Secure Sharing of Patient Controlled e-Health Record using an Enhanced Access Control Model with Encryption Based on User Identity

Mohinder Singh B., Jaisankar N.*
School of Computer Science and Engineering,
Vellore Institute of Technology, Vellore, Tamilnadu, India

***Abstract*—Healthcare industry is converting to digital due to the constantly evolving medical needs in the modern digital age. Many researchers have put up models like Ciphertext Policy Attribute Based Encryption (CPABE) to provide security to health records. But, the CPABE-variants failed to give total control of a medical record to its corresponding owner i.e., patient. Recently, Mittal et al. suggested that Identity Based Encryption (IBE) can be used to achieve this. But, this model used a Key Generation Center (KGC) to maintain keys that reduces the trust as the keys may get leaked. To overcome this problem, an enhanced access control model along with data encryption is presented where a separate key generation center is not needed. Because of this, the processing time for setting-up and extraction of keys is minimized. The total processed time of proposed is 74.42ms. But, the same is 92.89ms, 165.42ms, and 218.75ms in case of Boneh-Franklin, Zhang et al., and Yu et al., respectively. Our proposed model also gives a patient the complete control of his/her own health record. The data owner can decide who can access the record (full/ partial) with what access rights (read/ write/ update). The data requestors can be a doctor/ nurse/ insurance providers/ researchers and so on. The requestors are not based on groups or roles but based on an identity that is accepted by the data owner. The proposed model also withstands the key leakage attacks that are due to the key generation center.**

Keywords*—Access permissions; fine-grained access control; identity based encryption; key generation center; electronic health record

I. Introduction

Infrastructure like cloud is constantly evolving, enabling the storage of immense information manageable via various devices. The past decade has witnessed numerous developments in cloud technology, leading to its widespread application in diverse fields. One such field is the health sector.

The healthcare sector in India is currently gaining pace with technological advancements. Providing a comprehensive patient history to the doctor is crucial for accurate diagnosis, but maintaining records of every patient's past treatments is challenging. Patients often receive treatment from multiple doctors, resulting in scattered treatment details that need to be communicated to each doctor. It is a time and financial waste to recurrently perform a diagnosis with no extensive discussion, and the combination of various medications can end up in severe medical ailments. To ensure accuracy, possessing detailed medical records is essential. The current methodology of transferring information through paper or personal communication can lead to errors and potentially fatal outcomes. Electronic healthcare, also known as e-healthcare, is a solution that allows for the efficient maintenance of medical records digitally. Electronic Health Records (EHRs), interchangeably noted as Electronic Medical Records (EMRs), utilize cloud servers for high-quality infrastructure at a lower cost. However, ensuring confidentiality on top of security for digital medical information is crucial towards realizing the change of medical information from paper to digital. Among many, encryption is an effective as well as basic technique for safeguarding medical information before sending it to cloud.

In healthcare, multiple organizations and users having alike responsibilities access similar information. To analyze medical info, claim medical bills, and deliver accurate treatment, a particular patient's medical info needs to be accessed. However, as patients' medical info involves private data, it needs to be secured to prevent unauthorized usage. If not secured properly, the data may become public.

Identity-Based Encryption (IBE) was used to secure the data by encrypting as well as controlling the data. This IBE was proposed by Adi Shamir [1] in 1984 and was first implemented by Boneh and Franklin [2] in 2001. The name itself states that the encryption and decryption of the data depend on the identity of user. As it did not provide better access control of data, Attribute-Based Encryption (ABE) was introduced to achieve fine access control of data.

ABE has gained significance in recent years, with several studies exploring its potential to mitigate privacy risks. The integration of ABE with sensitive health-record sharing provides granular access and integrity maintenance. These two are crucial to provide better confidentiality and security. There are two variants of ABE exist, namely Key-Policy ABE(KPABE) and Ciphertext-Policy ABE(CPABE). Both variants of ABE provide medical-record access to various groups of users who satisfy a policy. CPABE is a promising solution towards cryptographic access control on data. With CPABE, data owners use attributes to define access policy. Data is accessed by only those users who satisfy the set policy. CPABE is considered to be more efficient than IBE in terms of granularity of access control. So, it gained importance in controlling access to healthcare data also.

Fig. 1 represents basic process of KPABE. In KPABE, attributes were associated with encryption algorithm. The access structure, which was defined to control the data, was controlled at key generation center (KGC).

So, having no control over data, data-owner cannot define

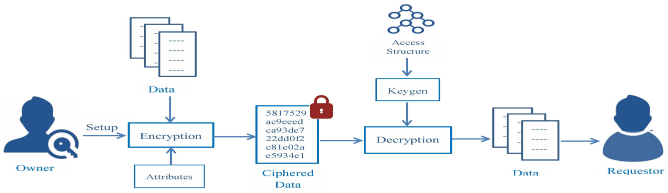


Fig. 1. Process of Key-Policy ABE.

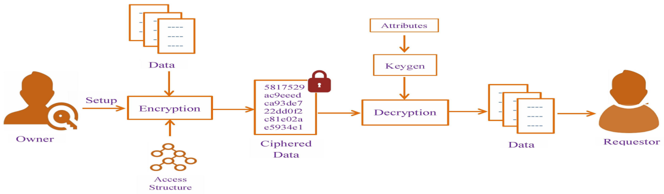


Fig. 2. Process of Ciphertext-Policy ABE.

who can access the data, up to what level with what access permissions. To overcome this, CPABE was introduced [3]. Fig. 2 depicts the generic process of CPABE. In this, access structure was associated with encryption algorithm whereas the attributes were taken care of by KGC. Also, data owner has the chance to define control over data to some extent.

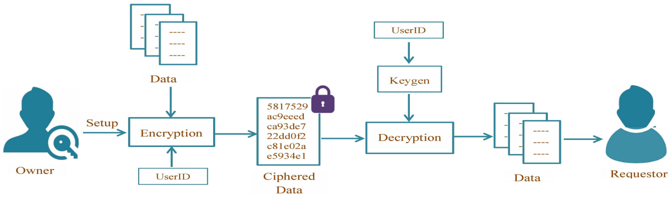


Fig. 3. Process of IBE resembling ABE.

Even though ABE was introduced after IBE, IBE can be treated as a specific kind of ABE. In IBE, only the identity of user is considered rather than multiple attributes as in case of ABE. The basic architecture of IBE in Fig. 3 resembles the ABE process.

