode a key that their friends have encrypted by using this key. Only the user receives the FRS private key from the CA. The SLMS network server cannot tamper with the creation, alteration, or transfer of the FRS private key because it is transmitted to the user through the CA. By using this transmission mechanism, the SLMS network server cannot access the key, decode it, or receive notification of it. As a result, neither the user's primary key nor the FRS private keys are accessible to the network server.

- *Main Key (FRS Key):* In the suggested arrangement, the main key is the third of the user's five keys. The user created this key. There is no contribution from the network server, CA, or other users to its creation. Data unique to each user is encrypted and decrypted using this key. Every user encrypts their data and sends it to their pals using a unique primary key. After receiving the encrypted material, friends use this key to decode it as well. The user states that only authorized users have access to this key. It is unknown to the SLMS network server, CA, and unauthorized users and they cannot access it. This key is transferred via the SLMS network server.

- *ABE-CP Public Key:* The fourth user key is this one. In the suggested approach, this key is utilized for the second layer of encryption. ABE-CP is the second layer. The network server generates this key; neither the user nor the CA may alter its derivation. Users encrypt their second stage of data using this key. The network server only gives this key to the data owner; other users are not aware of it. Additionally, CA does not have access to this key or know what is inside of it. To keep CA and other users from accessing this key, it is also sent to the user via the network server.

- *CP-ABE Private Key:* Out of the five keys that the user has, this is the last one. In addition to the SLMS network server and CA, no other user not even the key owner-participates in the creation of this key. Based on the characteristics of each user, a unique key is created for them. This key is used to decode the encrypted data using ABE-CP, allowing the data to be decrypted provided the data access policy is satisfied by the user's hidden attributes in the private key based on his attribute. This key belongs to the user, who is the only one with access to it; other users are not privy to it. Considering that this key is shared via the network.

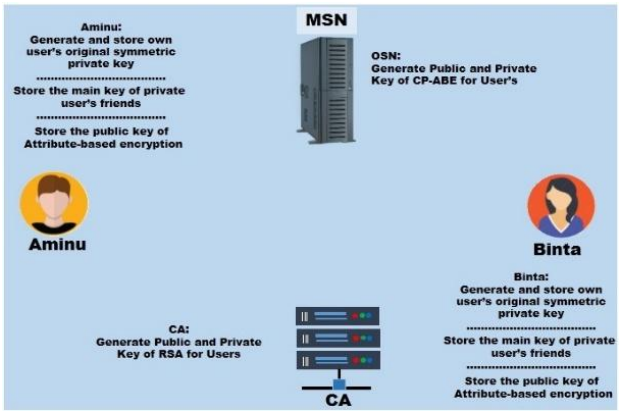
Now that all five users' keys have been examined, it is clear which key is utilized to safeguard users' privacy. The SLMS network server cannot access the users' primary keys thanks to the use of FRS private and public keys. The primary key is used to obstruct the network server's access to user-exchanged data content. Additionally, public and private CP-ABE keys were utilized to implement fine-grained access control, stop unauthorized users from accessing user data, and stop CA from accessing user data. The specs of the five keys used by the users are compared in Table 1.

Table 1: Five Keys' User Specifications Comparison

Type of Key	Key Producer	Operation of Key	Authorized User to Access	Unauthorized User to Access	Communication Platform
FRS public key	CA	Main key Encryption	The user and all friends own key	Unauthorized users & MSN	CA
FRS private key	CA	Main key Decryption	Key owned by User	Unauthorized users, MSN and CA	CA
Main key (FRS key)	User	Encryption and decryption of data at the first step	All users and Users who authorize access by his key.	Unauthorized users, MSN, and CA	MSN
ABE-CP public key	MSN	Encryption of data at the second step	Key owned by the user	All users and CA (except the key owner)	MSN
ABE-CP private key	MSN	Decryption of data at the second step	Key owned by the user	All users and CA (except the key owner)	MSN

B. Proposed Scheme Privacy Assessment

The access of different components of the proposed scheme to the five user keys is depicted in Fig. 5. Figure 5 shows which elements of the suggested scheme are aware of and have access to users' keys. The following examines every scenario that can arise about user privacy protection:



[Fig.5: Privacy Assessment]

▪ *CA:* There are two reasons why the CA cannot be notified about the contents of the data if it wants to access user data. The first explanation is that users communicate encrypted data over network servers; the CA is not privy to this data transmission. The second reason is that, even if the CA can access user data, it will not be able to learn the primary contents of the data since it is unaware of the private keys to the second layer of encryption, or CP-ABE.

