IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 23, 2019, accepted June 12, 2019, date of publication June 27, 2019, date of current version July 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2925390

Privacy Provision in Collaborative Ehealth With Attribute-Based Encryption: Survey, Challenges and Future Directions

KENNEDY EDEMACU, HUNG KOOK PARK, BEAKCHEOL JANG, AND JONG WOOK KIM
Department of Computer Science, Sangmyung University, Seoul 03016, South Korea

Corresponding author: Jong Wook Kim (jkim@smu.ac.kr)

This work was supported by the Institute for Information and Communications Technology Promotion (IITP) grant funded by the Korean Government (MSIP) (Development of Integraphy Content Generation Technique for N-Dimensional Barcode Application) under Grant 2017-0-00515.

ABSTRACT Collaborative eHealth that enables the collection and sharing of patients' data is eliminating the location and accessibility barriers in healthcare service delivery. With advances in technology, patients' data can be collected from anywhere, at any time, stored in a central location, and shared across multiple service providers to improve the quality of healthcare service delivery. Maintaining patients' data privacy during the storage and sharing process is critical for collaborative eHealth success. The attribute-based encryption, which has shown potential in data privacy protection in the cloud, has the necessary ingredients to be used in guarding against privacy violations in collaborative eHealth. In this paper, we survey the different attribute-based encryption schemes for usage in collaborative eHealth. We discuss some challenges associated with the use of attribute-based encryption schemes in collaborative eHealth and point out some future research directions. Also, we perform a comparative analysis of the surveyed schemes in terms of security, revocation ability, and efficiency.

INDEX TERMS Attribute-based encryption (ABE), collaborative ehealth, electronic health record (EHR), privacy.

I. INTRODUCTION

Technological developments in the last few decades have led to significant improvements in healthcare digitization. The power of cloud computing, the diminishing cost and size of health sensors, the improvement in speed and capacity of wireless technologies, and the ubiquity of mobile-handheld devices have all contributed to this cause. Health care services for the elderly and other patients have tremendously improved. Reportedly, improved healthcare service delivery has resulted in an increased average life expectancy because of remote patient monitoring technologies [1]. This digitization has paved the way for sharing and use of electronic health records (EHR), thus enabling patients to be collaboratively treated by different healthcare institutions and professionals.

Collaborative eHealth describes aspects of the central storage of patients' EHRs collected from multiple sources for sharing amongst different healthcare parties for improved healthcare service delivery. This enhances clinical and other health-related decision-making processes by availing health information whenever required. Patients can move from one location to another seeking medical treatment without access limitations to their historical health records. Through cloud computing platforms, the need to allow EHRs to be stored centrally is achieved. With patients' consent, different parties can access and use this information.

In a typical collaborative eHealth environment, where there are multiple collaborating parties coming from a diverse range of authorities under different managements [2], the number of security weaknesses and threats violating patients' privacy are constantly high. In this kind of setup, adversaries always try to snoop through stationary or moving EHRs. Several privacy violation issues in collaborative eHealth have already been discussed in [3], [4] (see section III for their summary). Also, allowed parties in collaborative eHealth need to trust the integrity of information displayed

The associate editor coordinating the review of this manuscript and approving it for publication was Claudio Ardagna.

on their monitors. Consequentially, this makes security and privacy provision through incorporating social, legal and technological (which is our focus in this work) aspects [5], a key component of collaborative eHealth.

Several technological techniques are being used to achieve privacy provision in EHR sharing in the cloud, ranging from conventional cryptographic techniques like advanced encryption standard (AES) [6], elliptic curve cryptography (ECC) [7], [8], elliptic curve diffie-hellman key exchange [9] and elliptic curve integrated encryption scheme (ECIES) [10] to more recent techniques like attribute-based encryption (ABE) [11], [12] and homomorphic encryption (HE) [13]. These methods provide data collection, storage, sharing, and computation privacy to EHRs through encryption. The emerging encryption schemes differ from conventional cryptographic schemes by preserving privacy during information sharing to multiple users (a case for ABE) and computations (a case for HE). The homomorphic encryption scheme permits computations on encrypted data, hence reducing the chances of information leakage during computations. Details about homomorphic encryption can be accessed from [14], [15] surveys. On the other hand, ABE is a one-to-many public key cryptographic scheme allowing one-time encryption of the message for multiple recipients.

ABE has already been used for privacy provision in several cloud computing applications [16]–[28] and is similarly being widely accepted in cloud-based eHealth applications [18], [29]–[34]. This is because ABE can provide fine-grained access control, scalability and availability while maintaining data integrity [35] during information exchange.

A. MOTIVATIONS AND CONTRIBUTIONS

The advent of cloud computing has sparked the development of lots of life-changing applications, including health. Patients' health information can now be stored centrally in the cloud for sharing with multiple health stakeholders for collaboration. However, the centralized storage subjects the patient to several privacy violations. A promising approach to curb this is to encrypt the data before out-sourcing it to the cloud. Secondly, access to the encrypted data should be controlled. Attribute-based encryption presents the qualities to encrypt and provide fine-grained access control at the same time. Much as ABE schemes have attracted tremendous attention for privacy provision in collaborative eHealth, comprehensive surveys discussing the different ABE categories are scarce. Therefore, in this study, we survey the different attribute-based encryption schemes for privacy provision in collaborative eHealth, and in the process, make the following contributions.

1) We present an overview of collaborative eHealth and highlight the privacy violation issues that may arise.
2) We discuss mechanisms that can address the highlighted privacy issues and state collaborative eHealth security requirements.
3) We survey and discuss attribute-based encryption schemes for privacy provision in collaborative eHealth.
4) We provide comparisons of the different attribute-based encryption schemes for the different user domains of collaborative eHealth.
5) We point out the challenges of attribute-based encryption.
6) Finally, we present future directions that require further investigation.

B. PAPER ORGANIZATION

We organize the rest of the paper as follows: In section II, we present the related ABE surveys. Section III covers an overview of the collaborative eHealth, privacy violation issues, remedies to the privacy issues and discussions of the basic security requirements. In section IV, we present attribute-based encryption including access structures and the different attribute-based encryption schemes. In section V, we present the challenges of attribute-based encryption. Sections VI and VII present the comparative analysis of the different ABE schemes and the future research directions, respectively. Section VIII concludes the paper.

II. RELATED WORK

Due to the growing popularity of attribute-based encryption for security and privacy provision in cloud environments, a number of surveys tracking its progress have been conducted over the years. To our surprise, few have being as comprehensive as one might think. In this section, we present some related surveys, highlighting their coverages and gaps as summarized in Table 1. Unlike the presented surveys, our work is comprehensive. We have covered the entire attribute-based encryption spectrum ranging from its initiation to now. We have discussed the various access structures used in the attribute-based encryption schemes as well. Additionally, we have presented the likely privacy violation issues in collaborative eHealth and remedial approaches to address these issues. A comparative analysis of different attribute-based encryption schemes is also included in our work.

III. AN OVERVIEW OF COLLABORATIVE EHEALTH AND ITS SECURITY REQUIREMENTS

In this section, we present an overview of collaborative eHealth. We highlight the privacy challenges that may arise in such a setting. We summarize the different mechanisms to address the privacy issues. We give an example use-case scenario, highlighting a security concern that may arise, and show how to combat it with the help of attribute-based access control. We complete the section by highlighting the security requirements of collaborative eHealth.

A. COLLABORATIVE EHEALTH

Figure 1 shows the general architecture of the collaborative eHealth depicting the three major entities involved.

1) **Patient:** The patient is the data (EHR) owner. The patient's data are collected through various means

TABLE 1. Summary of related works.

Study	Covered	Not included
Lui [36]	Reviewed ABE schemes with revocation functionalities (in particular, [37]–[40] schemes), and evaluated them in terms of data confidentiality, fine-grained access control, scalability, collusion resistance, forward and backward secrecy, computation and storage cost, etc.	Discussions of the other ABE categories, security threats and attacks ABE is expected to address, and the different access structures for use during ABE construction are omitted.
Moffat [41]	Surveyed CP-ABE and multi-authority ABE scheme for mobile devices. Also, highlighted on constant-sized ciphertext and secret key ABE schemes.	Discussions of the KP-ABE, revocable, hidden-policy and hierarchical ABE schemes, and security threats and attacks in the cloud environment are omitted. No comparison of the presented schemes.
Pang [42]	Presented the following, CP-ABE, KP-ABE, multi-authority, revocable and accountable ABE schemes, and the proxy-re-encryption based ABE schemes.	Hidden-policy and Hierarchical ABE schemes are omitted. No discussion of the different access structures used in ABE construction. There is no discussion of the security threats or attacks expected to be addressed by ABE schemes.
Al-dahhan [43]	Discussed the CP-ABE, KP-ABE, multi-authority and revocable ABE schemes.	The study omits the discussion of hierarchical, hidden-policy ABE schemes, among others. It also does not present the different access structures used in ABE construction.
Qiao [44]	Presented the required features of attribute-based encryption such as data confidentiality, fine-grained access control, scalability, user accountability, user revocation, and collusion resistance.	The study omits the discussions of the various ABE categories.
Lee [45] and Cheng [46]	Reviewed CP-ABE, KP-ABE, ABE based on non-monotonic threshold tree access structure and hierarchical ABE.	Discussions of revocable, hidden-policy and multi-authority ABE schemes are omitted. Also, security threats or attacks are not presented.

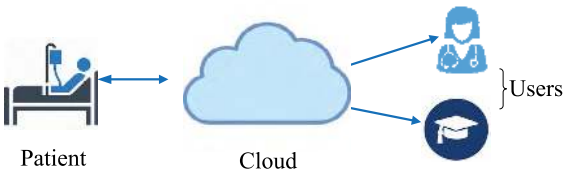

FIGURE 1. A general architecture of collaborative eHealth showing the major entities.

and stored centrally in the cloud. Digitized laboratory examinations, digitized pharmacy prescriptions, digitized physical examinations obtained either through sensors or physical examination by specialists are some example patient data gathered and sent to the cloud. The data is stored in a cumulative manner. An example hierarchical arrangement of the gathered data is shown in Figure 2.

2) **Cloud:** The cloud is a powerful distributed computing facility for storing and processing the patient's data whenever required. Cloud services are provided by cloud service providers.
3) **User:** The user accesses the patient's data from the cloud and uses it to make informed decisions. Examples of users are doctors, nurses, pharmacists, laboratory specialists, researchers, insurance companies, friends, relatives, etc. As we shall see later, the users are categorized into two domains, *private* and *public* domains.

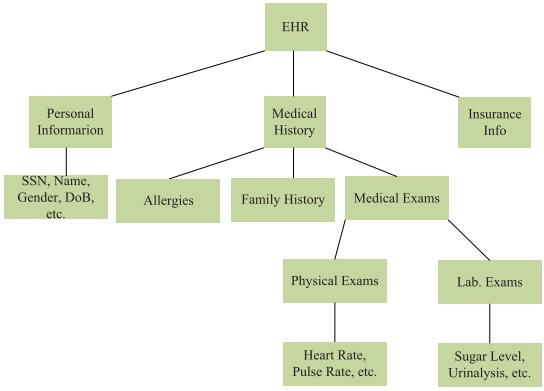

FIGURE 2. Hierarchical arrangement of a patient's EHR showing the categories of the gathered and stored data.

B. PRIVACY CHALLENGES IN COLLABORATIVE EHEALTH

In [4], a number of privacy challenges that are likely to be encountered in collaborative eHealth environments are presented and some of which are summarized as follows.

1) LOSS OF CONTROL OVER THE OUT-SOURCED DATA

The patient has minimal control over her data once it gets outsourced to the cloud. In turn, the data gets exposed to several privacy breach threats, which include, among others:

- **Malicious insiders**: A malicious insider is an employee of the cloud service provider who illegally takes the patient's data either for economic reasons or just to hurt the patient.
- **Data loss**: The data loss can be because of malicious attacks, accidental deletions by the cloud service providers, and natural disasters among others. Apart from providing notifications for unauthorized data modifications, data integrity maintenance can help in recovering lost data.
- **Data breaches**: Because of the multi-tenancy nature of the cloud, compromises that happen in one virtual environment may affect others in another virtual environment residing in the same physical machine.
- **Insecure interfaces and APIs**: APIs provide interfaces through which patients can access and control their data. Thus, the interfaces need to be secure to prevent intruders from accessing the out-sourced data. An insecure interface can grant wholesome access rights to intruders.

2) VIRTUALIZATION ISSUES

Virtualization plays a significant role in the clouds. However, it has presented cloud systems with a critical security challenge. Side channel attacks are known to exploit the nature of cloud virtualization to learn secret information. The attacker resides in the same physical machine as the victim and both share the same cache and processor. As the attacker takes

turns with the victim in sharing these resources, he can get vital secret information about the victim [3].

C. ACCESS CONTROL MECHANISMS TO MITIGATE THE PRIVACY CHALLENGES

To mitigate the stated privacy challenges, we present several access control methods that have been suggested for the cloud over the years. In [47], Tourani *et al.* have categorized these methods into, Encryption-based and Encryption-independent methods.