Existing healthcare access control schemes in EHRs use CPABE to control the data. But, the data control is done at group level users or role based users, but not at the individual level. However, in scenarios where granular data access is required, such as doctors having access to all patient healthcare data while nurses and pharmaceutical firms have access to limited, insensitive data, conventional CPABE mechanisms fall short. Practical concerns such as computation requirements and security remain a major obstacle to ABE systems, as well as the increasing volume of sensitive data stored in the cloud.

Fig. 4 represents a basic digital health record system where the complete medical records of patients are stored and the users access patient record whenever needed, provided the accessing rights are satisfied. To make patients involved in this digitalization of health records, their trust is to be gained. To gain trust of the patients, EHR system is tending towards Personal Health Records (PHR) system. In PHR, the patient will be the true owner of entire personal medical record. The patient of a health record will decide the accessing possibilities for users like doctors, nurses, and so on, based on their identity but not roles or groups.

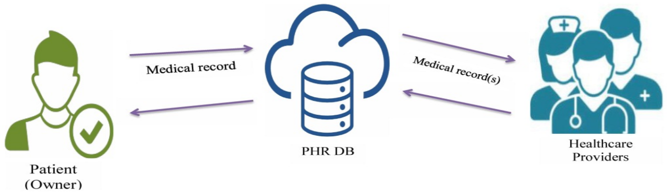


Fig. 4. Basic PHR model.

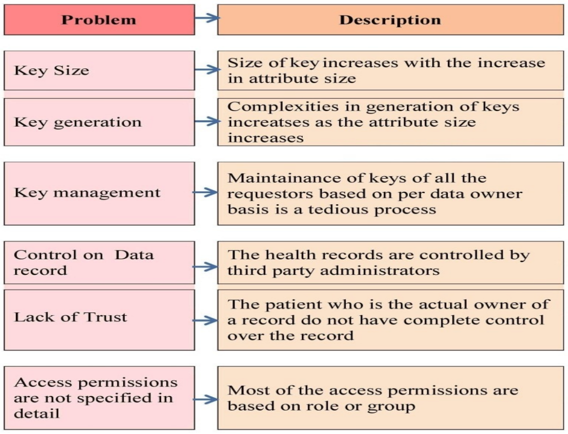
A. Motivation and Objectives

1) Motivation: The main *motivation* is to make a patient the complete owner of his/her own health record and to reduce dependency on KGC, and master key. The data owner will get the immunity to decide who can access the record (full/ partial) with what access rights (read/ write/ update). The data requestors can be a doctor/ nurse/ insurance providers/ researchers and so on. The requestors should not be given access based on groups or roles but based on identity that is accepted by the data owner.

This is achieved by defining an efficient access control scheme with encryption using basics of ABE. Concerning ABE and IBE, this research addresses the problems of increasing key size with an increasing number of attributes, lack of trust, dependency on key generation centers for keys, attribute management, and patient record ownership as in Fig. 5 and the probable solutions as given in Fig. 6.

The following problems are the motivation to do this research:

- Lack of Trust while sharing data: In ABE, the access controls are defined upon user groups or categories rather than individual users. But, in terms of trust, the patient may have more trust in a particular doctor rather than a group of doctors. A patient wants to share the health data with personally known or identified doctor(s) but not with some doctor(s). So, in some sectors like healthcare, ABE alone cannot be used.
- Dependency on KGC: For the private keys to decrypt the required resource, the requestor of the data should depend on KGC.
- Key Size: As attribute size increases, key size also increases. This adds to computational overhead.
- Attribute Management: The KGC has to manage the attribute universe and ensure that attributes are defined consistently and accurately. Changes or updates to attributes might require coordination with the KGC.
- Key Distribution Complexity: The KGC's role in generating keys becomes more complex in ABE. It generates master keys and policy-specific keys, and it needs to ensure that these keys are properly distributed to authorized users.



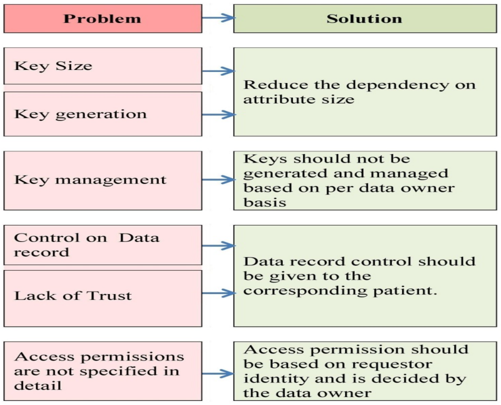
Problem	Description
Key Size	Size of key increases with the increase in attribute size
Key generation	Complexities in generation of keys increatses as the attribute size increases
Key management	Maintainance of keys of all the requestors based on per data owner basis is a tedious process
Control on Data record	The health records are controlled by third party administrators
Lack of Trust	The patient who is the actual owner of a record do not have complete control over the record
Access permissions are not specified in detail	Most of the access permissions are based on role or group

Fig. 5. Summary of problems that motivated to carryout this research.

- Fine-Grained Access Control: While ABE allows for fine-grained access control, this granularity can sometimes lead to overly complex policies that are difficult to manage and understand.
- Trust in Authorities: ABE requires a central authority like KGC to manage attributes and access policies. The trustworthiness of this authority is critical; if compromised, it can lead to unauthorized access.

2) Objectives: The *objective* of this research is to work towards refining the EHR system. This paper focuses on

- To propose an enhanced access control system that achieves fine-grained access control for EHRs.
- To achieve a PHR environment where the patient will have control over his/her own medical record and decide access permissions for users.
- To reduce the dependency on KGC and Master key.
- To minimize the problems of increasing key size due to increasing attribute size.
- To gain the trust of patients to involve themselves in the digitalization of health records.



Problem	Solution
Key Size	Reduce the dependency on attribute size
Key generation	
Key management	Keys should not be generated and managed based on per data owner basis
Control on Data record	Data record control should be given to the corresponding patient.
Lack of Trust	
Access permissions are not specified in detail	Access permission should be based on requestor identity and is decided by the data owner

Fig. 6. Summary of problems and their probable solution.

B. Organization of Paper

The rest of this paper is organized as follows: The review of previous efforts, including research gaps, is covered in Section II. In Section III, the notations used, system and threat model along with the security requirements for the proposed are defined. Section IV details the research methodology of the proposed Access model of PHR with suitable figures, equations, and process flow diagrams. Section V describes the security as well as the comparative analysis and gives comparisons of proposed with existing approaches. Section VI provides the conclusion and future guidelines of the proposed work.