▪ *MSN:* MSN cannot access the original users' data if it wishes to access user data since it is unaware of the private keys for the symmetric encryption used in the first layer of encryption. Users' primary keys are transmitted to each other over a network server and encrypted using FRS encryption.

▪ *Unauthorized User:* Users' data cannot be accessed by unauthorized people. The unauthorized user does not have access to any private encryption keys or the first or second layer of encryption. The ABE-CP private key is not generated for a user who is not the data owner since they do not have the data owner's primary key and are not in the data owner's community.

▪ *Fine-Grained Access Control:* Fine-grained access control is best done by giving attributes to users and classifying them into various groups and categories by employing ABE-CP. Unauthorized users are prevented from accessing private data by fine-grained access control.

▪ *Data Confidentiality:* End-to-end encryption ensures complete preservation of data confidentiality. There is no requirement for a reliable SLMS network server in this system.

In this section, our proposed scheme will be compared with the other schemes in terms of meeting security and privacy requirements and common attacks, such as replay attacks, man-in-the-middle attacks, support for protection against online network providers, secure key exchange, end-to-end data encryption, Fine-grained access control. We present comparison results in Table 2.

Table 2: Security Requirements Comparison

Scheme	Prevention of Replay Attacks	Prevention of Man-in-the-Middle Attacks	Protection Against Online Network Providers	Secure Key Exchange	End-to-End Data Encryption	Fine-grained Access Control
Shalini	✓	–	✓	✓	–	✓
Wang	✓	✓	✓	✓	–	–
Pozo	✓	✓	–	–	–	–
Brenna	–	–	–	–	–	–
He	✓	✓	✓	–	–	✓
Halevi	✓	✓	✓	–	–	✓
Proposed	✓	✓	✓	✓	✓	✓

Table 2's comparison results demonstrate how secure and very practical our approach is.

V. PERFORMANCE ASSESSMENT

The efficiency of the suggested plan will be evaluated in this section. Java Pairing-Based Cryptography (JPBC) and the Java Development Kit serve as the foundation for the implementation of the suggested system. The Samsung Galaxy A50 Android phone, which has an Octa-core processor, 8 GB of RAM, and the Android 9 operating system, was used to evaluate the suggested scheme's implementation.

The size of the messages that the users exchanged was the first factor. Because exchangeable messages and data on SLMS networks are often modest and most online networks have less than 30 MB of data to share and exchange data with sizes ranging from 0 to 30 MB were chosen for evaluation.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

The number of attributes that users in ABE-CP utilized for their shared data in the access policy constituted the second factor. The majority of SLMS users use 0 and 100 attribute values, as such it should be noted that the Pairing-Based Cryptography (PBC) library and the competent toolkit offered in Bethencourt are the sources of the ABE-CP that is being used.

Additionally, the users' primary keys were encrypted using FRS-2048 encryption. These keys may have a bit length of 128 192 or 256, depending on the FRS algorithm type that is chosen. To compare our technique with the others, the 128-bit FRS encryption time of the keys was left out.

A. Performance Evaluation Based on the Size of Data Exchanged Between Users

The proposed scheme was compared with Shalini's', Halevis', and Hes' schemes in terms of the amount of data exchanged between users for shared data to assess encryption and decryption times. A fixed value of 30 was taken into consideration by each user for the shared data across all sizes, as this quantity is near the majority of features that are provided by other online networks. Less than 5, 5, 10, 15, 20, 25, and 30 MB worth of data were also utilized. Following a 4-time run of our experiments to gather evaluation data, mean values were computed. The encryption process was completed at an interval of 10 turns, 20 turns, 30 turns, and 40 turns respectively. As such, the proposed scheme's encryption time is very short making it suitable for mobile online networks. Results can be seen in Figures 6-9.