1) ENCRYPTION-INDEPENDENT ACCESS CONTROL METHOD

In this approach, the access control does not depend on any encryption mechanism. An example, in this case, is the use of an access control server to verify the credentials of the requesting data user. This approach requires patients and service providers to collaborate and create access policies. The access policies are then stored and leveraged by the access control server to authenticate and allow the requesting data users.

However, this approach has some drawbacks. First, the use of the access control server introduces additional communication and computation overhead. Also, the access control server can become a single point of failure and has to be always online. Lastly, dynamic access rights revocation is a challenge.

2) ENCRYPTION-BASED ACCESS CONTROL METHOD

In this approach, the patient encrypts her data before uploading it to the cloud. Data users authenticate themselves before decrypting or been given a decryption key to decrypt the data for use. Public key infrastructure-based access control, identity-based access control and attribute-based access control fall under this category.

a: PUBLIC KEY INFRASTRUCTURE-BASED ACCESS CONTROL

Public key infrastructure (PKI) is the set of hardware and software components, policies and processes used to create and manage digital certificates and public keys. It enables the establishment of user identities through their public keys, essential in the provision of controlled access to cloud data. Certificate authorities do the binding of public keys to their respective user identities. Once a user's identity is successfully verified, she is issued with a decryption key and granted access to the encrypted data.

The main drawback of PKI-based access control in collaborative eHealth is unscalability. In collaborative eHealth, data users come in and out dynamically. Under PKI-based access control, each of these data users has to obtain a certificate for their authentication which can be time wasting and cumbersome. Secondly, management of private keys in PKI-based access control is a challenging task. Any exposure of the private key compromises the security of the entire system.

TABLE 2. A summary of the access-control methods.

Access-control approach	Key points
Encryption-independent	- Access-right revocation is not flexible. - Additional communication and computation overhead.
Encryption-based a) PKI-based	- Unscalable. - Key management problem.
b) Identity-based	- One-to-one scheme. - Unsuitable for collaborative eHealth environment.
c) Attribute-based	- One-to-many scheme. - A better candidate for collaborative eHealth enviroment.

b: IDENTITY-BASED ACCESS CONTROL

The private key management problem in PKI is solved using identity-based access control. In identity-based access control, public keys are the users' identities. A data user is authenticated based on her identity. In this case, the ciphertext and the decryption key are associated with an identity. Decryption is successful if the ciphertext and decryption key identities match. As a drawback, the identity-based access control is a one-to-one scheme unsuitable for fine-grained access control and data sharing in cloud-based systems.

c: ATTRIBUTE-BASED ACCESS CONTROL

In attribute-based access control, the ciphertext and the decryption keys are labeled with attributes. A data user successfully decrypts the ciphertext if her attributes match the attributes in the access policy. The access policy is embedded either in the ciphertext or the decryption key. It is a one-to-many scheme enabling the same data to be encrypted for multiple data users, thus making it a fine-grained access control scheme and suitable for data sharing in collaborative eHealth.

A summary comparing the various access-control mechanisms is presented in Table 2.

D. AN INTUITIVE USE-CASE EXAMPLE

To elaborate on the necessity of a suitable access control mechanism in collaborative eHealth, we present an intuitive example scenario as follows.

Suppose *Alice* is a diabetic patient who suffers from hypertension and kidney problems. Assuming *Alice* is to undergo kidney treatment in *Hospital A* next month. *Dr.Bob* or any other doctor who works in *Hospital A* and is responsible for *Alice*'s treatment has to access *Alice*'s EHRs to study her medical history. Furthermore, assume that last week *Alice* was admitted in *Hospital B* for diabetes treatment. In *Hospital B*, *Dr.Charles*, who was responsible for *Alice*'s treatment, advised her to join a *Diabetes Management Program* offered by *Hospital B*. As a result, *Alice* is given a smart medical sensor-based gadget that monitors and records her activities, location history and calorie level on a daily basis. *Alice* also joins a *Diabetes Social Media Group* that mandates sharing of her personal information with the group members (e.g., diet and sugar level) on a weekly basis. Owing to *Alice*'s condition, she wants to allow full access of her EHRs to her *Family Members* and occasionally to an

Emergency Department. Moreover, owing to *Alice*'s participation in the *Diabetes Management Program*, her *insurance company* promises to reduce her premium if she shows significant improvement in her diabetic condition. Let us also assume due to *Alice*'s condition, her home is equipped with health sensors to detect and report emergency cases.

This is a dynamic scenario in which *Alice*'s recordings can be merged into an EHR and stored in the cloud and made accessible only to the authorized parties. The created EHR can be formatted into an XML format among other formats for the cloud storage [48]–[50] as used in the Indivo [51] system whose implementation is based on continuity of care record (CCR) and continuity of care document (CCD) standards.

1) EHR ACCESSIBILITY AND SECURITY CONCERN

In the above-presented example scenario, the patient is *Alice* who is the data owner. The staff of *Hospital A* and *Hospital B*, the *Insurance Company*, the *Social Media Group Members*, the *Emergency Response Staff*, and *Alice*'s relatives are the users. Assuming due to privacy concerns, users are supposed to have limited access to the stored EHR, i.e., each user only views some selected portions of the EHR. However, without a proper access control mechanism in place, some users may intentionally or accidentally cross their boundaries and access more than they are supposed to access.

The access control server, identity-based access control, PKI-based access control and attribute-based access control mechanisms among other methods can be leveraged to regulate access in this scenario. However, considering the advantages and disadvantages of the stated access control mechanisms, and bearing in mind the dynamic nature of the stated scenario, the attribute-based access control method offers the most suitable features to regulate access to the patient's data.

2) REGULATING ACCESS TO THE EHR WITH ATTRIBUTE-BASED ACCESS CONTROL

Using an ABE scheme, *Alice* can encrypt her EHR for access by the target users based on an access policy. For instance, *Alice* can encrypt her EHR for cloud storage allowing only doctors who work in *Nephrology* department in *Hospital A* to have access to it. The policy used to achieve this under ABE looks like this: *[(Organization = Hospital A) AND (Department = Nephrology) AND (Profession=Doctor)]*. Any other user who tries to have access to the ciphertext has to satisfy the defined policy. *Dr.Bob* who is working in the department of *Nephrology* in *Hospital A* can satisfy the access policy and decrypt this ciphertext. Meanwhile, *Dr.Charles* who is working in *Hospital B* cannot satisfy the access policy and thus, he cannot decrypt the ciphertext.

E. SECURITY REQUIREMENTS IN COLLABORATIVE EHEALTH

An ideal scheme has to meet the following requirements for its effective adoption in collaborative eHealth systems [18], [52].

- *Data confidentiality:* As there are multiple users in collaborative eHealth, the most important question to be answered by the data owners and authorities is, ''who accesses what?'' As a result, a scheme should allow data owners to define policies to help determine who accesses what data. Mostly, accessibility is always a balance between security and quality of service. Exposing more information improves the quality of the healthcare service but may compromise the security of the system. Exposing little information improves security at the expense of reducing the quality of healthcare services. Additionally, the system has to prevent users from colluding with each other to compromise the confidentiality of the patient.
- *Policy flexibility:* The success of a security scheme in the collaborative eHealth depends on its access policy flexibility. The scheme should allow access policies to be modified at any given time without re-running the whole setup. This access policy flexibility comes in handy in cases of emergency.
- *Attribute/user revocation:* An ability to revoke access rights from a user is paramount in collaborative eHealth. This can be done through revoking attributes from a user for reasons that, the user may be considered as a security threat or her relevance is considered being obsolete.
- *Scalability and Efficiency:* The scheme should be scalable, allowing the addition of new users. The addition of the new users should be smooth without affecting the existing users. Power usage, communication, storage and computation overheads, together with the key management complexity, should be minimal to make the system usable.
- *Traceability:* An ideal security scheme should be able to trace accidental or intentional access key leakages by users. This helps in building user trust levels.

IV. ATTRIBUTE BASED ENCRYPTION

Attribute-based encryption (ABE) is a one-to-many public key cryptography proposed by Sahai *et al.* [11]. In the basic ABE scheme, there are three players involved: a trusted authority, a data owner and data users [53].

The trusted authority generates public keys (*pp*) and a secret master key (*mk*). The data owner uses the public keys received from the authority to encrypt the data and store them remotely in a cloud. The authority also generates a secret key for each data user using the secret master key along with a set of attributes. The user is allowed to decrypt the ciphertext only if her set of attributes match the attributes included in the ciphertext.

In the rest of this section, we present the ABE usage in collaborative eHealth, a summary of the mathematical concepts and access structures used in its construction, and the different categories of ABE and some of their limitations.

TABLE 3. Notations.

Notation	Meaning
$\mathbb{G}, \mathbb{G}_T$	Bilinear Groups
p	Order of the groups
g	Group generator
e	Bilinear map
κ	Security parameter
mk	Master Key
pp	Public parameter
CT	Ciphertext
CA	Central Authority
AA	Attribute authority
n_u	Total number of users
N_{AA}	Number of attribute authorities
n_{c_A}	Number of attributes in a ciphertext
n_{nu,r_A}	Number of non-revoked users with revoked attributes
$n_{aa,k}$	Number of attributes managed by an attribute authority
$n_{c,r}$	Number of ciphertexts with revoked attributes
$n_{aa,u,k}$	Number of attributes assigned to a user by an authority
n_{u_A}	Number of user attributes
n_{cs}	Number of ciphertexts
n_o	Number of owners

A. PRELIMINARIES

First, we begin by summarizing the fundamental concepts used in ABE scheme constructions: Bilinear maps and access structures.

1) BILINEAR MAPS [54]

Let, $\mathbb{G}$ and $\mathbb{G}_T$ be cyclic multiplicative groups of order p, and let g be a generator of group $\mathbb{G}$. A bilinear map e, is defined as $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$, subject to satisfaction of the following conditions:

- Bilinearity condition, i.e., $e(g^a, g^b) = e(g, g)^{a,b}$ for all a and b.
- Non-degenerate condition, i.e., $e(g, g) \neq 1$.
- e is computationally feasible.

Some of the notations used in this study are summarized in Table 3.

2) ACCESS STRUCTURES

Access structures are the formal representations of user-defined access policies used during ABE scheme constructions. Several access structures have been proposed over time and we present summaries of some of them as follows.

- **Threshold Gates**: In threshold gate access structure, an access tree is constructed in which the interior nodes of the tree are *AND* and *OR* gates and the leaf nodes are associated with the attributes. A decryption secret s is distributed to the leaf nodes of the access tree as shown in Figure 3. To perform decryption, one needs to satisfy or regenerate the secret, if the parent node is an AND-gate, all the children need to be satisfied to regenerate the secret but if the parent node is an OR-gate, only one child is needed to regenerate the secret. From the illustration in Figure 3, apart from the doctors, only a nurse in hospital A (Hos. A) can regenerate the decryption secret allowing them to perform the ciphertext decryption.

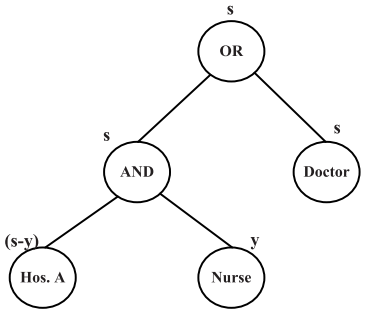

FIGURE 3. An access tree in threshold-gate access structure showing distribution of a secret amongst attributes.

- **ANDing** operation on multivalued attributes is another access structure being used in the construction of ABE schemes. The ANDing can be done on both positive and negative attributes. The AND-gate access structure also takes care of ''don't care'' attributes by using the wildcard * sign as their attribute values. For example, an AND-gate access structure used to permit nurses in hospital A to have access to some data can be written as: *(Organization: Hospital A AND Department: * AND Profession: Nurse)*, i.e., the department value does not matter and so long as the nurse works in hospital A, she will access the data.
- **Linear Secret Sharing Scheme (LSSS) [55]–[57]**: LSSS is another popularly used access structure in the construction of ABE schemes. In LSSS access structure, there is a matrix $\mathbb{M}$ referred to as the share-generating matrix. $\mathbb{M}$ has l rows and n columns. Rows of $\mathbb{M}$ $(1, \cdots, l)$ are mapped to the attributes using a function $\rho(i)$. Suppose the secret to be shared between the attributes is s. A column vector $v = (s, r_2, \cdots, r_n)$ is chosen where, $r_2, \cdots, r_n \in \mathbb{Z}_p$. Using techniques of matrix multiplication, it can be seen that, $\mathbb{M}v$ generates l shares of the secret s and $\mathbb{M}v_i$ is a secret share for attribute $\rho(i)$. The l shares are then reconstructed during decryption to regenerate s.
- **Ordered Binary Decision Diagram (OBDD) [58]**: Recently, OBDD has been proposed for usage as an access structure during ABE scheme construction. In an OBDD access structure, an OBDD tree is constructed in which apart from the leaf nodes which are binary 1 and 0, the rest of the nodes represent the attributes in the access policy as shown in Figure 4. The order of the attributes is pre-defined. Each path from the root node to the leaf node 1 is considered to satisfy the tree. If a user possesses a set of attributes that can satisfy the tree, she is permitted to decrypt the ciphertext. Otherwise, she does not have permission to decrypt the ciphertext. Figure 4 is an OBDD access structure representation of the access policy we presented in the threshold gate access structure section, permitting doctors and all the nurses in hospital A to access the data.