II. Literature Survey

A. ABE in Securing Digital Health Records

In recent years, several studies have focused on developing access control systems based on CPABE for secure and better management of medical information. These proposed solutions aimed to protect patients' privacy and improve the security of EHRs in cloud-based architectures and provide granular access to patients' medical information.

A secure EHR system was presented by Wang and Song [4] in 2018 that used advanced encryption techniques such as ABE, IBE, and identity-based signing for digital signatures. The authors stated the new technique as Combined Attribute Based/Identity Based Encryption and Signature (C-AB/IB-ES). In an effort to strengthen cloud architecture's information outsourcing system, Ramu G. et al. [5] developed an improved CPABE scheme with user deactivation by employing an immediate attribute change technique. Also, to resolve key-escrow issue, a 2-authority collaboration was implemented between cloud server and KGC. The suggested method was proficient in attaining security in outsourced EHRs on cloud.

An investigation by Sudha and Nedunchelian [6] was published in 2019 and demonstrated how CPABE as well as hierarchical attribute-based encryption (HABE) were used in recovering secured info. In their method, actual data was encrypted and provided only the necessary data to others. The sensitive data was kept encrypted. The author also claimed that the owner of the data gained actual data from the processed data of the cloud using an owner-generated key. Wei et al. [7] introduced Revocable Storage and Hierarchical ABE(RS-HABE) in 2019 to handle the security issue that arose while exchanging EHR info securely in public cloud using CPABE. With RS-HABE, every single stakeholder was instructed to generate private keys for their offspring, ensuring both for/back-ward secrecy of encoded EHR. In 2020, Liu et al. [8] devised a hidden EHR distribution technique centered on decentralized HABE to secure privacy of the patient while enhancing data distribution.

Routray et al. [9], in 2020 produced an enhanced CPABE that supported the outsourced decryption and obfuscation of access rules. Also, the computational efficiency was improved using the matrix-based LSS and prime-order bilinear group. In their proposed approach, Ghosh et al. [10] in 2020 suggested two keys for each user to ease the frequent updates of attributes in outsourced CPABE. Among the two keys, one was static and the other was dynamic. Whenever there was a change in user's attributes, only the dynamic key was changed.

To lessen computational complexity of encryption procedure while enhancing security level of system, Lin and Jiang [11] in 2021, suggested a multi-user CPABE method with keyword search. This approach resulted in reduced communication costs and smaller ciphertext lengths. In 2021, a Secure Healthcare Framework (SecHS) was proposed to secure healthcare data by Satar et al.[12]. This scheme was employed with an improved CPABE comprising two additional functionalities to simplify the encryption and hashing techniques. Joshi et al. [13] in 2021 introduced a new unified AB-authorization scheme using CPABE to enable permitted safe access to patient information and streamline the privilege management to a granular range. In this practice, the service control was shifted to medical professionals instead of patients.

Many researchers worked towards the security of health data while supporting EHR system. But, in EHR, the health data is not completely under the control of patient. The patient cannot decide the users of his/her own medical record. To achieve this, some researchers have paved way for PHR environment. In this environment, patient of the record is owner and decides who and all can access the medical record.

Tembhare et al. [14] introduced a system called MediTrust that combined RBAC with ABE systems and utilized a contextualized repository to enhance efficiency of PHR domain. Lin et al. [15] suggested a coordinated CPAB-PHR access control with user accountability (CCP-ABAC-UA). This scheme provided synchronous generation and distributed storage of private keys, which effectively prevented the exposure and escrow of private keys. It also accurately detected key abuse and identified the traitor during decryption. CCP-ABAC-UA was a user-side lightweight scheme that does not require bilinear pairing computations, making it suitable for a secure mobile PHR application with minimal computational overhead. This paper presented a novel provably secure construction of CCP-ABAC-UA, which was secure against selectively chosen-plaintext attacks.

Tao et al. [16] familiarized a unique GO-CPABE-CCS scheme in 2019 for group-oriented CPABE in which users were divided along groups with like-identification, allowing several users to combine their attributes to finish decryption. In 2019, Li et al. [17] offered a scheme based on a threshold policy update. Likewise, Belguith et al. [18] in 2020 utilized signcryption-based CPABE with policy updates and outsourced computations in their work. Both worked on CPABE policy updates but both of these schemes had high computational costs.

In 2020, Guo Rui et al. [19] presented a CPABE method with ability to secure hierarchical health records in a multi-authority PHR environment. The encryption of the hierarchical files was carried out based on an integrated access structure allied with ciphertext. This enabled authorized users with a single private key to decrypt all of the encrypted files. An access control scheme for smart medical systems was offered by Rana S et al. [20] in 2020. This scheme was about the suggested policy-hiding mechanism that encrypts and hides access policies. Zhang et al. [21] in 2021, recommended a notion of PHR distribution that aligned with patients' preferences to secure PHRs before outsourcing using MA-ABE.

In 2021, Liu et al. [22] advised a privacy protection and dynamic share system (PPADS) for PHRs based on CPABE. This approach offered full policy concealment with manageable access control, hiding entire attributes using attribute bloom filters and updating ciphertext using transforming keys. In 2021, Edemacu et al. [23] introduced CPABE featuring lucidity, performance, duplicity prevention, and instant withdrawal of attribute/user. This solution used Ordered Binary Decision Diagrams(OBDD) access structure for expressiveness and outsourced attribute operations to cloud eliminating false attributes. In the same year, Saravanan et al. [24] submitted a well-organized model contingent on HAP-centric CPABE to secure private data. This method includes authentication, secure upload and download stages. This method outperformed traditional security techniques in algorithm complexity, memory utilization, en/de-cryption time, and up/down-load time.

Khan et al. [25] proposed a granular data access control model for healthcare that was patient-centric and was updated by patient or by their designated representatives, in 2021. The proposed model used ABE to provide granular access to patient records stored in a cloud-based system. The model also incorporated a policy update mechanism that allowed patients to modify access permissions for their data. The authors suggested that the proposed model improved patient privacy and control over their data. It also shared health information among authorized parties efficiently. However, the data owner needed to maintain logs of all secret values for policy updates, which was a tedious process for imminent purposes . In the same year, Zhang et al. [26] suggested a PHR system where a recreation of decoding key was not required. This helped in communicating the data to many users.

B. IBE in Securing Digital Health Records

Recently in 2022, a role-based proxy decryption approach was given by Mittal et al. [27] to delegate decryption rights to users and ensure secure retrieval of intimate patient information from EHRs. The approach used IBE to generate public keys based on user information such as phone number and email ID. The encrypted data was sent to cloud and decrypted using a role-specific model upon request from a nurse, lab technician, or physician. The physician group received a public key as per user's request, and proxy decryption was used to extract the data securely. The proposed approach reduced the time of decryption, making it more efficient while ensuring data security.