The proposed scheme on average performs encryption operations 1.5 times quicker than the Hes', 1.4 times quicker than the Halevis', and 2.5 times quicker than the Shalini s' scheme in different data size modes as can be seen from figures 6-9.

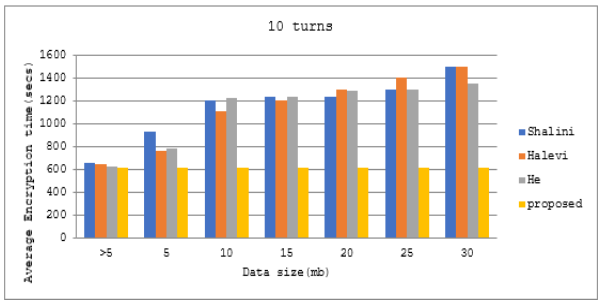


Fig 6: Average Encryption Time of 10 Turns

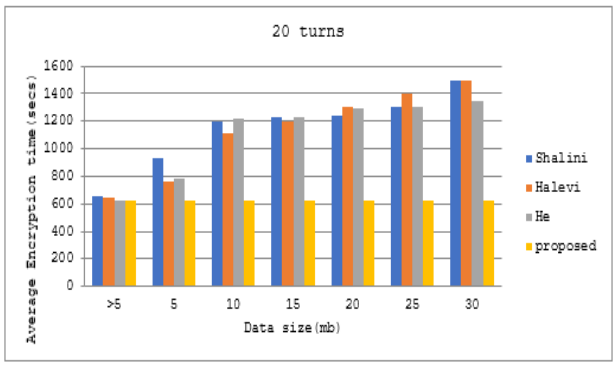
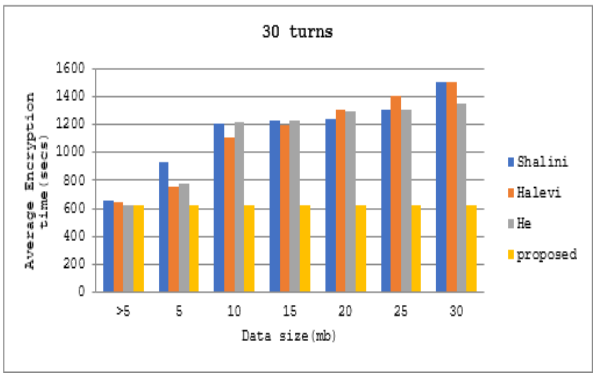
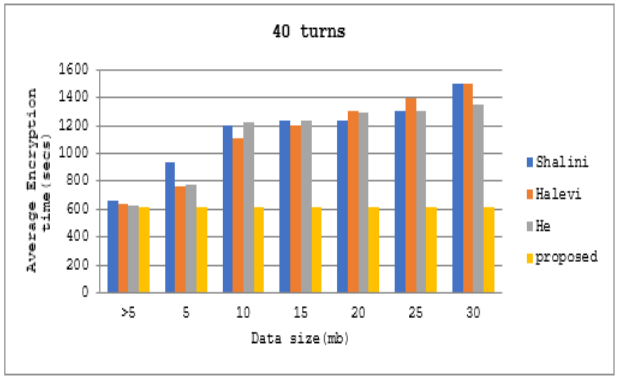


Fig 7: Average Encryption Time of 20 Turns



[Fig.8: Average Encryption Time of 30 Turns]



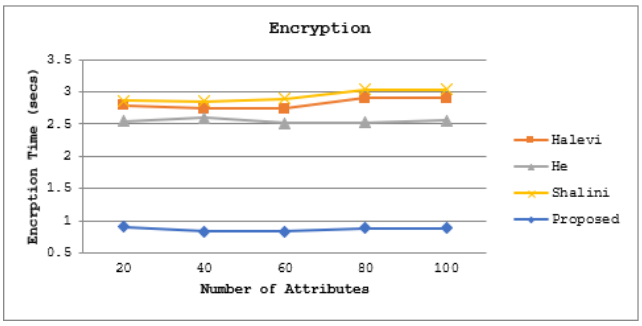
[Fig.9: Average Encryption Time of 40 Turns]

Meanwhile, on average, the time taken to encrypt is the same time taken to decrypt data while maintaining the same parameters.