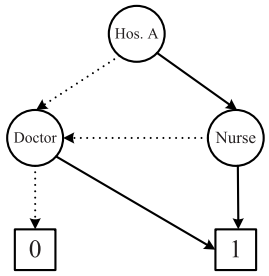

FIGURE 4. An OBDD tree showing generation of valid paths from root node to the leaf node 1. The bold arrows indicate the attribute is present and the dotted arrows indicate the attribute is absent.

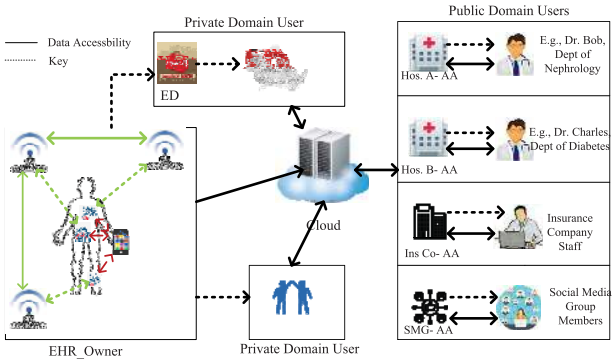

FIGURE 5. An illustration of the ABE usage in collaborative eHealth showing the categorization of the data users. The dotted arrows indicate key grants and the bold arrows indicate EHR accessibility.

These presented access structures have some limitations as well: The secret distributions and re-generations in LSSS and threshold-gate access structures increases the number of exponentiation operations in the ABE scheme which can be computationally demanding, the AND-gate access structure does not permit a repetition of attributes, and the OBDD access structure is still new and not well studied yet as it still lacks mechanisms to represent different values of an attribute.

B. ATTRIBUTE BASED ENCRYPTION IN COLLABORATIVE EHEALTH

In the context of collaborative eHealth, there is a slight modification to the basic architecture of the ABE scheme. In an ideal collaborative eHealth system, different categories of users and multiple trusted authorities are involved, and as a result, data users are divided into two groups *private domain users and public domains users* [18], [59] as shown in Figure 5.

Private domain users (e.g., friends, relatives, etc.) have a personal relationship with the data owner and are considered to be fewer in number. The EHR owner is an attribute authority to personal domain users. In other words, a single-authority scheme is suitable for managing personal domain users. Meanwhile, public domain users (e.g., doctors, nurses, pharmacists, researchers, insurance companies, etc.) are professional data users who use the data for professional purposes and are considered being many in number.

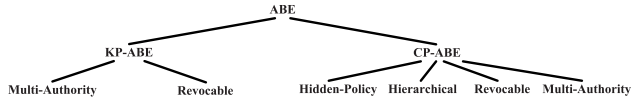

FIGURE 6. Categories of attribute-based encryption.

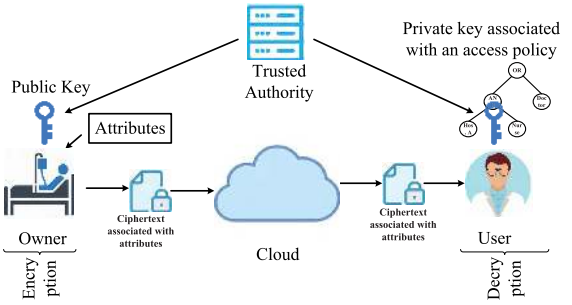

FIGURE 7. Parties and operations in KP-ABE. Owner and user refer to data owner and data user respectively.

The data/EHR owner manages the decryption keys of the private domain users. The data/EHR owner can add or remove them whenever needed. Public domain users are managed by different and multiple attribute authorities and their decryption keys for accessing the stored cloud-based EHR are generated and controlled by those attribute authorities. Likewise, those attribute authorities are responsible for adding or removing users from the public domain category.

The cloud servers are assumed to be honest but curious [60]–[63], i.e., they try to learn as much information as possible but they follow the procedure as it is.

C. PRIMARY CATEGORIES OF ATTRIBUTE BASED ENCRYPTION

Primarily, attribute-based encryption schemes are divided into two categories. The other categories shown in Figure 6 are extensions of the two primary categories. In this section, we discuss the primary categories as follows.

1) KEY POLICY ATTRIBUTE-BASED ENCRYPTION (KP-ABE)

In KP-ABE [12], the private keys are associated with access policies, while the ciphertexts are associated with attributes. The data owner encrypts the data using the public keys obtained from the trusted authority and a set of attributes. The data user is able to decrypt the data if and only if, the attributes incorporated in her private key access policy match the attributes in the ciphertext. The user private key is associated with an access policy and is obtained from the trusted authority. An illustration is shown in Figure 7.

The KP-ABE scheme is implemented in four different algorithms, and its construction is based on the mathematical fundamentals summarized in section IV-A and proceed as follows.

- **Setup**($\kappa \rightarrow pp, mk$): Takes a security element κ and outputs the master key (mk) as, $\alpha \in \mathbb{Z}_p$ and the public keys (pp) as, $(g, e(g, g)^{\alpha}, H(i) \forall i \in U)$, where, $H(i) \in \mathbb{G}$ and H is a hash function and U is an attribute universe.

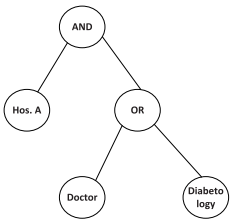

FIGURE 8. Monotonic access structure.

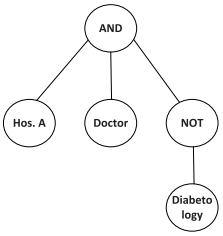

FIGURE 9. Non-monotonic access structure.

- **KeyGeneration**$(\tau, mk \rightarrow sk)$: Takes the access policy τ and the master key mk and outputs the private key sk corresponding to attributes in the access policy. sk is produced as: $(g^{\lambda_i} H(i)^{r_i}, g^{r_i})$, where $r_i \in \mathbb{Z}_p$ and λ_i is a secret share generated using LSSS.
- **Encryption** $(M, S \subseteq U \rightarrow CT)$: Takes as input the data M, a set of attributes S and outputs a ciphertext CT associates with the attribute set S. CT is produced as $(Me(g, g)^{\alpha s}, g^s, H(i)^s{}_{i \in S})$, where $s \in \mathbb{Z}_p$.
- **Decryption**$(CT, sk, pp \rightarrow M/\perp)$: Takes as input the ciphertext CT, the data user private key sk and the public keys pp to recover the encrypted data M. The decryption is successful if and only if the data user attributes satisfy the ciphertext attributes. Otherwise, the algorithm outputs $\perp$.

The construction of this basic KP-ABE is based on monotonic access structure, i.e., the used access structure does not permit the inclusion of negative attributes.

An improvement to this basic KP-ABE scheme was done by Ostrovsky *et al.* [64] using a non-monotonic access structure, i.e., an access structure that allows the inclusion of negative attributes. The access structure is constructed using **AND, OR, NOT** and threshold gates. This makes it possible to represent any access policy as negative attributes can be included in the access structure. We present a comparative illustration of the monotonic and non-monotonic access structures in Figure 8 and Figure 9. In Figure 8, the access structure allows a user in Hospital A, who is a doctor or in the Diabetology Department to decrypt a data. This can be written as *{Hospital A AND Doctor OR Diabetology}*. However, in a circumstance the data owner does not want to grant access to the Diabetology Department staff, a non-monotonic access structure provides the negation ability to deny the accessibility as shown in Figure 9 and the access policy, in this case, can be written as *{Hospital A AND Doctor NOT Diabetology}*.

Ostrovsky's approach has a problem of communication overhead as every attribute has a respective negative version in the system. This makes the ciphertext too large and undesirable for the network.

The ciphertext sizes in KP-ABE schemes grow linearly with an increase in the number of attributes. Attrapadung *et al.* [65] proposed a KP-ABE scheme based on non-monotonic access structure with fixed ciphertext size. Their scheme additionally reduces the number of pairing operations during decryption. On the downside, their proposed scheme has a drawback of quadratic increase in key sizes.

Most of the existing ABE schemes are based on bilinear pairing, which is resource demanding and unsuitable for resource-constrained devices [66]. The emergence of the internet of things (IoT) and mobile devices which are having tremendous influences in our lives [67]–[70], including the health sector, require computationally less demanding privacy provision schemes. As a result, Yao *et al.* [71], suggested a lightweight ABE scheme for resource-constrained devices. Their scheme is a KP-ABE because it encrypts data based on a set of attributes and incorporates access policies in the user private keys. The key difference with the other KP-ABE schemes such as [72] is that it is based on elliptic curve cryptography (ECC) instead of bilinear pairing.

Generally, KP-ABE schemes have a drawback of being less expressive and not suitable for collaborative eHealth systems as the access right is associated with the private keys but not with the ciphertext. Any user who manages to obtain the private key can decrypt the ciphertexts as the ciphertexts are not self-protective. The strength of this scheme lies in the hope that the private keys are only given to legitimate users. KP-ABE schemes are best suited for pay-per-view channels, audit logs, and targeted broadcast. This limitation led to the proposal of the CP-ABE scheme in which the access rights are embedded in the ciphertexts making them self-protective.

2) CIPHERTEXT POLICY ATTRIBUTE-BASED ENCRYPTION (CP-ABE)

CP-ABE was proposed by Bethencourt *et al.* [73] and is like the Role-based access control scheme [74]. In the CP-ABE scheme, private keys are associated with user credentials and ciphertexts are associated with access policies as shown in Figure 10. The data owner receives the public key from the trusted authority and defines an access policy. The two are then used to encrypt the data. Using the private key received from the trusted authority, the data user decrypts the data if her attributes satisfy the access structure embedded in the ciphertext. CP-ABE has gained popularity for access control of electronic health records in clouds, social networking sites and secure key exchanges in fog computing [75].

Similar to the KP-ABE, implementing the basic CP-ABE scheme involves four (4) algorithms and using the mathematical fundamentals summarized in section IV-A, it proceeds as follows.

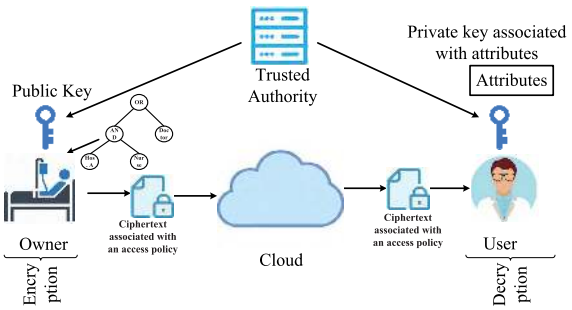

FIGURE 10. Parties and operations in CP-ABE. Owner and user refer to data owner and data user respectively.

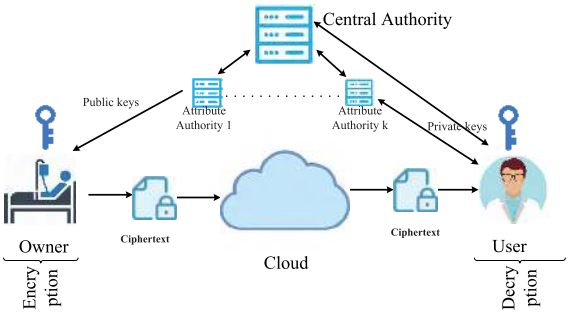

FIGURE 11. Centralized multi-authority attribute based encryption.

1) **Setup**($\kappa \rightarrow mk, pp$): Takes the security parameter κ as input and outputs a master key mk as (β, g^{α}) and public keys pp as, $(\mathbb{G}, g, h, f, e(g,g)^{\alpha})$. $h = g^{\beta}$, $f = g^{\frac{1}{\beta}}$. Where: $\alpha, \beta \in \mathbb{Z}_p$.
2) **KeyGeneration**($mk, S \rightarrow sk$) This algorithm takes the master key mk and a set of data user attributes S to produce a secret key sk as, $(D = g^{(\alpha+r)/\beta}, \forall j \in S : D_j = g^r.H(j)^{r_j}, D'_j = g^{r_j})$, where: $r, r_j \in \mathbb{Z}_p$ and H is a hash function.
3) **Encryption** $(pp, M, \tau \rightarrow CT)$: Takes as input the public keys pp, the data M and a threshold-gate access structure τ and outputs a ciphertext CT as $(\tau, \tilde{C}, C, C_y, C'_y)$, where $\tilde{C} = Me(g,g)^{\alpha s}$, $C = h^s, \forall y \in Y : C_y = g^{q_y(0)}, C'_y = H(att(y))^{q_y(0)}$, H is a hash function, $s \in \mathbb{Z}_p$, q is a polynomial such that $q_x(0) = s$ and Y represents the attributes in the access structure.
4) **Decryption** $(CT, sk \rightarrow M/\perp)$. The decryption algorithm takes as input the ciphertext CT and the private key sk and outputs the decrypted data M. The decryption is successful only if the user attributes satisfy the access structure included in the ciphertext. Otherwise the output is $\perp$.