In 2023, Yu et al. [28] suggested an efficient IBE with a hierarchical model for limited computing devices. The authors proved that their model is efficient. They considered the identity of user for key generation. The problem in this model is that they require a master key and KGC to generate required keys for specified identities.

One of the main drawbacks of IBE is that it requires a KGC that can generate keys for end users with its own master key. This creates a possible privacy concern, as the KGC has the ability to decrypt all encrypted data. As a result, IBE has not been widely adopted.

To address this issue, various suggestions were put forward that aimed to lessen trust in KGC. These proposals often comprised threshold mechanisms or separation-of-duty architectures. However, these solutions can be problematic as they frequently depend on non-collusion conventions that in practical circumstances are not ensured. One such strategy was put forward by Adams [29] in 2022, which used separation architecture to instantiate multiple intermediate CAs (ICAs), as opposed to only one. However, computation cost for user and communication cost with the ICAs were increased in the process of gaining the key.

In most of the research papers, the accessing of data is granted subject to the role of user or group to which user belongs. Also, the patient, who is considered to be the true owner of a particular medical record, is not given total control of that record so far. In a true PHR environment, the patients may be willing to give access to specific doctors whom they know well but not to a group. In this proposed work, the effort is to attain the right PHR environment.

III. System and Threat Model

A. Notations

The notations used in this paper are represented in Table I.

TABLE I. Some Related Notations

Notation	Description
U_a	Authenticated user
uid	User id
pid	Patient id
huid	Hashed user id
hpid, hpid’	Hashed patient id
acode	Access code
sk, key	Secure key
S	PHR system
prkey	Secret key
sac	Single-use authentication code
A	Requested resources code
AAL	Accessible Attributes List
A_{ext}, $Attr_{ext}$	Extracted attributes list
$Attr_{enc}$, A_liste	Encrypted attributes
$Attr_{dext}$, $Attr_{d}$, $Attr_{dp}$, A_listd	Decrypted attributes
extract(...)	Attributes extraction function
member(..., ...)	User membership checking function
access(...)	Attributes accessing function
Enc(..., ...)	Encryption function
Dec(..., ...)	Decryption function
sha3(...)	SHA3 function

B. System Model

The entities in proposed access model are the data owner, doctors/users, Authorization system and PHR server.

- Data Owner: The data owner will have entire control of own health record. The owner decides who will access his/her health record and to what extent. The deciding factor is majorly the in-person trust. This mean that the patient will give permission to those he/she knows in-person or to those recommended by the persons he/she trusts more. In access policy, for each user, identity and attributes code with access permission is included. The complete list is maintained as Accessible Attribute List(AAL) in encrypted format.
- Doctor or Requestor: Whenever a user, like a doctor or a nurse, requests the details of a particular patient, then user details along with requested resource details are sent to the server in an encrypted format. After successful authorization, the requestor receives the encrypted data. Requestor has to enter correct access code to decrypt the data of requested patient.
- Authorization System and PHR: The detail sent by the requestor is decrypted. Then, the membership of the requestor who requested the resources is verified against the corresponding patient for the user. The requested resources for which the memberships are found are encrypted and sent to the user. The resources are selected according to corresponding user’s permission code in AAL. User has to enter correct access code to decrypt the data of the requested patient.

C. Security Model

The proposed model includes the following algorithms:

- setup()→acode,sk: The access code (acode) and the secret code (sk) are precomputed.
- encrypt(A) using (acode,sk)→A’: The requested attribute-list A is encrypted using the acode and sk. The encrypted result is A’.
- member(huid,hpid)→bool: Returns the Boolean value based on the membership status of the user with the corresponding patient.
- extract(Attr)→Attr’: Extracts the list of attributes(Attr) based on constraints. Here Attr’∈Attr. Attr’ represents requested Attr or subset(Attr) or empty.
- decrypt(A’) using (sk,acode)→A: The A’ is decrypted using sk and acode to retrieve the plain data A.

D. Security Requirements

As our goal is to design a patient-oriented health record system, there requires some security issues to be concerned as follows:

- Data privilege: The accessing of patient health records should be restricted based on policies defined by corresponding health record owner (i.e., patient). The dependency on KGC should also be minimized.
- Key theft: In ABE, KGC plays a major role in generation and distribution of master keys to the users. But, KGC is a third-party. The chance of leakage of keys through KGC is a big concern.
- Collision tolerance: In ABE, collision attacks should also be avoided. Different users may join each other and combine their attributes to acquire the ability to decrypt the required encrypted text. This is called the collision.

IV. Proposed Access Model of PHR

In this proposed work, the patient will be the owner of their complete medical record. This means that the patient will have total control of his/her own medical record. The record in the PHR system will be in encrypted format using a secret key only known to server. Whenever a user, like a doctor or a nurse, requests the details of a particular patient, then the user details along with the requested resource details are sent to the server in an encrypted format.

At the server end, the requested resource indexes are decrypted and verified at the indexed location for the read/write value. The encryption / decryption of data that is moved between the server and the user is done by using a key. This particular key is generated by the user and the server separately on their nodes using the unique credentials of the user. These credentials were shared with the server by the client using a quantum-resilient algorithm like Kyber [30].

The memberships of requestors those requested the resources are verified against corresponding patient for the user. The requested resources for which the memberships found are encrypted and sent to the user. User has to enter correct key and access code to decrypt the data of the requested patient. In the proposed method, there is no requirement for the KGC. So, master key generator and private-public key generators for all users are not necessary explicitly.

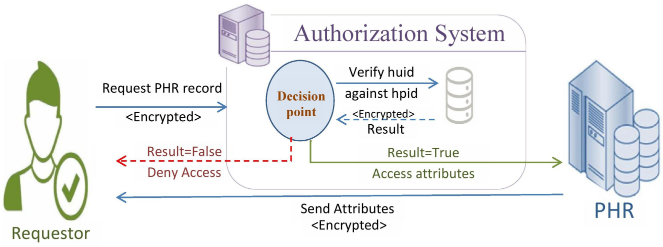


Fig. 7. Proposed access model of PHR.

Fig. 7 shows the proposed model of accessing a patient's health record by a requestor from PHR server. All the data that is transferred from one node to another always is in encrypted format. Only the intended one can decrypt the data. In this model of PHR system, the true owner of a record is its patient. This means that the total control of a record is with the corresponding patient.