B. Performance Evaluation Based on Attributes of Data Shared by Users

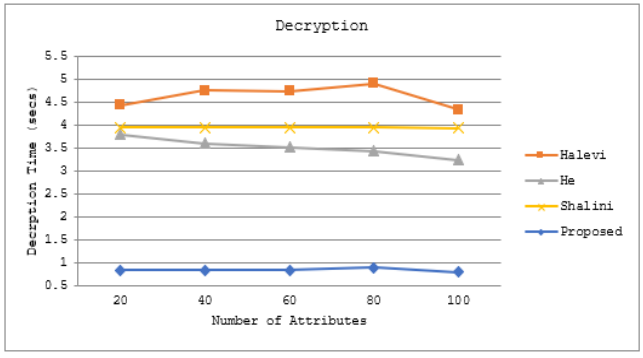
The proposed system was compared to schemes (Shalini', Halevis, and Hes') in terms of their execution timings based on the leaves in the access policy tree to assess encryption and decryption times. Furthermore, it was decided that the quantity of data transferred would always be 15 MB. The ABE-CP is based on the number of leaves in the access policy tree; the performance of the schemes was assessed using the values of these leaves on the same characteristics, which were evaluated as 0, 20, 40, 60, 80, and 100.

Based on the quantity of shared attributes in the user's access control policy for data, Fig. 9 shows the data encryption time in the suggested system. The findings indicate that adding more features lengthens the data encryption time. The proposed approach has the fastest encryption time, as shown in Fig. 9. In various modes concerning the number of characteristics, the proposed strategy executes the encryption operation 1.05 times faster than the Halevis' scheme, 1.79 times faster than the Hes' scheme, and 4.1 times faster than the Shalini s' scheme on average. In the proposed technique, the user encrypts data using FRS before ABE-CP, which also shortens the encryption time overall.



[Fig.10: The Data Encryption Time Based on Attributes Number]

Based on the number of features in the user's key, the decryption times of the suggested scheme and the other schemes stated are contrasted in Fig 10. It should be noted that because attribute-based decryption takes place on the online network server, the data decryption proposed time only comprises the symmetric encryption phase. For a 15MB data size, the number of characteristics has no bearing on the decryption time; so, the proposed method decryption time is fixed. But in other methods, adding more properties to the user's private key also lengthens the decryption time. As a result, while comparing decryption times, our approach performs faster and more effectively than the benchmarked schemes (Halevi, He, Shalini). In various configurations concerning the number of attributes, the proposed technique executes the decryption operation on average 4.4 times faster than the Halevis' scheme, 3 times faster than the Hes' scheme, and 3.6 times faster than the Shalini' scheme as can be seen from fig 11 below.



[Fig.11: The Data Decryption Time Based on Attributes Number]

VI. CONCLUSION

A completely encrypted privacy scheme for SLMS networks was presented in this paper. All things considered, it is a safe, end-to-end encrypted method for exchanging user data on SLMS while safeguarding user privacy. Within our method, users can readily exchange encryption keys with one another at any time. FRS was applied in this manner to ensure the safe transfer of the primary encryption keys for data. The user was not subjected to a large overhead in the implemented simulations. With our strategy, privacy may be safeguarded in every way while still providing a safe, workable, and accessible framework that makes use of the current encryption techniques. Based on the obtained results, the proposed scheme offers a good level of security, according to the security study, when compared to similar researched schemes. In the meantime, the experimental findings showed that it outperforms other techniques in terms of user data encryption and decryption times. Apart from safeguarding the privacy of consumers, the proposed scheme is extremely secure, useful, and implementable. Future work will focus on the error recovery process by the FRS in case some portions of the fragments get corrupted along the way.