This basic CP-ABE achieves the objectives of efficiency, expressibility, and security [76]–[78]. However, its construction is based on monotonic access structure. Efforts to improve the expressibility of this scheme have been made through the use of non-monotonic access structures.

As a result, AND-gate [79] and OBDD [58] are some non-monotonic access structures being used in the construction of the CP-ABE schemes. These access structures support the use of both positive and negative attributes in the access structure. However, as stated earlier, the AND-gate access structure does not allow the repetition of attributes within an access structure which limits the expressibility of the resulting scheme. OBDD has no such restrictions, and it appears to allow various expressions to be represented effectively in an access structure.

The CP-ABE schemes similarly suffer from the problem of ciphertexts increasing linearly with the increase in the number of attributes. Herranz *et al.* [80] proposed a CP-ABE scheme with constant ciphertext sizes. Their scheme is constructed for threshold case, i.e., if a data user possesses at least t attributes, she is allowed to decrypt the ciphertext. Zhou *et al.* [81] proposed a CP-ABE scheme with constant size ciphertext with user anonymity functionality. Their scheme hides the attribute values of attributes making them anonymous in the ciphertext.

D. SECONDARY CATEGORIES OF ATTRIBUTE BASED ENCRYPTION

The other categories of attribute-based encryption adapt some features of either of the primary categories and are discussed as follows.

1) MULTI-AUTHORITY ATTRIBUTE BASED ENCRYPTION (MA-ABE)

The previously presented ABE schemes are single authority ABE schemes. Single authority ABE schemes are ABE schemes where only one trusted authority is responsible for setup and key generation tasks. First, these schemes experience performance degradation with an increase in the number of users in the system. Second, the schemes are inadequate for systems where certain users receive attributes from over one authority. Third, the authority can decrypt all the ciphertexts as it has all the user keys. Chase [82] proposed multi-authority ABE scheme whereby users' attributes are managed different trusted authorities to solve single authority problems.

The suggested multi-authority ABE system is made up of K attribute authorities and a central authority (CA). Every user in the system has a unique global identifier (GID) such as name or SSN verifiable by all the authorities. The attribute authorities run pseudo-random functions (PRF) on users' GIDs to generate secret keys for the users. The CA does not get any attribute information from users but provides users with setup keys. We illustrate the scheme in Figure 11.

In this scheme, a user's attributes are distributed amongst attribute authorities. Each attribute authority generates a key for the user depending on the access structure τ_k, chosen for the user. The user can only decrypt the ciphertext if and only if a subset of attributes, A_C^k, from an attribute authority satisfies the access structure τ_k. This scenario is divided into five phases (setup, attribute authority k, central authority, encryption, and decryption) and proceeds as follows.

1) **Setup:** Is run by the central authority. Generates the public parameters and master key. Two groups $\mathbb{G}$ and $\mathbb{G}_T$ of prime order p are chosen. Let g be a generator for $\mathbb{G}$, and the bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is defined. $s_1, \ldots, s_K$ and y_0 are randomly chosen from $\mathbb{Z}_p$, and $g_2 \in \mathbb{G}_T$ is also chosen. The public parameter pp of the system is $g_1 = g^{y_0}$.
2) **Attribute Authority k:** Attribute authority k receives its secret key s_k from the CA. The authority's public key is $t_{k,1}, \ldots, t_{k,n+1} \leftarrow \mathbb{G}_T$. A polynomial $h(x)$ of degree n defined by $t_{k,1}, \ldots, t_{k,n+1}$ is chosen, and $T_k(x) = g_2^{x^n} g^{h(x)} = g_2^{x^n} \Pi_{i=1}^{n+1} t_{k,i}^{\Delta_i(x)}$ is computed. To generate a secret key for user u based on access structure τ_k, the master key of the attribute authority k becomes $y_{k,u}$, where $y_{k,u} = F_{sk}(u)$(pseudo-random function run on a user's GID), and the public key is $(g^{y_{k,u}}, g_2, t_{k,1}, \ldots, t_{k,n+1})$. A root node polynomial q_r is chosen for the access structure τ_k such that $q_r(0) = y_{k,u}$. Other nodes' polynomials are also chosen such that $q_x(0) = q_{parent(x)}(x)$. For every leaf node x, $r_{k,x}$ is randomly chosen to compute the secret key as $(D_{k,x}, R_{k,x})$, where $D_{k,x} = g_2^{q_x}(0) T(i)^{r_{k,x}}$, $i = att(x)$, and $R_{k,x} = g^{r_{k,x}}$.
3) **Central Authority:** Generates the secret key for all the k authorities as y_0 and secret key (D_{CA}) for a user u having attributes in k authorities as, $(g_2^{y_0 - \sum_{i=0}^{K} y_{k,u}})$.
4) **Encryption:** To encrypt a message M under a set of attributes A_C, choose $s \in \mathbb{Z}_p$ randomly and the ciphertext CT is $(CT = e(g_1, g_2)^s M, CT' = g^s, \{CT_{k,i} = T_k(i)^s\}_{i \in A_C^k, \forall k})$.
5) **Decryption:** For every leaf node, the algorithm computes $\frac{e(D_{k,x}, CT')}{e(R_{k,x}, CT_{k,i})} = e(g, g_2)^{q_x(0)s}$. The results of the child nodes are combined to obtain $e(g, g_2)^{q_x(0)s}$ for their parent node. This continues until all the nodes in the access structure are fully decrypted. For a given subset of attributes that satisfy the access structure τ_k, the combined leaf decryption results into $e(g, g_2)^{q_r(0)s} = e(g, g_2)^{y_{k,u}s}$ for an authority. This is done for all the attribute authorities responsible for managing user u's attributes. Once completed, compute $Y_{CA}^s = e(CT', g_2^{y_0 - \sum_{i=0}^{K} y_{k,u}}) = (g^s, g_2^{y_0 - \sum_{i=0}^{K} y_{k,u}}) = e(g, g_2)^{y_0 s}$. The message is recovered as $CT / e(g, g_2)^{y_0 s}) = e(g_1, g_2)^s M / e(g, g_2)^{y_0 s}) = e(g, g_2)^{y_0 s} M / e(g, g_2)^{y_0 s}) = M$.

The strength of this scheme relies on the security strength of the CA, as it manages all the attribute authorities. Once the CA is compromised, the system breaks down. Second, the consistent use of the GID makes it possible for the CA to assemble all the users' keys making it possible for it to decrypt the ciphertexts. These affect the acceptance of this scheme for practical implementations.

Elimination of reliance on the CA is important for the practical implementation of multi-authority ABE schemes [83]–[85]. Han *et al.* [85] proposed a decentralized multi-authority ABE scheme as illustrated in Figure 12. In their scheme, all the attribute authorities have to be online and they can join and leave any time. However, their scheme does not prevent user collusion. Lin *et al.*'s work [84] used the capabilities of a distributed key generation protocol and joint zero secret sharing protocol [86] to eliminate the CA. However, to avoid collusion amongst attribute authorities, the scheme restricts the number of users to be no more than the number of attribute authorities, hence eliminating the attribute addition ability.

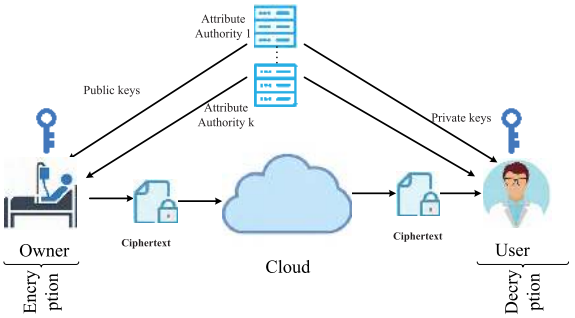

FIGURE 12. Decentralized multi-authority attribute based encryption.

Further studies by Li and Zang [87] and Wang *et al.* [88] have instead resulted in degradation of computational and decryption efficiencies. Li *et al.* [18] and Pussewalage *et al.* [59] have suggested multi-authority ABE schemes for EHR exchange in the cloud. However, in their schemes, the keys are associated with the access policies and the ciphertexts are associated with attributes which limit their expressiveness.

2) ATTRIBUTE BASED ENCRYPTION SCHEMES WITH REVOCATIONS

Revocation of attributes and users in ABE schemes is a desirable feature for practical applications. The basic ABE schemes do not provide mechanisms for removing misbehaving users. Once a user satisfies the access control requirements at the beginning of the setup stage, she continues to have access to the data irrespective of whether she breaks the security boundaries. Without a revocation mechanism, the only way such a user can be removed from the system is when the administrator re-runs the entire system all over again. Revocation methods in ABE are primarily categorized into two: *direct and indirect revocations*.

Direct revocation: In direction revocation, the revocation is specified during the encryption stage.

Wei *et al.* [89] and Liang *et al.* [90] (which extends [77]) introduced an approach for user revocation based on a binary tree shown in Figure 13. For each node in the tree, a random number is selected and stored in the node. Each user u has an associated path ($path(u)$) from the root node to a leaf node. For example, all the blue nodes from the root node to the blue leaf node form the $path(u)$ for the user represented by the blue leaf node. An algorithm *KUNode* is then used periodically to generated a minimum set of nodes ($minCover$)

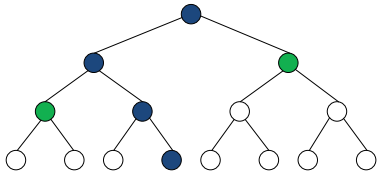

FIGURE 13. Revocation binary tree correspondong to time $\mathfrak{t}$. Users are assigned to the leaf nodes. Blue nodes represent the path for a revoked user. Green nodes are the minimum nodes that cover the non-revoked users.

that cover the non-revoked users (shown in the Figure 13 by the green nodes) such that, $path(u) \cap minCover = \emptyset$ for revoked users and $path(u) \cap minCover = \zeta$ for non-revoked users. ζ is used during decryption by the non-revoked users. A similar approach is employed by Yiliang *et al.* [91], the only difference being the replacement of the binary trees with the system state. In these schemes, the revocation is effected through the addition of three new algorithms on top of the four algorithms of the conventional CP-ABE as highlighted below.

- **KUNode**($\mathbb{T}, RL, \mathfrak{t} \rightarrow minCover$)**:** This algorithm takes as input the current attribute revocation list RL, the revocation time period $\mathfrak{t}$ and the revocation tree $\mathbb{T}$ and outputs a minimum cover set $minCover$. The $minCover$ is used by the non-revoked users (i.e., users who are not listed in the revocation list RL) during decryption.
- **Revoke**($UID, RL, \mathbb{T}, \mathfrak{t}' \rightarrow RL'$)**:** The algorithm takes as input the user identity UID (i.e, the ID for the user to be revoked), the revocation list RL, the revocation tree $\mathbb{T}$ and a new time period $\mathfrak{t}'$ and outputs a new revocation list RL'.
- **CTUpdate**($CT, \mathfrak{t}' \rightarrow CT'$)**:** The algorithm takes as input the ciphertext CT and a new time period $\mathfrak{t}'$ ($\mathfrak{t}' > \mathfrak{t}$) and outputs a new ciphertext CT' for the time period $\mathfrak{t}'$.

In summary, the revocation in these schemes is achieved through periodic broadcasting of key updates to unrevoked users. The scheme is inefficient because of the extra network traffic generation and their revocations are not immediate.

Pirretti *et al.* [92] proposed an ABE scheme in which expiration periods are included in users' keys. Users' keys are regenerated at the end of the expiration period. This makes the scheme inefficient in terms of performance degradation owing to frequent key re-generations, especially when the expiration period is short and inefficient security when the expiration period is longer.

In [93] and [94], whenever revocation happens, the data owner withdraws the entire ciphertext and re-computes the ciphertext components affected by the revocation before uploading it back to the cloud. This raises the computation burden on the data owner and thus, making the schemes impractical.

Indirect revocation: In indirect revocations, the revocations are dynamically performed by the trusted authority whenever required. In [95]–[102], CP-ABE schemes with attribute revocations are proposed. These schemes combine CP-ABE with proxy re-encryption to achieve the attribute revocation. In the schemes, whenever an attribute revocation happens, a new master key is generated and the corresponding public keys are updated. Then, the user attribute keys get updated for data access, except for the revoked user attribute. This is done through proxy re-keys generated for the proxies by the trusted authority. These proxy re-keys are further used to update the ciphertexts stored in them. All the items (i.e., attributes, keys, ciphertext, etc,) are assigned a version number which is incremented whenever an update happens. In [98], the scheme efficiency is enhanced by mapping the access policy to a weighted access list which minimizes the computation overhead during secret key and ciphertext generation.

These mechanisms require three additional algorithms on top of the Bethencourt's [73] CP-ABE algorithms and they are summarized as follows.

- **ReKeyGen**($\gamma, mk \rightarrow mk', pp', rk$)**:** Takes an attribute set γ which includes attributes to be updated and the master key mk to be updated as inputs and outputs a new master key mk', updated public keys pp' and proxy re-key rk for the attributes in the attribute universe. Version is incremented by 1.
- **ReEnc**($CT, rk, \beta \rightarrow CT'$)**:** Takes the ciphertext CT, proxy re-keys rk and a set of attributes β which includes attributes in the ciphertexts access structure whose proxy re-keys are not 1 as input and outputs a new ciphertext CT'.
- **ReKey**($\overline{D}, rk, \theta \rightarrow sk'$)**:** Takes as input the attribute component $\overline{D}$ of the user secret key sk, proxy re-keys having the same version as the user secret key sk and attributes in sk whose proxy re-keys in rk are not 1 and outputs an updated user secret key sk'.