A. Proposed Access Control Model

The proposed model used the enhanced Attribute-Based Access Control model that was used by the ABE along with the basic idea of IBE, i.e., user-identity.

Enhanced access control model consists of the following:

- Policy Enforcement Point (PEP): This is to secure applications and data by analyzing requests and disseminating authorization needs to the Policy Decision Point (PDP).
- Policy Matching Point (PMP): This bonds external resource of attributes related to a particular patient's record only when compatible with the requestor.

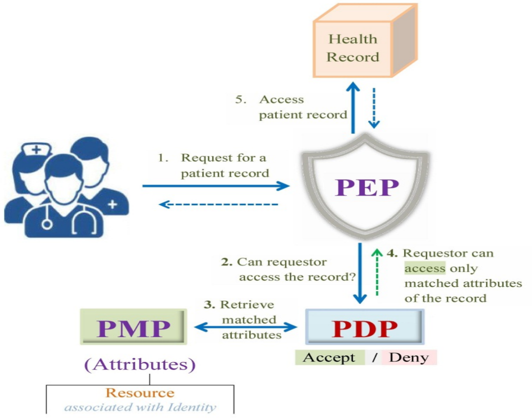

Health Record
5. Access patient record
1. Request for a patient record
PEP
2. Can requestor access the record?
4. Requestor can access only matched attributes of the record
3. Retrieve matched attributes
PMP
(Attributes)
Resource
associated with Identity
PDP
Accept / Deny

Fig. 8. Enhanced access control model.

In Fig. 8, the PEP and PDP are same as that of in generic Attribute Based Access Control (ABAC) model. But, instead of Policy Information Point (PIP) and Policy Administration Point (PAP), the PMP was introduced. In PMP, the requested resource attributes that were associated with the requestor id will only be retrieved. But, the requestor id should have been associated with the requested patient-data. Then only the Access is given. Otherwise the decision – Deny will be taken by PDP and forwarded it to PEP.

B. Encryption / Decryption

The requested attributes that are to be transferred are encrypted or decrypted based on the identity of requestor. Here, the requestor should be adhering to the access control rights and its decision. Before the attributes request, a requestor have to log-in to the system successfully. Whenever a user logs in to system, the login details in it's hashed form are encrypted using Kyber [30] and sent to the server.

1) Setup and Extract: Given that U_a has successfully logged into S. S will send an *acode* to U_a. *sk* is used to secure the data to be transferred between U_a and S. The key (sk) used to encrypt or decrypt is an AES-GCM [31] based symmetric key. *sk* is generated using a *sac* at both client side by U_a and at S for further transactions. The generation of *sk* was detailed in our previous research [32]. Also, S maintains its own *prkey*.

2) Encrypt:

- *Pre-encrypt*
  AAL is a code to represent the list of attributes that are given access permission to U_a by the data owner.

$$uid = ID(U_a) \tag{1}$$

$$huid = sha3(uid) \tag{2}$$

$$hpid = sha3(pid) \tag{3}$$

$$member(huid, hpid) = \begin{cases} True, & \text{if, for a hpid } \exists \text{ huid} \\ False, & \text{otherwise} \end{cases} \tag{4}$$

If (4) results in *True* then,

$$\begin{aligned} A_{ext} &= extract(A) \\ &= \begin{cases} A, & \text{if } \forall \text{A} \in \text{AAL} \\ A \cap AAL, & \text{otherwise} \end{cases} \end{aligned} \tag{5}$$

$$\begin{aligned} Attr_{ext} &= extract(A_{ext}) \\ &= \begin{cases} A_{ext} \text{ from PHR}, & \text{if hpid=hpid'} \\ False, & \text{otherwise} \end{cases} \end{aligned} \tag{6}$$

- *Encrypt*

$$Attr_{enc} = Enc(Enc(Attr_{dext}, acode), sk) \tag{7}$$

Where,

$$Attr_{dext} = Dec(Attr_{ext}, prkey)$$

3) Decrypt and Access:

$$Attr_{dp} = Dec(Attr_{enc}, sk) \tag{8}$$

$$\begin{aligned} Attr_d &= access(Attr_{dp}) \\ &= \begin{cases} Dec(Attr_{dp}, acode_{U_a}), & \text{if K=True} \\ DENY, & \text{otherwise} \end{cases} \end{aligned} \tag{9}$$

where,

$$K = \begin{cases} True, & if \text{U=}U_a \text{ and } acode_{U_a}\text{=}acode_s \\ False, & \text{otherwise} \end{cases}$$

Fig. 9 shows the flow of an user request to access a patient record.

- Whenever a user is logged into the system and is authenticated properly, the PHR system will send an *acode* that is valid for the entire session. *acode* is also used as user's identity confirmation. The user can access requested resource only if authenticated properly and *acode* for that session is validated correctly.
- U_a requests the resource from PHR system. The *huid* (2), *hpid* (3), and *A* are encrypted and sent the request for authorization checking.
- The system decrypts the request and checks the membership of *huid* associated with *hpid*. This is given at (4). If found, the *A* is compared with the *AAL*. This results in the exactly matched attributes list along with their read/write access permission. This is represented at (5).
    - AAL is combination of a health record's column-index and their corresponding read/write permission code.
- The resulting attributes ($Attr_{ext}$) are retrieved at (6) and then decrypted by the server using *prkey* to get $Attr_{dext}$ at (7). Now, $Attr_{dext}$ is encrypted using *sk* and the *acode* as in (7). Then, it is sent to the user. Each legitimate user and the PHR system have agreed on a key(sk). This is unique from other users. This *sk* is used for securing the data that is to be transmitted between U_a and server.
- The authenticated user has to use the correct *sk*, *acode* to decrypt the resource completely. This is represented in (8) and (9).

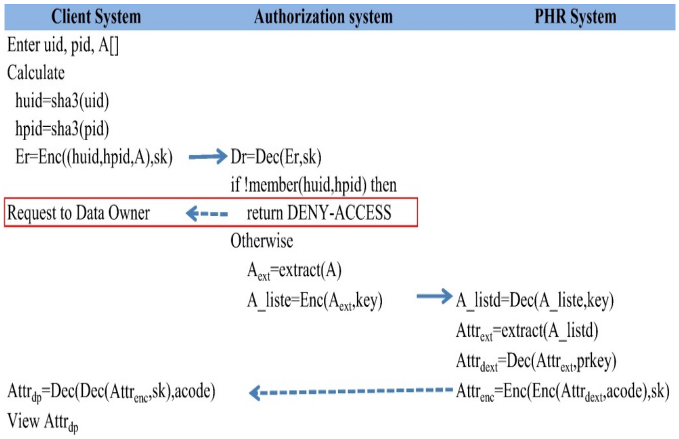


Fig. 9. Process flow of an user accessing a patient record.