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/ Competing Interests:** Based on my understanding, this article has no conflicts of interest.
- **Funding Support:** This article has not been sponsored or funded by any organization or agency. The independence of this research is a crucial factor in affirming its impartiality, as it has been conducted without any external sway.
- **Ethical Approval and Consent to Participate:** The data provided in this article is exempt from the requirement for ethical approval or participant consent.
- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Authors Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. Bal, R., Ashish K., Surendra, K., (2023), 'Applications of Internet of Things In Library and Data Management', IP Indian Journal of Information Technology, DOI: https://doi.org/10.18231/j.ijlsit.2023.003
2. Panjun, S., Yi, W., Zongda, W., Zhaoxi, F., Qi, L., (2025), 'A survey on privacy and security issues in IoT-based environments: Technologies, protection measures and future directions', Computers & Security, Volume 148, January 2025, 104097, Elsevier, https://doi.org/10.1016/j.cose.2024.104097
3. Kumar, A., Kumar, R., Pandey, S.K., (2023), 'Analysis of the Collection Development Policies: A Case Study of the Libraries of Kumaun University', Indian Journal of Library Science. 2023;2(5):54–9. DOI: https://doi.org/10.18231/j.ijlsit.2023.003
4. Aliyu, M., Murali, M., Gital A., Boukari, S., Rumana, K., Maryam, A., Zambuk, Z., Caleb, C., Ibrahim, M., (2021). 'A Multi-Tier Architecture for the Management of Supply Chain of Cloud Resources in a Virtualized Cloud Environment: A Novel SCM Technique for Cloud Resources Using Ant Colony Optimization and Spanning Tree', International Journal of Information Systems and Supply Chain Management Volume 14 Issue 3, DOI: https://doi.org/10.4018/IJISSCM.2021070101
5. Aliyu, M., Murali, M., Gital, A. Y., Boukari, S., (2019), 'An Efficient Ant Colony Optimization Algorithm for Resource Provisioning in Cloud', International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-4, DOI: https://doi.org/10.35940/ijrte.D6968.118419
6. Aliyu, M., Murali, M., Zhang, Z., Gital, A., Boukari, S., Huang, Y., Yakubu, I. (2021), 'Management of Cloud Resources and Social Change in a Multi-Tier Environment: A Novel Finite Automata Using Ant Colony Optimization with Spanning Tree'. *Technological Forecasting and Social Change*, 166, 120591. DOI: https://doi.org/10.1016/j.techfore.2021.120591
7. Oukemeni, S., Rifà-pous, H., Manuel, J., and Puig, M., (2024), 'Privacy Analysis on Microblogging Online Library Networks', A Survey', 52 (3) (2024) *ACM Computing Surveys*, 52(3), 1–36. DOI: https://doi.org/10.1145/3321481
8. Mu, C. (2024), 'Application of Optimizing Advanced Encryption Standard in Vehicle Controller Local Area Network Bus Secure Communication System', *Frontiers in Mechanical*