However, [99]–[102] schemes rely on the centralized multi-authority ABE and hence suffer from the weaknesses of the centralized multi-authority ABE discussed earlier in section IV-D.1.

Further studies [103]–[106] have used the proxy re-encryption technique for attribute revocation. Generally, when using the proxy re-encryption technique for revocations, the proxy server is in charge of revocation related tasks. In [17], [107], and [108], the proxy keeps part of the user's decryption key and revocation information. During a decryption process, if a user is in the revocation list of the proxy, the proxy denies giving part of its stored decryption key to the user or it regenerates a different access structure for the unrevoked users or it does not generate a decryption token for the revoked user. As a result, the users included in the revocation list cannot be in a position to decrypt the ciphertext. This makes revocations immediate but has several drawbacks; first, the proxy has to be online all the time. Second, the proxy becomes a single point of failure, and last, fine-grained access control cannot be achieved through this mechanism.

In [37] and [109]–[111], the attribute group concept is leveraged for attribute revocation. In this concept, users bearing the same attribute are made to belong to the same attribute

group. A user can belong to multiple attribute groups. Each group has an associated group key G_{key} only known to its members. Whenever a user is revoked from the group, a new group key G'_{key} is generated and made available to the group's members except for the revoked user. The following are the significant algorithms integrated onto the conventional CP-ABE schemes to achieve the revocation.

- **Re-encrypt**$(CT, \{G_{key}\} \rightarrow CT')$**:** The algorithm takes the ciphertext CT and a set of attribute group keys $\{G_{key}\}$ as inputs and produces a new ciphertext CT' which is associated with the attribute groups as its output.
- **KeyUpdate**$(sk \rightarrow sk')$**:** This algorithm is executed by the non-revoked group members. The algorithm takes as input the secret key sk and produces a new secret key sk' as its output.
- **CTUpdate**$(CT', \{G'_{key}\} \rightarrow CT'')$**:** The algorithm takes as input the ciphertext CT' and a set of new group keys $\{G'_{key}\}$, and produces a new ciphertext CT'' as its output.

Indirect revocation method has an advantage of reducing the burden on the data owner as she does not have to know about the revocation computations. However, it increases computational overhead on the trusted authority and the participating proxies. Also, the passing of new key components to the users increases the system communication overhead in some circumstances.

3) HIERARCHICAL ATTRIBUTE BASED ENCRYPTION

Although the previously discussed ABE schemes provide fine-grained access to data, they are not suitable for large enterprises due to their inability to offer a full delegation. As a result, Wang *et al.* [112] proposed the hierarchical attribute-based encryption scheme (HABE) by combining hierarchical identity-based encryption (HIBE) with CP-ABE to achieve full delegation and high performance while providing fine-grained access control. HABE uses a combination of the techniques of Gentry and Silverberg's work on HIBE [113] to generate keys and CP-ABE based on the disjunctive normal form (DNF) access control policy to achieve the stated results. Parties involved in the HABE scheme include: the cloud service provider, an enterprise user (e.g., a company), a trusted third party (TTP), end users (e.g., personnel in the company), and internal trusted parties (e.g., a department that delegates keys to users inside the company).

The TTP is considered to be a root master (RM) and performs the role of a private key generator (PKG). An enterprise user represents a domain and has many domain masters (DMs) representing internal trusted parties (ITPs). A DM either delegates keys to the next level DMs or distributes keys to the end users or performs both tasks.

In summary, the key generation in this scheme is done in a hierarchical manner. The RM generates public parameters and a master key for the first level DM, which in turn generates keys for the next level DM. The last DM in the hierarchy generates the keys for users directly attached to it. We show an illustration of the hierarchical delegations in Figure 14. The entire process is broken down into; Setup,

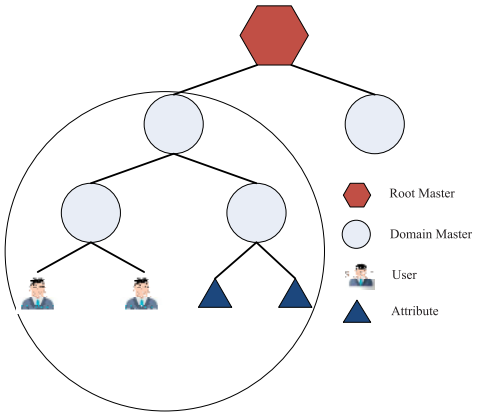

FIGURE 14. Delegations in hierarchical attribute based encryption.

CreateDM, CreateUser, Encryption, and Decryption algorithms [112] [45] and proceed as follows.

1) **Setup**$(\kappa \rightarrow mk, pp)$. This algorithm is run by the RM and it generates the public parameters and a master key by taking in a security parameter κ. Two groups $\mathbb{G}$ and $\mathbb{G}_T$ of order p are first chosen. Let P_0 be a random generator of $\mathbb{G}$, a bilinear map, $e := \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$, and mk_0 is chosen from $\mathbb{Z}_p$. The public parameters, pp are $(p, \mathbb{G}, \mathbb{G}_T, e, n, P_0, Q_0, H_1, H_2, H_A)$, and the master key, MK_0, is mk_0, where $H_1 : \{0,1\}^* \rightarrow \mathbb{G}$, $H_2 : \mathbb{G}_T \rightarrow \{0,1\}^n$, and $H_A : \{0,1\}^* \rightarrow \mathbb{Z}_p$ are random oracles, and $Q_0 = mk_0 P_0 \in \mathbb{G}$.
2) **CreateDM**(pp, MK_i, PK_{i+1}). This algorithm is executed by either the RM or DM to generate the master key for the next hierarchical level DM, (DM_{i+1}). From the current level, the next level DM has a decryption key D_{i+1} and public key PK_{i+1}. All the entities have their own unique IDs and the public key of an entity in the system comprises of the public key of its monitoring DM and its own ID. Therefore, $PK_{i+1} = (PK_i, ID_{i+1})$. The RM or current DM takes as input its own master key MK_i, public parameters, pp, and public key, PK_{i+1}, of the next level DM to generate DM_{i+1}'s master key, MK_{i+1}, as $(mk_{i+1}, D_{i+1}, Q - tuple_{i+1})$, where mk_{i+1} is randomly chosen from $\mathbb{Z}_p$, $D_{i+1} = D_i + mk_i P_{i+1}$, $P_{i+1} = H_1(PK_{i+1}) \in \mathbb{G}$, $Q - tuple_{i+1} = (Q - tuple_i, Q_{i+1})$, $Q_{i+1} = mk_{i+1} P_0 \in \mathbb{G}$, and assuming D_0 as an identity element of $\mathbb{G}$ and $Q - tuple_0 = Q_0$.
3) **CreateUser**(pp, MK_i, PK_u, PK_a) generates a secret key for a user who monitors an attribute of interest. For a user, u having a public key, PK_u who is monitoring an attribute a with a public key PK_a to be given a secret key, the DM_i has to check whether a is being legibly administered by u. If true, it computes $mk_u = H_A(PK_u) \in \mathbb{Z}_p$ and $D_{i,u} = (Q - tuple_{i-1}, mk_i mk_u P_0 \in \mathbb{G})$ followed by a user attribute secret key based on the administered attribute, a as $D_{i,u,a} = D_i + mk_i mk_u P_a \in \mathbb{G}$, where $P_a = H_1(PK_a) \in \mathbb{G}$ and a tuple $Q - tuple_i$ is given as well. If false, "$NULL$" is generated.

4) **Encryption**$(pp, \mathbb{A}, \{PK_{aij}|1 \leq i \leq N, 1 \leq j \leq n_i\}, M)$. This encrypts message M under the DFN access control policy $\mathbb{A} = \vee_{i=0}^{N}(CC_i) = \vee_{i=1}^{N}(\wedge_{j=1}^{n_i} a_{ij})$, where N represents the number of conjunctive clauses CC_i in policy $\mathbb{A}$, and n_i is the number of attributes in the CC_i, and a_{ij} is an attribute in the CC_i. Furthermore, the attributes' public keys are $\{PK_{aij}|1 \leq i \leq N, 1 \leq j \leq n_i\}$ for attributes in $\mathbb{A}$. The generated ciphertext CT is $(\mathbb{A}, [U_0, U_{12}, \ldots, U_{1t_1}, U_1, \ldots, U_{N2}, \ldots, U_{Nt_N}, U_N, V])$, where $U_0 = rP_0$, $U_12 = rP_{12}$, $U_{1t_1} = rP_{1t_1}$, $U_1 = r\sum_{j=1}^{n_1} P_{a_{1j}}$, $U_{N2} = rP_{N2}$, $U_{Nt_N} = rP_{Nt_N}$, $U_N = r\sum_{j=1}^{n_N} P_{a_{Nj}}$, $V = M \oplus H_2(e(Q_0, rn_{\mathbb{A}}P_1))$, r is chosen randomly from $\mathbb{Z}_p$, $P_{ij} = H_1(PK_{ij})$ for $1 \leq i \leq N$ (number of CC_is) and $1 \leq j \leq t_i$ (DM level), and $P_{a_{ij}} = H_1(PK_{ij}, \ldots, PK_{it_i}, PK_{a_{ij}})$ for $1 \leq i \leq N$ (number of CC_is) and $1 \leq j \leq n_i$ (number of attributes in a CC_i). Note that $n_{\mathbb{A}}$ is the lowest common multiple (LCM) of $n_1, \ldots, n_N$.

5) **Decryption**$(pp, CT, D_{it_i,u}, \{D_{it_i,u,a_{ij}}|1 \leq j \leq n_i\}, Q - tuple_{i(t_i-1)})$. This algorithm recovers M by using the secret key of the last DM in the hierarchy, D_{it_i} and the secret keys of all its legibly administered attributes $D_{it_i,u,a_{ij}}$. The user computes

$$V \oplus H_2\left(\frac{e(U_0, \frac{n_{\mathbb{A}}}{n_i}\sum_{j=1}^{n_i} D_{it_i,u,a_{ij}})}{e(D_{it_i,u}, \frac{n_{\mathbb{A}}}{n_i}U_i)\Pi_{j=2}^{t_i} e(U_{ij}, n_{\mathbb{A}}Q_{i(j-1)})}\right)$$

$$= V \oplus H_2(e(Q_0, n_{\mathbb{A}}rP_1)) = M$$

This scheme gets complex in case an attribute is maintained by more than one domain master. Also, it does not take into account hierarchical attributes but it simply distributes the authorities that manage the attributes into different layers [112], [114], [115].

Further studies on HABE are done in [116]–[120]. Qi *et al.* [121] proposed a multi-authority HABE scheme using the decentralized multi-authority approach. Hierarchical attribute trees are constructed and all the attributes in the scheme are involved in the construction of the attribute trees, and each attribute is represented by an attribute tree and a path. The attribute trees are divided using the hierarchical clustering algorithm [122]. A user is only allowed to decrypt the ciphertext if her attribute path satisfies the ciphertext path. Their scheme is proved secure under the DBDH assumptions.

4) HIDDEN POLICY ATTRIBUTE BASED ENCRYPTION

In hidden policy ABE schemes, the ciphertext access structure is hidden. Remember, in CP-ABE, the access structure is included in the ciphertext and anyone who accesses the ciphertext can know the policy. Thus, this compromises policy secrecy. To improve privacy in ABE schemes, [26], [123]–[132] suggested the hiding of access structures. A user only decrypts a ciphertext if her attributes match the policy, and there is no way she can guess the policy included in the ciphertext.

There are two types of the hidden policy ABE schemes: *fully hidden and half hidden* policy attribute-based encryption schemes.

Fully hidden policy ABE scheme [78]: Hides the entire access structure and that includes both the attribute name and its value. This has a disadvantage of making the decryption very difficult since the user cannot know the attribute set that satisfies the access structure [133].

Half or partially hidden policy ABE scheme [123], [130], [134], [135]: Hides the attribute values leaving the attribute names clear. This allows the decryptor to easily identify the attribute set to be used to satisfy the access structure. Nishide *et al.* [123] proposed an ABE scheme with partially hidden access structure and the construction was based on the "AND" gate access structure. In this work, wildcards are used to represent the attributes whose values do not matter in the access policies. For example, consider an access structure $\mathbb{A} = (W_1, \ldots, W_n)$, where, n is the number of attributes in the policy. For a value of $n = 5$, this access structure can be represented as $(1, 1, *, *, 0)$, indicating that the attribute values for the third and fourth attributes do not matter during the decryption process. This can be seen as an "AND-gate" on all the attributes in the policy. The main objective of the study [123] is to enable decryption of ciphertexts without seeing the value of attributes in the ciphertexts (i.e., without knowing whether W_i is 1 or 0 or $*$) and is done through, setup, keygeneration, encryption, and decryption algorithms, and proceeds as follows.