The U_a can view or update the patient data based on access permission associated with the attribute. If uploader is other than the owner then data is added to the record but needs to be approved by the owner. If user updates any data then the upload details like the id of the data uploader, time-date of upload, owner id, owner approval time-date, and hashed value of the upload detail and approval detail are logged into the upload-logs.

When a new patient record is created, the corresponding hashed value is logged as it is. After that, the hash of current hashed value along with previous hashed value is logged into the log-field. This is to maintain the non-repudiation of data uploaded.

V. Results and Discussion

A. Security Analysis

1) Key generation center: In proposed method, there is no need for KGC. So, problems like key theft through KGC, and dependency for key generation on KGC will not arise.

2) Data privilege: The data owner is given complete control over his/her own record and defined the strict access policy. The access permission was given to only the in-person trusted users. The dependency on KGC is also minimized. Instead, the keys were generated by the client and server based on a unique key generation algorithm [32].

3) Collision tolerance: The policies were defined based on the unique identity of the user. The keys generated and used were also independent of each user. The key generated includes a portion of user's unique password. So, chance of collision was also minimized.

4) Multiple key maintenance: When a medical record is considered, there may be many users for a particular record with different accessing levels and accessing rights on attributes. To handle this, in existing schemes, multiple keys were generated based on the requirement for data to be accessed. In proposed system, there is no need to maintain multiple keys as maintained in existing systems.

TABLE II. Comparison of Proposed Model with the Related Models

References	Key Generator/ Attribute Authority	Intermediate Certification Authority (ICA)	Attribute size	Encryption/ Decryption type	Master key	Privacy breach
[29]	Yes	Yes -Multiple	1	Identity based	Required	No
[7]	Yes	Single	1	Identity based	Required	Yes
[27]	Yes	No	Multiple	Role/ Group based	Required	Yes
[33]	Yes	No	Multiple	Role/ Group based	Required	Yes
[28]	Yes	No	1	Identity based	Required	Yes
[25]	Yes	-	Multiple	CPABE	Required	Yes
[26]	Yes	No	1	Role based	Required	Yes
[34]	Yes	No	Multiple	Attribute based	Required	No
Proposed	No	No	1	User-identity based	Not Required	No

5) Key delegation time: As the existing models involve the KGC for generating the required key for a resource requestor to access the required data, surely there will be some key generation time. In the proposed system, there is no need for the KGC. So, there will not be any generation of keys explicitly. This implies no key delegation time.

6) Non-repudiation: As every transaction with the PHR is logged and the log-details are hashed properly as a chain of hash; any user cannot deny the action made with the PHR system.

B. Comparative Analysis

This was analyzed based on some parameters like the dependency on key generation/ attribute authority, involvement of intermediate certification authority, attribute size, encryption/ decryption type, dependency on master key to generate requestor's keys, and whether prone to privacy breach or not. Table II states comparison of proposed model with related ones. In this comparison, proposed model stands better than its counterparts.

- *Attribute Authority*
  The Attribute authority or KGC is required to manage the attributes associated with the encryption/ decryption process. Based on these attributes, keys are generated to control the access of sensitive data. The schemes at [7], [25], [26], [27], [28], [29], [33] and [34] involved KGC. But, this KGC is not required in proposed scheme.
- *Encryption /Decryption Type*
  The [26], [27], and [33] used Role/Group based encryption, [25] used CPABE for encryption/ decryption of data. Whereas, [7], [28], and [29] used Identity based encryption. The proposed scheme used user-identity as attribute for encryption.
- *Master Key*
  All the related schemes required the master key to get public/private keys that are to be distributed to respective participants. Whereas, it is not required in proposed.

C. Simulation Setup

The proposed model is implemented using java and libraries like Java pairing-based cryptography (jpbc) on a computer with specifications of Intel Core i5 with 4GB RAM, 2.30GHz processor on Windows 10 32-bit OS. The time comparison of the various basic steps involved is given in Table III.

TABLE III. Time Comparison of Steps Involved in Various Schemes

Scheme / Step	Setup (in ms)	Extract (in ms)	Encrypt (in ms)	Decrypt (in ms)
BF-IBE[2]	14.01	27.48	33.18	18.22
IBDD[26]	14.01	28.32	34.3	88.79
HEIE[28]	59.63	71.58	47.31	40.23
Proposed	16.52	10.42	31.1	16.38

Table III shows the time taken by four different schemes for four steps in a process. The schemes are BF-IBE [2], IBDD[26], HEIE [28], and Proposed. Fig. 10 represents the graph for the processing time comparisons of the basic methods involved in existing schemes with the proposed one. The steps are Setup, Extract, Encrypt, and Decrypt.

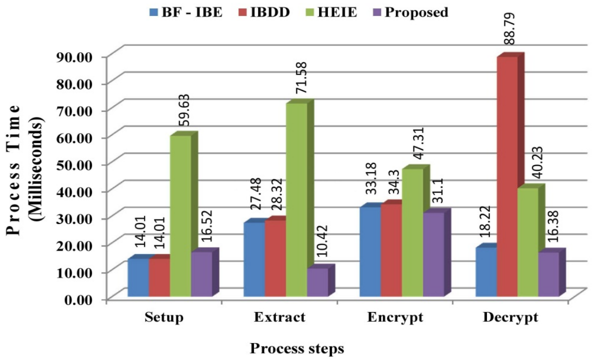


Fig. 10. Processing time for different steps involved in different schemes.

- *Setup*

The [2] and [26] schemes are the fastest at setup, taking 14.01ms. This is followed by proposed, which take 16.52ms. [28] is the slowest at Setup, taking 59.63ms.

- *Extract*
  The Proposed scheme is the fastest at Extract, taking 10.42ms. This is followed by [2], which takes 27.48ms. [28] is the slowest at Extract, taking 71.58ms.
- *Encrypt*
  The Proposed scheme is the fastest at Encrypt, taking 31.1ms. This is followed by [2], which takes 33.18ms. [28] is the slowest at Encrypt, taking 47.31ms.
- *Decrypt*
  The Proposed scheme is also the fastest at Decrypt, taking 16.38ms. This is followed by [2], which takes 18.22ms. [26] is the slowest at Decrypt, taking 88.79ms.