Engineering, 10, Article 1407665. DOI: https://doi.org/10.3389/fmech.2024.1407665
9. Bethencourt, J. & Sahai, B., (2023), 'Ciphertext-Policy Attribute-Based Encryption' In *Proceedings of the IEEE Symposium on Security and Privacy* (pp. 321–334). IEEE. DOI: https://doi.org/10.1109/SP.2007.11
10. Chingath, V., & Babu, R., (2020), 'Advantage of Blockchain Technology for the libraries', International Conference on DigitaI Transformation: A Cognitive Learning towards Artificial Intelligence, www.researchgate.net/publication/341725555_Advantage_Blockchain_Technology_for_the_Libraries
11. Ram, B., & Singh, K., (2020), 'Innovative Library Services in Mobile Technology: A Recent approach'. Int J Inf Dissemination Technol. 2020; 10(4):192–4. DOI: https://doi.org/10.5958/2249-5576.2020.00035.7
12. Li, R., Chenglin, S., Heng, H., Zhiyong, X., & Cheng-Zhong, X., (2017), 'A Lightweight Secure Data Sharing Scheme for Mobile Cloud Computing', IEEE Transactions on Cloud Computing 7161 (c): 1–1. DOI: https://doi.org/10.1109/TCC.2017.2649685.
13. Fugkeaw, S., & Sato, H., (2018), 'Enabling Dynamic and Efficient Data Access Control in Cloud Computing Based on Attribute Certificate Management and CP-ABE', 2018 26th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), 454–61. DOI: https://doi.org/10.1109/PDP2018.2018.00079.
14. Starin, D., Baden, R., Bender, A., Spring, N., & Bhattacharjee, B., (2019), 'Persona: An Online Network with User-Defined Privacy Categories and Subject Descriptors', Sigcomm'09 (2019), pp. 135-146, DOI: https://doi.org/10.1145/1592568.1592585
15. Jahid, S., Prateek, M., & Nikita, B., (2021), 'EASiER: Encryption-Based Access Control in Online Networks with Efficient Revocation', *Proceedings of the 6th International Symposium on Information, Computer and Communications Security, ASIACCS 2021*, 411–15. DOI: https://doi.org/10.1145/1966913.1966970.
16. Shalini, D., Ganesh, K., & Swarnendu, K., (2024), 'Homographic Encryption Library, Framework, Toolkit, and Accelerator', SPRINGER journal, vol 5, article 24 DOI: https://doi.org/10.1007/s42979-023-02316-9
17. Ram, B., and Yadav, S., (2022), 'Bibliometric Study of DESIDOC' Journal of Library and Information Technology from 1981-2018', Libr PhilosophyPract. 2022;p. 1–13. https://digitalcommons.unl.edu/libphilprac/7457/
18. Abdulla, A., & Bakiras, S. (2019). 'HITC: Data Privacy in Online Social Networks with Fine-Grained Access Control. In *Proceedings of the 24th ACM Symposium on Access Control Models and Technologies* (pp. 123–134). ACM. DOI: https://doi.org/10.1145/3322431.3325104
19. Wang, Z., Ma, Z., Luo, S., & Gao, H., (2018), 'Enhanced Smart Library Security and Privacy Protection Scheme for Mobile Online Network Systems', IEEE Access, 6 (2018), pp. 13706-13715, DOI: https://doi.org/10.1109/ACCESS.2018.2813432
20. Pozo, D., and Iturralde, M., (2023), 'CI: A New Encryption Mechanism for Smart Library in Mobile Devices', Procedia Computer Science, 63 (2023), pp. 533-558, DOI: https://doi.org/10.1016/j.procs.2022.08.381
21. Brenna, L., Isak, S., Havard, D., & Dag, J., (2022), 'TFHE-re: A Library for Safe and Secure Remote Computing Using Fully Homomorphic Encryption and Trusted Execution Environments', ScienceDirect, vol13, DOI: https://doi.org/10.1016/j.array.2021.100118
22. Halevi, S., & Polyakov, Y., (2019), ' An Improved RNS Variant of the BFV Homomorphic encryption Scheme in Cryptographers', Track at the RSA Conference (pp83-105. Springer, Cham, 2019) DOI: https://doi.org/10.1007/978-3-030-12612-4_5
23. Omame, I., & Juliet, C., (2021), 'Application of Blockchain in Libraries and Information Systems', Research gate .net, DOI: https://doi.org/10.1108/LHTN-02-2023-0020
24. Kayes, I., Lamnitchi, A., (2017), 'Privacy and Security in SLMS Networks: A Survey', Libraries and their Applications, 3 (2017), pp. 1-21, DOI: https://doi.org/10.1016/j.osnem.2017.09.001
25. Ramu, G. & Mishra, Z., (2019), 'Hardware Implementation of Piccolo Encryption Algorithm for Constrained RFID Application', In: 2019 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference, IEEE, India, pp. 85-89, 2019. IEEE. DOI: https://doi.org/10.1109/IEMECONX.2019.8877071
26. Li, Y., Jiawen, Z., & Weina, F., (2022), 'Intelligent Privacy Protection of End User in Long Distance Education', *Mobile Networks and Applications*, no. February. DOI: https://doi.org/10.1007/s11036-022-01950-6.
27. Pujari, V., & Gadgay, B., (2023) 'Smart Library System using IoT'. International Journal for Research in Applied Science and Engineering Technology, vol. 6, no. 7, pp. 471-476, DOI: https://doi.org/10.22214/ijraset.2018.7068
28. Mohammadi, M., Yegane, M., Library, C., & Branch, Q., (2023), 'IOT: Applied New Technology in Academic Libraries', In: International Conference on Distributed Computing and High-Performance Computing (DCHP 2023) 25th-27th November 2023, Qom, At University of Qom, Qom, pp. 1-12, 2023, https://www.researchgate.net/publication/333634140_IOT_Applied_New_Technology_in_Academic_Libraries
29. Philosophy, L., Kumar, R., & Kaliyaperumal, K., (2022), 'Applications of GSM Technology for Documents Identification in a Library System', International Journal of Academic Library and Information Science, vol. 2, no. 1, pp. 1-6, 2022, https://academicresearchjournals.org/IJALIS/PDF/2014/January/kumar%20and%20%20Kaliyaperumal.pdf
30. He, Z., Cai, Z., Han, Q., Tong, W., Sun, L., & Li, Y., (2016), 'An Energy Efficient Privacy-Preserving Content Sharing Scheme in Smart Library Networks', Personal and Ubiquitous Computing, 20 (5) (2016), pp. 833-846, DOI: https://doi.org/10.1007/s00779-016-0952-6
31. Churi, P. (2019). Performance Analysis of Data Encryption Algorithm. In International Journal of Recent Technology and Engineering (IJRTE) (Vol. 8, Issue 3, pp. 3230–3235). DOI: https://doi.org/10.35940/ijrte.c5775.098319
32. Bonde, S. Y., & Bhadade, U. S. (2019). Encryption Algorithm using Shuffled 2-Dimension Key. In International Journal of Engineering and Advanced Technology (Vol. 9, Issue 2, pp. 1105–1109). DOI: https://doi.org/10.35940/ijeat.a2236.129219
33. Subraja, K., Geetha, N., & Mahesh, Dr. K. (2020). BITS – A Novel Video Encryption Algorithm. In International Journal of Innovative Technology and Exploring Engineering (Vol. 9, Issue 8, pp. 101–105). DOI: https://doi.org/10.35940/ijitee.h6196.069820