1) **Setup**$(\kappa \rightarrow mk, pp)$: Generates the public parameters and master key. Two bilinear groups $\mathbb{G}$ and $\mathbb{G}_T$ of order p are chosen. Let g be a generator of $\mathbb{G}$ and $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be a bilinear map. Choose $w \in \mathbb{Z}_p^*$ randomly. Compute $Y = (g, g)^w$. The generated public parameter pp as $(Y, p, \mathbb{G}, \mathbb{G}_T, g, e, \{\{A_{i,t}^{a_{i,t}}, A_{i,t}^{b_{i,t}}\}_{1 \leq t \leq n_i}\}_{1 \leq i \leq n})$, and the master key mk is $(w, \{\{a_{i,t}, b_{i,t}\}_{1 \leq t \leq n_i}\}_{1 \leq i \leq n})$, where, $\{a_{i,t}, b_{i,t} \in Z_p^*\}_{1 \leq t \leq n_i}$ is a value of a randomly chosen attribute and $\{A_{i,t} \in \mathbb{G}\}_{1 \leq t \leq n_i}$ is a randomly chosen point, for every value t of attribute i, and $1 \leq i \leq n$.
2) **KeyGeneration**$(mk, L \rightarrow sk_L)$: Generates the secret key sk_L for a user as $(D_0, \{\{D_{i,j}\}_{0 \leq j \leq 2}\}_{1 \leq i \leq n})$ by taking the master key mk and list of user attributes $L = [L_1, \ldots, L_n] = [v_{1,t_1}, \ldots, v_{i,t_i}]$ (v_{i,t_i} is the value of an attribute), where for $1 \leq i \leq n$, $(D_{i,0}, D_{i,1}, D_{i,2}) = (g^{s_i}(A_{i,t_i})^{a_{i,t_i}b_{i,t_i}\lambda_i}, g^{a_{i,t_i}\lambda_i}, g^{b_{i,t_i}\lambda_i})$, $s_i, \lambda_i \in \mathbb{Z}_p^*$ and $s = \sum_{i=1}^{n}$. $D_0 = g^{w-s}$.
3) **Encryption**$(pp, M, \mathbb{A} \rightarrow CT)$: This encrypts the data M under the access structure $\mathbb{A} = (W_1, \ldots, W_n) = W_i$ to produce a ciphertext CT as $(\tilde{C}, C_0, \{\{C_{i,t,1}, C_{i,t,2}\}1 \leq t \leq n_i\}1 \leq i \leq n)$, where $\tilde{C} = MY^r$, $C_0 = g^r$, $r \in \mathbb{Z}_p$, and for every attribute i, $1 \leq i \leq n$, a random value $\{r_{i,t} \in \mathbb{Z}_p^*\}_{1 \leq t \leq i}$ is picked to compute $\{C_{i,t,1}, C_{i,t,2}\}1 \leq t \leq n_i$. $(C_{i,t,1}, C_{i,t,2})$'s computation is based on whether $v_{i,t}$ is in set W_i or not. If $v_{i,t} \in W_i$, $(C_{i,t,1}, C_{i,t,2}) = ((A_{i,t}^{b_{i,t}})^{r_{i,t}}, (A_{i,t}^{a_{i,t}})^{r-r_{i,t}})$. Otherwise $(C_{i,t,1}, C_{i,t,2})$ are considered randomly.

4) **Decryption**($CT, sk_L \rightarrow M/\perp$): Regenerates M by taking CT and sk_L as inputs. It recovers M if the access structure is satisfied by computing $\frac{\tilde{C}Pi_{i=1}^{n} e(C_{i,t_i,1}, D_{i,1}) e(C_{i,t_i,2}, D_{i,2})}{e(C_0, D_0) \Pi_{i=1}^{n} e(C_0, D_{i,0})} = M$, where $L_i = v_{i,t_i}$. Otherwise, it returns $\perp$.

A revocation functionality is added to the half-hidden ABE scheme by Xu and Joshi [136]. In their scheme, the revocation task is delegated to the cloud server. The cloud server re-encrypts the ciphertext received from the data owner and in the process removes the access rights for the revoked users. They added the access pattern privacy preservation technique based on oblivious RAM [137] to their work. The pattern privacy preservation technique helps to prevent adversaries from analyzing the access patterns and learning from them [138] which further strengthens the privacy provision capability of their scheme.

V. CHALLENGES ASSOCIATED WITH ABE USAGE IN COLLABORATIVE EHEALTH

Using attribute-based encryption in collaborative eHealth presents a number of challenges and some of which are discussed in this section.

As discussed earlier in section IV-B, collaborative eHealth has a diverse range of users which most likely comes with the usage of diverse kinds of computing devices including less powerful ones. The majority of ABE construction is based on pairing operations that are computationally demanding and thus inappropriate for less powerful computing devices. Share recovery operations in access structures like threshold-gate and LSSS also increase the number of exponentiations in the scheme, which equally increases computational demand. These present problems for data users in the system that have resource-scarce devices. Alternative approaches have been suggested to enable ABE schemes to be used in resource-scarce environments. Lin *et al.* [139] proposed a collaborative key management protocol to improve the efficiency and security of key management in CP-ABE schemes used during cloud data sharing involving mobile devices. Their scheme distributes components of the decryption key between the user, cloud server, and key authority. Park *et al.* [20] proposed schemes to improve key management in mobile devices accessing outsourced cloud-resources by employing a trusted server. Their schemes achieved backward security after revocation and improved efficiency in resource-constrained devices. In [140], Chen *et al.* proposed a cloud-based partial decryption multi-authority ABE scheme in which part of the decryption process is done by the cloud servers to reduce the decryption burden on mobile devices. Recently, Sedaghat *et al.* [141] suggested the integration of message integrity into CP-ABE by using signcryption which combines encryption and signature algorithms [142], [143]. They proposed a ciphertext policy attribute-based signcryption scheme to securely share smart grid data, and part of the decryption process is done by a third-party server. The security of this system relies on the security strength of the server.

The collaborative eHealth environment is dynamic. Data comes in an unregulated manner, and users are allowed and removed at any time. This demands mechanisms of revocation and access control modification from the adopted ABE schemes. As much as a lot has been done concerning attribute/user revocation in ABE schemes, little has focused on-demand access structure modification. The rudimentary access structure modification approach is for the data owner to first retrieve the ciphertext from the cloud, then modify the access policy and re-encrypt the data using the new access policy before uploading it back to the cloud. This is very inefficient due to incurred communication and computation overheads and tedious for the data owner. Authors in [111], [144], proposed schemes that outsource the policy update task to the cloud. Access policy update keys are generated and sent to the cloud for updates. However, the reliance on untrusted cloud servers to perform this update is a drawback of these schemes.

Data integrity challenges during delegations. In [145]–[149] delegation and proxy re-encryption techniques are investigated for access policies to be updated and extended, i.e., the data originally encrypted for Bob is transformed to be accessed by Alice with the help of proxies. The drawback with these schemes is that there is no provision for data integrity checks. Susilo *et al.* [150] proposed a scheme with an additional feature for data integrity check. In their scheme, a legitimate user (Bob) first decrypts the data before re-encrypting it with a new policy for the originally illegitimate user (Alice). Bob uploads the data in the cloud and the cloud performs an integrity check on the data before making it available to Alice. However, this implies an increase in computational demand on the user side. Remember, some users' gadgets are resource-scarce.

Much as a lot of research has been done to improve the attribute-based schemes for privacy provision in the cloud environment, little focus has been on addressing the challenges posed by side-channel attacks. In a side-channel attack, the adversaries exploit the implementation of cryptosystems to learn secret information. Reference [151]–[154] have proposed schemes to address the side-channel attacks in attribute-based schemes. The main drawback of these schemes is their construction is based on composite order bilinear groups which are known to be computationally demanding. As a result, these schemes are far from being accepted for practical usage.

Storage overhead is another challenge to deal with when using ABE schemes in collaborative eHealth. Storage overhead is due to unimportant data that remains stored in the different entities especially after an update is performed. ABE schemes that completely remove the outdated key and ciphertext components are considered effective as compared to those that keep the outdated components.

VI. COMPARISONS

In this section, we perform a comparative analysis of the different ABE schemes for collaborative eHealth system. First,

TABLE 4. Public domain ABE-Based schemes.

Scheme	Revocation	Security	Access Structure
RNS [94]	Yes	Against N-1 AA Collusion	LSSS or Boolean Functions
LYZRL [18]	Yes	Against N-2 AA collusion	Conjunctive Normal Form
CC [83]	No	Against N-2 AA collusion	Threshold gate
QLT [121]	No	Against N-1 AA collusion	Hierarchical Attribute Tree [155]–[157]
Chase [82]	No	Against collusion using CA	LSSS
LCLS [84]	No	Against collusion, iff, $n_u \leq n_{AA}$	LSSS
XZWZ [98]	Yes	Against collusion using CA	Weighted Access Structure
YYYL [99]	Yes	Against collusion using CA	LSSS
YJ [102]	Yes	Against collusion using CA	LSSS
YJR [100]	Yes	Against collusion using CA	LSSS
YJ [101]	Yes	Against collusion using CA	LSSS
WQWT [88]	No	Against collusion using CA	LSSS
CWWPJ [140]	Yes	Against collusion using CA	LSSS
WLH [89]	Yes	Hash value of each user's GID is included in the secret key	LSSS
QLZ [96]	Yes	Anonymous key issuing to users	Threshold gate
FTWLY [97]	Yes	Against N-1 AA collusion	LSSS
HSMY [85]	No	Collusion attack possible	Threshold-gate

we divide the ABE schemes primarily into two, depending on whether they are suitable for either the private or public domain sectors of the collaborative eHealth system. Then, we compare the efficiency of the different schemes in terms of computation complexity and storage and communication overheads.

A. PUBLIC DOMAIN ABE SCHEMES

In the public domain of collaborative eHealth, the number of users is big and these users are most likely under the management of different authorities. Furthermore, these authorities are responsible for assigning attributes to the users. Therefore, it is important if the different authorities manage keys associated with users they directly manage. Thus, we consider multi-authority ABE (MA-ABE) schemes for privacy provision in the public domain.

1) COMPARISON OF PUBLIC DOMAIN ABE SCHEMES

A number of MA-ABE schemes have been selected from among those being surveyed for this comparison. The comparison is done in terms of their security provision and revocation capabilities, and the used access structure as shown in Table 4.

It can be observed that the presented MA-ABE schemes are to a greater extent all collusion resistant except HSMY [85], which is a superb ingredient for the much-needed confidentiality of EHRs in collaborative eHealth systems. The schemes RNS [94], WLH [89], QLZ [96], FTWLY [97], LYZRL [18], XZWZ [98], YJ [102], YJR [100], YJ [101] and CWWPJ [140] have incorporated revocation in their construction. This is important to remove obsolete and malicious users from the system, and there is no need to re-run the entire system set up in doing so. This further makes the scheme scalable, hence allowing the addition and removal of users in collaborative eHealth systems efficiently.

TABLE 5. Comparison of computation of the public domain ABE schemes.

d = the binary tree depth and n_{d_A} = the number of attributes involved in decryption.

Scheme	Encryption	Decryption	Revocation
YJR [100]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(1)$	$\mathcal{O}(n_{nu,r_A})$
LYZRL [18]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A})$
RNS [94]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{c,r} \times n_{nu,r_A})$
XZWZ [98]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A})$
YJ [101]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A})$
YYYL [99]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(1)$	$\mathcal{O}(n_{nu,r_A})$
YJ [102]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A})$
WLH [89]	$\mathcal{O}(n_{c_A} \times d)$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{c_A} \times d)$
QLZ [96]	$\mathcal{O}(N_{AA} + n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A})$
FTWLY [97]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(1)$	$\mathcal{O}(n_{nu,r_A})$

2) EFFICIENCY ANALYSIS OF THE PUBLIC DOMAIN ABE SCHEMES

The ABE scheme used in collaborative eHealth needs to be efficient as stated in the requirement section. This has motivated us to perform efficiency comparisons for public domain ABE schemes. Due to revocation being an important aspect of a security scheme for collaborative eHealth systems, we focus on the schemes with revocation functionalities in conducting this comparative efficiency analysis. The comparison is done in terms of computation complexity and storage and communication overheads.

COMPUTATION COMPLEXITY COMPARISON OF PUBLIC DOMAIN ABE SCHEMES

Let n_u be the total number of users, n_o the number of owners, N_{AA} the number of attribute authorities, n_{c_A} the number of attributes in the ciphertext, n_{nu,r_A} the number of non-revoked users with revoked attributes, n_{u_A} the number of user attributes and $n_{c,r}$ be the number of ciphertexts with revoked attributes. We compare the computation of the different public domain ABE schemes in terms of their computation complexity as illustrated in Table 5.

Each ciphertext is associated with a set of attributes that are used to authorize a decryptor before the actual decryption process. Therefore, the more the number of attributes, the more the number of associations that need to be performed. As a result of this, it can be observed that encryption complexity increases with the number of attributes in the ciphertext. For WLH [89], an additional component associated with time period is included in the ciphertext for each attribute, and for QLZ [96], a combined public key component of all the authorities is included in the ciphertext. These result in the additional components contributing to the increase in encryption computation complexity. For the decryption, the decryption computation complexity increases with the number of user attributes except for the YJR [100], FTWLY [97], and YYYL [99] schemes. For the YJR [100],

TABLE 6. Comparison of storage overhead for public domain ABE-Based schemes.