The total processing time of all the schemes are also calculated and analyzed in terms of percentage at Table IV. The percentage decrease in processing time of proposed when compared with related schemes is calculated using (10).

$$PercentageDecrease(PD) = \frac{EPT - PPT}{EPT} \times 100\% \quad (10)$$

Where,
EPT = Existing scheme's Processing time
PPT = Proposed scheme's Processing Time

TABLE IV. Decrease Percentage in Total Processing Time of Proposed

Scheme	Total processing time (in ms)	% decrease (in ms)
[2]	92.89	19.89
[26]	165.42	55.04
[28]	218.75	65.94
Proposed	74.42	0

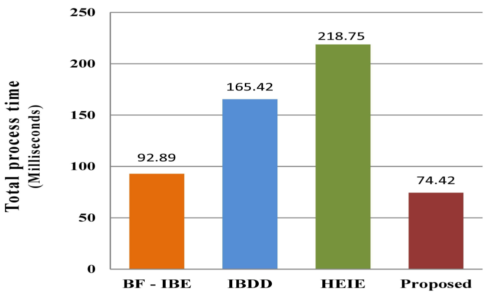


Fig. 11. Total Processing time for different schemes.

Fig. 11 shows the overall processing time comparison graph of the proposed scheme with other related schemes. Overall, the proposed scheme is faster than [2], [26], and [28]. The proposed scheme is approximately 19.89% faster than [2], 55.04% faster than [26], and 65.94% faster than [28] as given in Fig. 12.

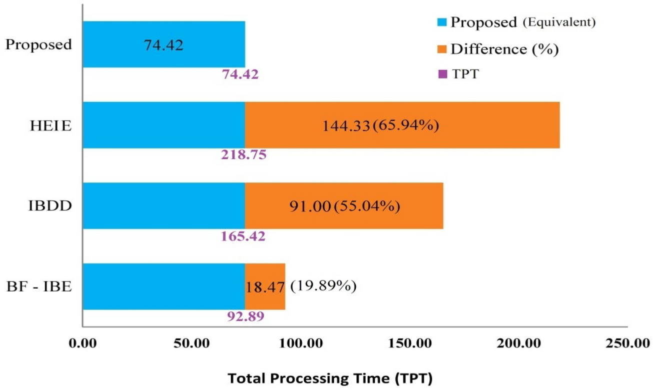


Fig. 12. Total processing time of different schemes comparing to % decrease in processing time of proposed.

VI. Conclusion

Personal health records are to be managed by patients themselves. As the health data is very sensitive, patients will not be willing to share their health data online. To gain their trust and involve them in digital health, each patient should be given complete control on his/ her health record. The proposed scheme used the enhanced access control model that has given patients the immunity to decide who can be the accessor of his/her medical record and with what access permissions (read/write). The total processing time of proposed is 74.42ms. But, the same is 92.89ms, 165.42ms, and 218.75ms in case of Boneh-Franklin, Zhang et al., and Yu et al., respectively. Also, the threats that arise because of KGC are not there in the proposed method as it does not have the role of KGC and master keys along with their counterpart keys. The AES method is used to secure the data at rest by the server. In future work, current work has to be extended with an emergency phase, where the patient's record should be accessed easily in an emergency situation. Also, the traditional encryption algorithm is to be completely replaced with the quantum resistant encryption algorithm.

References

[1] A. Shamir, "Identity-Based Cryptosystems and Signature Schemes," in *Proceedings of CRYPTO 84 on Advances in Cryptology*, 1985, pp. 47-53.

[2] D. Boneh and M. Franklin, "Identity Based Encryption from the Weil Pairing," in *Advances in Cryptology-CRYPTO 2001: 21st Annual International Cryptology Conference Proceedings*, 2001, pp. 213–229. [Online]. Available: https://eprint.iacr.org/2001/090

[3] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in Proceedings - *IEEE Symposium on Security and Privacy*, 2007, pp. 321–334. doi: 10.1109/SP.2007.11.

[4] H. Wang and Y. Song, "Secure Cloud-Based EHR System Using Attribute-Based Cryptosystem and Blockchain," *J. Med. Syst.*, vol. 42, no. 8, p. 152, Aug. 2018, doi: 10.1007/s10916-018-0994-6.

[5] G. Ramu, B. E. Reddy, A. Jayanthi, and L. V. N. Prasad, "Fine-grained access control of EHRs in cloud using CP-ABE with user revocation," *Health Technol. (Berl).*, vol. 9, no. 4, pp. 487-496, Aug. 2019, doi: 10.1007/S12553-019-00304-9/METRICS.

[6] I. Sudha and R. Nedunchelian, “Protected health care application in cloud using ciphertext-policy attribute-based encryption and hierarchical attribute-based encryption,” *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 11, pp. 3245-3241, Sep. 2019, doi: 10.35940/IJITEE.K2529.0981119.

[7] J. Wei, X. Chen, X. Huang, X. Hu, and W. Susilo, “RS-HABE: Revocable-storage and Hierarchical Attribute-based Access Scheme for Secure Sharing of e-Health Records in Public Cloud,” *IEEE Trans. Dependable Secur. Comput.*, vol. 18, no. 5, pp. 1-1, 2019, doi: 10.1109/TDSC.2019.2947920.

[8] X. Liu, X. Yang, Y. Luo, L. Wang, and Q. Zhang, “Anonymous Electronic Health Record Sharing Scheme Based on Decentralized Hierarchical Attribute-Based Encryption in Cloud Environment,” *IEEE Access*, vol. 8, pp. 200180–200193, 2020, doi: 10.1109/ACCESS.2020.3035468.

[9] K. Routray, K. Sethi, B. Mishra, P. Bera, and D. Jena, “CP-ABE with Hidden Access Policy and Outsourced Decryption for Cloud-Based EHR Applications,” in *Smart Innovation, Systems and Technologies*, vol. 196, Springer, Singapore, 2021, pp. 291-301. doi: 10.1007/978-981-15-7062-9_29.

[10] B. Ghosh, P. Parimi, and R. R. Rout, “Improved Attribute-Based Encryption Scheme in Fog Computing Environment for Healthcare Systems,” in *2020 11th International Conference on Computing, Communication and Networking Technologies*, ICCCNT 2020, Jul. 2020, pp. 1–6. doi: 10.1109/ICCCNT49239.2020.9225606.

[11] H. Y. Lin and Y. R. Jiang, “A Multi-User Ciphertext Policy Attribute-Based Encryption Scheme with Keyword Search for Medical Cloud System,” *Appl. Sci.*, vol. 11, no. 1, p. 63, Dec. 2021, doi: 10.3390/APP11010063.