AUTHOR'S PROFILE

Muhammad Aliyu Msc (CS), PhD (CS) both from Abubakar Tafawa Balewa University Bauchi, Nigeria after he has attended Semester Abroad Program (SAP) in SRM University India is a Senior Lecturer in the Department of Computer Science, Federal Polytechnic Bauchi, Nigeria. His areas of specialization include Data communication and Computer Networks, Data Security, Network Security, Machine Learning, and Educational Technology. His research interests focus on innovative techniques for securing data and resources. Aliyu has numerous international and local journal publications and has presented his research at both local and international conferences. He is also actively involved in community service, mentoring students, and promoting digital transformation initiatives. He is passionate about leveraging technology for societal growth and fostering innovation through education and research.

Lele Mohammed Msc (CS) from Universiti Teknologi Malaysia is a Senior Lecturer at the Department of Computer Science, Federal Polytechnic Bauchi, Nigeria. Currently, Mohammed is a Ph.D. candidate in Computer Science at the Federal University Dutse, Nigeria. His areas of specialization include Artificial Intelligence (AI), Machine Learning, and Theoretical Computer Science. He has a keen research interest in developing innovative solutions to address real-world problems using advanced AI and machine learning techniques, with a particular focus on fault detection and classification in various domains. Mohammed has contributed to academic research through publications and is committed to the advancement of knowledge in the field of Computer Science. He is passionate about mentoring students and fostering the growth of technology-driven solutions to improve society.