$|A|_u$ = the total number of user attributes and $|\theta|$ = the number binary tree nodes

Entity	YJR [100]	LYZRL [18]	RNS [94]	XZWZ [98]	YJ [101]	YJ [102]
AA_k	$(n_{aa,k}+3)\|p\|$	$(N_{AA}+1+n_{aa,k})\|p\|$	$(2n_{aa,k})\|p\|$	$(n_{aa,k}+N_{AA})\|p\|$	$(n_{aa,k}+2n_u+3)\|p\|$	$(n_{aa,k}+2n_o+1)\|p\|$
Owner	$(3N_{AA}+1+\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|$	$(2N_{AA}+2+\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|$	$(n_{cs}+2\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|$	$(1+4N_{AA}+\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|$	$(3N_{AA}+1+4\sum_{k=1}^{N_{AA}} n_{aa,k}\|p\|$	$(2+\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|$
User	$(3N_{AA}+1+\sum_{k=1}^{N_{AA}} n_{aa,u,k})\|p\|$	$(N_{AA}(1+N_{AA})+\sum_{k=1}^{N_{AA}} n_{aa,u,k})\|p\|$	$(n_{c,r}+\sum_{k=1}^{N_{AA}} n_{aa,u,k})\|p\|$	$(1+\sum_{k=1}^{N_{AA}} n_{aa,u,k})\|p\|$	$(3N_{AA}+1+\sum_{k=1}^{N_{AA}} n_{aa,u,k}\|p\|$	$(n_o(1+\sum_{k=1}^{N_{AA}} n_{aa,u,k}))\|p\|$
Server	$(3n_{c_A}+3)\|p\|$	$(4+n_{c_A})\|p\|$	$(3n_{c_A}+1)\|p\|$	$(4+n_{c_A})\|p\|$	$(2n_{c_A}+3)\|p\|$	$(2+n_{c_A})\|p\|$

Entity	YYYL [99]	WLH [89]	QLZ [96]	FTWLY [97]
AA_k	$(n_o+n_u+n_{aa,k})\|p\|$	$2+\|\theta\|$	$(n_{aa,k}+N_{AA}+1)\|p\|$	$(2+2n_{aa,k})\|p\|$
Owner	$(4+4\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|$	$2+\|\theta\|+\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|$	$(N_{AA}+1+\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|)$	$(2+2\sum_{k=1}^{N_{AA}} n_{aa,k})\|p\|)$
User	$(2+\sum_{k=1}^{N_{AA}} n_{aa,u,k})\|p\|$	$2+2(1+\|\theta\|)(\sum_{k=1}^{N_{AA}} n_{aa,u,k})\|p\|$	$(2+\sum_{k=1}^{N_{AA}} n_{aa,u,k})\|p\|$	$(3+\sum_{k=1}^{N_{AA}} n_{aa,u,k})\|p\|$
Server	$(2+4n_{c_A})\|p\|$	N/A	$(3+n_{c_A})\|p\|$	$(5+n_{c_A}+\|A\|_u)\|p\|$

FTWLY [97] and YYYL [99] schemes, a decryption token generated by a server minimizes the decryption computation burden on the user and hence the constant decryption complexity. The revocation complexity mostly depends on the number of unrevoked users having revoked attributes. Computations must be performed to update the secret keys of these categories of users. For the RNS [94] scheme, the revocation complexity is high because all the ciphertexts affected by the revoked attributes are replaced. Also, for WLH [89], the additional increase in revocation in the revocation complexity is due to the modification of the time component included for each attribute in the ciphertext.

STORAGE OVERHEAD COMPARISON FOR PUBLIC DOMAIN ABE SCHEMES

Let $|p|$ denote the size of the elements in $\mathbb{Z}_p$, groups $\mathbb{G}$ and $\mathbb{G}_T$, N_{AA} the number of attribute authorities, n_u the total number of users, n_o the number of owners, $n_{aa,k}$ the number of attributes managed by authority AA_k, $n_{aa,u,k}$ the number of attributes assigned to a user u by AA_k, n_{cs} the number of ciphertexts in the cloud servers, and $n_{c,r}$ the number of ciphertexts with revoked attributes. We compare the storage overhead of the public domain schemes as illustrated in Table 6.

The storage overhead in an attribute authority AA_k in these schemes is normally due to the version number of attributes the authority manages and the authority's secret key. In this case, YJR [100] has the least storage overhead in the attribute authority (AA_k). For YJR [100], all the attributes managed by the attribute authority have a version number that is equivalent to the number of attributes managed by the authority, $n_{aa,k}$. The additional storage can also be attributed to the secret key of the authority, $3|p|$. Meanwhile, for instance, YJ [102]'s overhead is due to the version number of all the attributes the authority manages, the private key received from all the owners (n_o), and the identity received from the central authority (CA). For WLH [89], the authority does not keep information associated with attributes but it instead keeps information associated with system time periods represented inform of the binary tree.

The owner's storage overhead is due to the public attribute keys received from the attribute authorities and secret and master keys. The YJ [102] scheme incurs the least storage overhead on the owners' side, as it has only the public attribute keys and $2|p|$ secret components. Unlike the YJ [102] scheme, the RNS [94] scheme requires holding the encryption secret for all the ciphertexts, which results in a heavier storage overhead.

The user storage overhead is because of the received secret keys received from the attribute authorities and the global secret keys. XZWZ [98] incurs the least user storage overhead, as it only stores secret keys received from attribute authorities. Meanwhile, apart from the received keys from the attribute authorities, YJ [102] also stores secret keys from all data owners, and RNS [94] stores components of ciphertexts with revoked attributes, as re-encryption of ciphertexts with revoked attributes is part of the re-encrypted ciphertext associated with the revoked attribute to be sent to the non-revoked users.

The main contributors to the server storage overhead are the generated ciphertexts. Some schemes are used for encrypting only the symmetric keys used to encrypt the main data, and in this case, we only consider the ciphertext components encrypted by these schemes.

COMPARISON OF COMMUNICATION OVERHEAD FOR PUBLIC DOMAIN ABE SCHEMES

As shown in Table 7, in this case, we only compare the communication overhead as a result of the revocation functionality in these schemes. It can be observed that the communication overhead due to revocation is linear with the number of ciphertexts with revoked attributes for RNS [94], XZWZ [98], FTWLY [97], and LYZRL [18]. It becomes larger owing to a higher number of ciphertexts in the cloud. The communication overhead is least incurred by YJR [100] as only the non-revoked user private attribute keys are updated.

B. PRIVATE DOMAIN ABE SCHEMES

Unlike the users of the public domain, the users of the private domain are considered to be fewer and have some level of

TABLE 7. Comparison of communication overhead for public domain ABE-Based schemes due to revocation.

$|\theta|_{update}$ = the number of updated binary tree nodes and $|\zeta|$ = the number of nodes from the intersection between $path(u)$ and $minCover$

Operation	YJR [100]	LYZRL [18]	RNS [94]	XZWZ [98]	YJ [101]	YJ [102]	YYYL [99]
Key Update	$n_{nu,r_A}\|p\|$	$n_{nu,r_A}\|p\|$	N/A	$n_{nu,r_A}\|p\|$	$n_{nu,r_A}\|p\|$	$((1 + n_{r_A}) \times n_{nu,r_A})\|p\|$	$(1 + n_{r_A} \times n_{nu,r_A})\|p\|$
Ciphertext Update	$\|p\|$	$n_{c,r}\|p\|$	$(n_{c,r} \times n_{nu,r_A})\|p\|$	$n_{c,r}\|p\|$	$2\|p\|$	$2\|p\|$	$n_{c,r}\|p\|$

Operation	WLH [89]	QLZ [96]	FTWLY [97]
Key Update	$2\|\theta\|_{update}\|p\|$	$n_u(2 + n_{u_A})\|p\|$	$n_{r_A} \times n_u\|p\|$
Ciphertext Update	$1 + n_{c_A}(4 + \|\zeta\|)$	$n_u(3 + n_{c_A})\|p\|$	$n_{r_A} \times n_{c,r}\|p\|$

personal relationship with the EHR owner. Hence, EHR privacy protection during EHR accessibility by these users can be achieved using the single authority ABE schemes directly under the control of the EHR owner.

1) COMPARISON OF PRIVATE DOMAIN ABE SCHEMES

In a similar manner, we compare the different private domain ABE schemes. Several single authority ABE schemes are selected for comparison in terms of collusion resistance and revocation capabilities, access structure, and their construction mechanism as illustrated in Table 8. It can be observed that the majority of the schemes are proven to be collusion resistant except HN [37] and ILB [103], which is critical in addressing the confidentiality requirement of the collaborative eHealth. The constructions of all the schemes with the exception of YCT [71] are based on bilinear pairing, which is cumbersome for resource-constrained devices that are most likely deployed in smart environments to send patients' data to the cloud for storage. The schemes SZ [17], LYHZS [109], LYZQH [110], ILB [103], XJ [136], JMB [108], IPNHJ [107], HN [37], YWRL [95] and LLLS [90] have an integrated revocation functionality in their construction that is a very vital feature for an ABE scheme to be used in collaborative eHealth. This helps in achieving scalability in the collaborative eHealth without re-running the setup phase. Furthermore, this helps in the removal of obsolete or malicious users from the system.

2) EFFICIENCY ANALYSIS OF THE PRIVATE DOMAIN ABE SCHEMES

As stated in the previous sections, the efficiency of the ABE schemes is a key feature to be considered in using such a scheme in collaborative eHealth. As a result, in this section, we compare the efficiency of several private ABE schemes in terms of communication and storage overheads, and computation complexity. We only focused on schemes that have incorporated revocation functionalities.

a: COMPUTATION COMPLEXITY COMPARISON FOR PRIVATE DOMAIN ABE SCHEMES

Let n_{c_A} be the number of attributes in the ciphertext, n_u be the total number of users, n_o the number of owners, n_{nu,r_A} be the number of non-revoked users with revoked attributes, n_{u_A} be the number of user attributes, $n_{c,r}$ be the number of ciphertexts with revoked attributes, and n_{cs} be the number of ciphertexts. The comparison of the computation complexity for private domain ABE schemes of collaborative eHealth is illustrated in Table 9.

TABLE 8. Private domain ABE-Based schemes.

Sheme	Revocation	KP/CP-ABE	Security	Access Structure	Construction
GPSW [12]	No	KP	Against Collusion Attack	Threshold gate	Prime Order Bilinear Group
JMB [108]	Yes	CP	Against Collusion Attack	LSSS	Prime Order Bilinear Group
BSW [73]	No	CP	Against Collusion Attack	Threshold gate	Prime Order Bilinear Group
IPNHJ [107]	Yes	CP	Against Collusion Attack	Threshold gate	Prime Order Bilinear Group
Waters [77]	No	CP	Against Collusion Attack	LSSS	Prime Order Bilinear Group
LOSTW [78]	No	CP	Against Collusion Attack	LSSS	Composite Order Bilinear Group
HN [37]	Yes	CP	Collusion Attack Possible	Threshold gate	Prime Order Bilinear Group
CN [79]	No	CP	Against CPA and CCA	AND Gates	Prime Order Bilinear Group
LGCXLQ [58]	No	CP	Against Collusion Attack	OBDD	Prime Order Bilinear Group
AAHXC [75]	No	CP	Against Collusion Attack	Threshold gate	Prime Order Bilinear Group
YWRL [95]	Yes	CP	Against Collusion Attack	AND Gate	Prime Order Bilinear Group
YCT [71]	No	KP	Against Collusion Attack	Threshold gate	Elliptic Curve Cryptography
SZ [17]	Yes	CP	Against Collusion Attack	Threshold gate	Prime Order Bilinear Group
LLLS [90]	Yes	CP	Against Collusion Attack	LSSS	Prime Order Bilinear Group
LYHZS [109]	Yes	CP	Collusion Resistant	Threshold gate	Prime Order Bilinear Group
LYZQH [110]	Yes	CP	Collusion Resistant	Threshold gate	Prime Order Bilinear Group
ILB [103]	Yes	CP	Collusion Attack Possible	Threshold gate	Prime Order Bilinear Group
XJ [136]	Yes	CP	Collusion Resistant	LSSS	Composite Order Bilinear Group
LYZ [151]	No	CP	Collusion Resistant	LSSS	Composite Order Bilinear Group
ZZM [153]	No	CP/KP	Collusion Resistant	LSSS	Composite Order Bilinear Group
ZZ [154]	No	CP	Collusion Resistant	LSSS	Composite Order Bilinear Group

TABLE 9. Comparison of computation of the private domain ABE schemes.