[12] S. D. M. Satar, M. A. Mohamed, M. Hussin, Z. M. Hanapi, and S. D. M. Satar, “Cloud-based Secure Healthcare Framework by using Enhanced Ciphertext Policy Attribute-Based Encryption Scheme,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 393–399, 2021, doi: 10.14569/IJACSA.2021.0120643.

[13] M. Joshi, K. P. Joshi, and T. Finin, “Delegated Authorization Framework for EHR Services Using Attribute-Based Encryption,” *IEEE Trans. Serv. Comput.*, vol. 14, no. 6, pp. 1612–1623, 2021, doi: 10.1109/TSC.2019.2917438.

[14] A. Tembhare, S. Sibi Chakkaravarthy, D. Sangeetha, V. Vaidehi, and M. Venkata Rathnam, “Role-based policy to maintain privacy of patient health records in cloud,” *J. Supercomput.*, vol. 75, no. 9, pp. 5866–5881, Sep. 2019, doi: 10.1007/S11227-019-02887-6/METRICS.

[15] G. Lin, L. You, B. Hu, H. Hong, and Z. Sun, “A coordinated ciphertext policy attribute-based PHR access control with user accountability,” *KSII Trans. Internet Inf. Syst.*, vol. 12, no. 4, pp. 1832–1853, Apr. 2018, doi: 10.3837/TIIS.2018.04.024.

[16] X. Tao, C. Lin, Q. Zhou, Y. Wang, K. Liang, and Y. Li, “Secure and efficient access of personal health record: a group-oriented ciphertext-policy attribute-based encryption,” *S.ITransactions Chinese Inst. Eng. J. Chinese Inst. Eng.*, vol. 42, no. 1, pp. 80–86, Jan. 2019, doi: 10.1080/02533839.2018.1537810.

[17] J. Li et al., “An Efficient Attribute-Based Encryption Scheme with Policy Update and File Update in Cloud Computing,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 12, pp. 6500–6509, Dec. 2019, doi: 10.1109/TII.2019.2931156.

[18] S. Belguith, N. Kaaniche, M. Hammoudeh, and T. Dargahi, “PROUD: Verifiable Privacy-preserving Outsourced Attribute Based SignCryption supporting access policy Update for cloud assisted IoT applications,” *Futur. Gener. Comput. Syst.*, vol. 111, pp. 899–918, 2020, doi: https://doi.org/10.1016/j.future.2019.11.012.

[19] R. Guo, X. Li, D. Zheng, and Y. Zhang, “An attribute-based encryption scheme with multiple authorities on hierarchical personal health record in cloud,” *J. Supercomput.*, vol. 76, no. 7, pp. 4884–4903, Jul. 2020, doi: 10.1007/S11227-018-2644-7/METRICS.

[20] S. Rana and D. Mishra, “Efficient and Secure Attribute Based Access Control Architecture for Smart Healthcare,” *J. Med. Syst.*, vol. 44, no. 5, pp. 1–11, May 2020, doi: 10.1007/S10916-020-01564-Z/METRICS.

[21] L. Zhang, Y. Ye, and Y. Mu, “Multiauthority Access Control With Anonymous Authentication for Personal Health Record,” *IEEE Internet Things J.*, vol. 8, no. 1, pp. 156–167, 2021, doi: 10.1109/JIOT.2020.3000775.

[22] Z. Liu, J. Ji, F. Yin, and B. Wang, “Sharing and privacy in PHRs: Efficient policy hiding and update attribute-based encryption,” *KSII Trans. Internet Inf. Syst.*, vol. 15, no. 1, pp. 323–342, Jan. 2021, doi: 10.3837/TIIS.2021.01.018.

[23] K. Edemacu, B. Jang, and J. W. Kim, “CESCR: CP-ABE for efficient and secure sharing of data in collaborative ehealth with revocation and no dummy attribute,” *PLoS One*, vol. 16, no. 5, pp. 1–24, May 2021, doi: 10.1371/journal.pone.0250992.

[24] S. N and D. U. A, “Hap-Cp-Abe Based Encryption Technique With Hashed Access Policy Based Authentication Scheme For Privacy Preserving Of Phr,” *Microprocess. Microsyst.*, vol. 80, p. 103540, Feb. 2021, doi: 10.1016/J.MICPRO.2020.103540.

[25] F. Khan, S. Khan, S. Tahir, J. Ahmad, H. Tahir, and S. A. Shah, “Granular Data Access Control with a Patient-Centric Policy Update for Healthcare,” *Sensors*, vol. 21, no. 10, p. 3556, May 2021, doi: 10.3390/S21103556.

[26] Y. Zhang, D. He, M. S. Obaidat, P. Vijayakumar, and K. F. Hsiao, “Efficient Identity-Based Distributed Decryption Scheme for Electronic Personal Health Record Sharing System,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 2, pp. 384–395, Feb. 2021, doi: 10.1109/JSAC.2020.3020656.

[27] S. Mittal et al., “Using Identity-Based Cryptography as a Foundation for an Effective and Secure Cloud Model for E-Health,” *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/7016554.

[28] Q. Yu, J. Shen, J. Li, and S. Ji, “Hierarchical and Efficient Identity-based Encryption Against Side Channel Attacks,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 6, pp. 629–640, 2023, doi: 10.14569/IJACSA.2023.0140667.

[29] C. Adams, “Improving User Privacy in Identity-Based Encryption Environments,” *Cryptography*, vol. 6, no. 4, 2022, doi: 10.3390/cryptography6040055.

[30] J. Bos et al., “CRYSTALS - Kyber: A CCA-Secure Module-Lattice-Based KEM,” in *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, Apr. 2018, pp. 353–367. doi: 10.1109/EuroSP.2018.00032.

[31] M. Dworkin, “Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC.” Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD, 2007. [Online]. Available: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=51288

[32] B. Mohinder Singh and J. Natarajan, “A novel secure authentication protocol for eHealth records in cloud with a new key generation method and minimized key exchange,” J. King Saud Univ. - Comput. Inf. Sci., vol. 35, no. 7, p. 101629, 2023, doi: 10.1016/j.jksuci.2023.101629.

[33] H. Deng et al., “Identity-Based Encryption Transformation for Flexible Sharing of Encrypted Data in Public Cloud,” *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 3168–3180, 2020, doi: 10.1109/TIFS.2020.2985532.

[34] H. Hong, B. Hu, and Z. Sun, “An Efficient and Secure Attribute-Based Online/Offline Signature Scheme for Mobile Crowdsensing,” Human-centric Comput. Inf. Sci., vol. 11, 2021, doi: 10.22967/HCIS.2021.11.026.