n_{d_A} = the number of attributes involved in the decryption

Scheme	Encryption	Decryption	Revocation
HN [37]	$\mathcal{O}(n_{C_A}+\log n_u)$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A}\log\frac{n_u}{n_{nu,r_A}})$
JMB [108]	$\mathcal{O}(n_{C_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_u+n_{cs})$
IPNHJ [107]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(1)$
YWRL [95]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A})$
SZ [17]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A})$
LLLS [90]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A})$
LYHZS [109]	$\mathcal{O}(n_{c_A}+\log n_u)$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{nu,r_A}\log\frac{n_u}{n_{nu,r_A}})$
LYZQH [110]	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(n_{nu,r_A})$
ILB [103]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(2n_{d_A})$	$\mathcal{O}(n_{r_A}+n_{nu,r_A})$
XJ [136]	$\mathcal{O}(n_{c_A})$	$\mathcal{O}(n_{d_A})$	$\mathcal{O}(n_{r_A})$

From Table 9 it can be observed that the HN [37] and LYHZS [109] schemes incur the most encryption complexity because of the ciphertext re-encryption during outsourcing. The LYZQH [110] scheme has a constant computation demand due to encryption outsourcing. The rest of the schemes' encryption computation complexity increases with an increase in the number of ciphertext attributes, as a result of more ciphertext-attribute associations getting created. The decryption computation complexity for all the schemes increases with an increase in the number of user attributes except for LYZQH [110] which is constant. The higher revocation complexity for HN [37] and LYHZS [109] is due to updating the group keys of the non-revoked users, and for the JMB [108] scheme, it is due to the modification of the access structure for all the users and also adding an extra component to all the ciphertexts to deter the revoked users from decrypting them. The constant revocation complexity for IPNHJ [107] is due to revocations being completely controlled by a proxy and the proxy simply refusing to perform part of its decryption computation for the revoked user. For the rest of the schemes, the revocation complexity increases with an increase in the number of non-revoked users having revoked attributes.

b: STORAGE OVERHEAD COMPARISON FOR PRIVATE DOMAIN ABE SCHEMES

Let $|p|$ denote the size of the elements in $\mathbb{Z}_p$, groups $\mathbb{G}$ and $\mathbb{G}_T$, n_o be the number of owners, n_{u_A} be the number of attributes belonging to a user, n_{cs} be the number of ciphertexts, n_{c_A} denote the number of ciphertext attributes, $n_{c,r}$ be the number of ciphertexts with revoked attributes, and n_u be the number of users in the system.

The storage overhead of the private domain ABE schemes is mainly due to the attribute versions, secret keys, and stored ciphertext, and is shown in Table 10. It can be observed that the XJ [136] scheme incurs the most storage overhead for the owner. This comes from storing the access structure and multiple ciphertext components that have multiple attribute associations. The least storage overhead for the owner is incurred by the LYZQH [110] scheme, whose encryption is outsourced and there is no need for the owner to store attribute association components of the ciphertext. The heaviest storage overhead in the user is incurred by LLLS [90] as revocation paths are included in the attribute keys resulting in having numerous key components having attribute associations and the least storage overhead, in this case, is incurred by the IPNHJ [107] scheme. At the server, the most storage overhead is incurred by the XJ [136] scheme owing to storing extra ciphertext component associated with ciphertext attributes and JMB [108] incurs the least storage overhead at the server.

c: COMMUNICATION OVERHEAD COMPARISON FOR PRIVATE DOMAIN ABE SCHEMES

Table 11 illustrates the communication overhead incurred by the private domain ABE schemes as a result of the incorporated revocation functionality. It can be observed that the communication overhead is linear with the number of ciphertexts and the number of non-revoked users having revoked attributes. LYHZS [109] incurs the most communication overhead as it has to re-encrypt all the affected ciphertexts and perform key updates for all the non-revoked users with revoked attributes.

VII. FUTURE DIRECTIONS

Much as a lot of studies have been done on ABE schemes, further investigations are still required for its effective usage in collaborative eHealth. We present some areas that require investigations as follows.

- *Attribute based encryption schemes for mobile and IoT devices*: The pairing based attribute-based encryption schemes are cumbersome for mobile and IoT devices. Yet, these devices are widely being used to access health data from the cloud or gather health data for storage in the cloud. Thus, the elliptic curve-based ABE schemes would be a lighter alternative for mobile and IoT devices. However, a number of the following investigations need to be performed.
  1) Construction of a revocable, traceable, expressive and efficient ciphertext policy attribute-based encryption scheme based on an elliptic curve can be of great importance for fine-grained access to cloud-based health data by mobile users.
  2) A multi-authority attribute based encryption scheme relying on elliptic curve operations can also be looked at.
- *The ordered binary decision diagram access structure (OBDD)*: OBDD is one of the suggested non-monotonic access structures used for expressive ABE scheme constructions. However, several open areas that require further investigation regarding the OBDD access structure still exist.
  1) Attribute revocation, access structure modification, and traceability are some open areas yet to be studied regarding the OBDD access structure usage in ABE schemes.
  2) Access policy hiding in regards to OBDD access structure appears to be another interesting area to look at.

TABLE 10. Comparison of storage overhead for private domain ABE-Based schemes.

Entity	HN [37]	JMB [108]	IPNHJ [107]	YWRL [95]	LLLS [90]	SZ [17]
Owner	$(4 + 2n_{c_A})\|p\|$	$(5 + 2n_{c_A})\|p\|$	$(4 + n_{c_A})\|p\|$	$(4 + n_{c_A})\|p\|$	$(5 + n_{c_A})\|p\|$	$(4 + 2n_{c_A})\|p\|$
User	$(2 + 2n_{u_A})\|p\|$	$(3 + 2n_{u_A})\|p\|$	$(2 + n_{u_A})\|p\|$	$(3 + 2n_{u_A})\|p\|$	$(4 + 5n_{u_A})\|p\|$	$(2 + 2n_{u_A})\|p\|$
Server	$(4+3n_{c_A}+n_{cs})\|p\|$	$(n_{c_A})\|p\|$	$(1 + \sum_{u=1}^{n_u} n_{u_A})\|p\|$	$(5 + 2n_{c_A} + n_{cs})\|p\|$	N/A	N/A

Entity	LYHZS [109]	LYZQH [110]	ILB [103]	XJ [137]
Owner	$(4 + 2n_{c_A})\|p\|$	$12\|p\|$	$(4 + 2n_{c_A})\|p\|$	$(7 + 4n_{c_A})\|p\|$
User	$(2 + 4n_{u_A})\|p\|$	$(7 + 2n_{u_A})\|p\|$	$(2 + 3n_{u_A})\|p\|$	$(3 + n_{u_A})\|p\|$
Server	$(4+4n_{c_A}+n_{cs})\|p\|$	$(8+2n_{c_A}+n_{cs})\|p\|$	$(3+3n_{c_A}+n_{cs})\|p\|$	$(8+4n_{c_A}+n_{cs})\|p\|$

TABLE 11. Comparison of communication overhead for private domain ABE-Based schemes due to revocation.

Entity	HN [37]	JMB [108]	IPNHJ [107]	YWRL [95]	LLLS [90]	SZ [17]
Key Update	$(n_{nu,r_A})\|p\|$	N/A	N/A	$(n_{nu,r_A})\|p\|$	$(2n_{nu,r_A})\|p\|$	N/A
Ciphertext Update	$(6 + 2n_{c,r})\|p\|$	$\|p\|$	N/A	$(4 + n_{c,r})\|p\|$	N/A	$\|p\|$

Entity	LYHZS [109]	LYZQH [110]	ILB [103]	XJ [137]
Key Update	$(n_{nu,r_A})\|p\|$	$(3 \times n_{nu,r_A}\|p\|$	$(n_{nu,r_A})\|p\|$	N/A
Ciphertext Update	$(8 + 4n_{c,r})\|p\|$	$\|p\|$	$\|p\|$	$\|p\|$

- Further attention is needed for constructing expressive and efficient ABE schemes with constant ciphertext sizes. This can be beneficial for environments with resource-constrained devices.
- The majority of ABE schemes do not pay attention to the used file formats. Consider, for example, an XML file being used to store all the EHRs of a patient. Investigating fine-grained access to the different fields/parts of the file using ABE can be an interesting research idea.
- *Side-channel attack resilient attribute-based encryption schemes:* Little attention to this far has been paid to design attribute-based schemes that are resilient to side-channel attacks. As a result, further efforts to design efficient and usable attribute-based schemes that are resilient against side-channel attacks are worth pursuing. Below are some gaps that require more investigation.
  1) Prime order bilinear groups have been proven to out-perform composite order bilinear groups in terms of computational efficiency. As a result, constructing a side-channel resilient ABE scheme based on prime order bilinear group can be interesting for future consideration.
  2) Revocation and access structure modification require further investigation for leakage resilient attribute-based schemes.
  3) Similarly, designing a multi-authority attribute based encryption scheme that is resilient against leakages can be an interesting investigation.
- *An efficient attribute-based encryption scheme with data integrity check:* Data integrity checks have been sidelined during delegations and access structure modifications. The integrity checks in the existing works put a lot of burden on the delegating user. Therefore, minimizing the workload on delegating users by outsourcing the delegation task to a proxy while still performing data integrity checks can be looked at.

VIII. CONCLUSION

Privacy of EHRs is considered to be central to the success of collaborative eHealth. Owing to its reliance on the cloud for EHR storage, ABE which offers fine-grained access control is one of the best candidates for privacy provision in collaborative eHealth. In this work, we presented an overview of the collaborative eHealth, highlighted the likely privacy challenges faced in such an environment and the approaches to overcome the challenges. We then surveyed and discussed the different categories of ABE schemes for privacy provision in the collaborative eHealth. We performed comparisons of the different ABE schemes for the two user domains in collaborative eHealth. We analyzed single authority ABE schemes for private domain users while multi-authority ABE schemes for public domain users. Also, considering the collaborative eHealth requirements, we identified ABE schemes with revocation functionalities as the most suitable candidates for collaborative eHealth, as they enable removal of unwanted users from the systems without re-running the entire system from the setup phase. We then analyzed the efficiency of ABE schemes with revocation functionalities in terms of computation complexity and storage and communication overheads for both the public and private domains of collaborative eHealth. Finally, we highlighted the challenges associated with ABE and areas that require further investigations for ABE usage in the collaborative eHealth.

REFERENCES

[1] L. Coppolino, S. D'Antonio, L. Romano, L. Sgaglione, and M. Staffa, "Addressing security issues in the eheatlh domain relying on siem solutions," in *Proc. IEEE 41st Annu. Comput. Softw. Appl. Conf.*, Jul. 2017, pp. 510–515.

[2] R. Sánchez-Guerrero, F. A. Mendoza, D. Díaz-Sánchez, P. A. Cabarcos, and A. M. López, "Collaborative ehealth meets security: Privacy-enhancing patient profile management," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 6, pp. 1741–1749, Nov. 2017.

[3] O. Kocabas, T. Soyata, and M. K. Aktas, "Emerging security mechanisms for medical cyber physical systems," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 13, no. 3, pp. 401–416, May/Jun. 2016.

[4] S. Aldossary and W. Allen, "Data security, privacy, availability and integrity in cloud computing: Issues and current solutions," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 4, pp. 485–498, 2016.
[5] T. Sahama, L. Simpson, and B. Lane, "Security and Privacy in eHealth: Is it possible?" in *Proc. IEEE 15th Int. Conf. e-Health Netw., Appl. Services (Healthcom)*, Oct. 2013, pp. 249–253.
[6] N. F. Pub, "Advanced encryption standard (AES)," *Federal Inf. Process. Standards Publication*, vol. 197, no. 441, p. 311, 2001.
[7] V. S. Miller, "Use of elliptic curves in cryptography," in *Advances in Cryptology—CRYPTO*, H. C. Williams, Ed. Berlin, Germany: Springer, 1986, pp. 417–426.
[8] N. Koblitz, "Elliptic curve cryptosystems," *Math. Comput.*, vol. 48, no. 177, pp. 203–209, 1987.
[9] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. 22, no. 6, pp. 644–654, Nov. 1976.
[10] D. Hankerson, A. J. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*. New York, NY, USA: Springer, 2006.
[11] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Proc. EUROCRYPT*, 2005, pp. 457–473.
[12] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. 13th ACM Conf. Comput. Commun. Secur.*, 2006, pp. 89–98.
[13] O. Kocabas and T. Soyata, "Towards privacy-preserving medical cloud computing using homomorphic encryption," in *Enabling Real-Time Mobile Cloud Computing Through Emerging Technologies*, T. Soyata, Ed. Hershey, PA, USA: IGI Global, 2015, ch. 7, pp. 213–246.
[14] P. Martins, L. Sousa, and A. Mariano, "A survey on fully homomorphic encryption: An engineering perspective," *ACM Comput. Surv.*, vol. 50, no. 6, p. 33, Dec. 2017.
[15] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Comput. Surv.*, vol. 51, no. 4, p. 79, Jul. 2018.
[16] I. A. Sukhodolskiy and S. V. Zapechnikov, "An access control model for cloud storage using attribute-based encryption," in *Proc. IEEE Conf. Russian Young Researchers Elect. Electron. Eng. (EIConRus)*, St. Petersburg, Russia, Feb. 2017, pp. 578–581.
[17] R. S. da Silva and S. D. Zorzo, "An access control mechanism to ensure privacy in named data networking using attribute-based encryption with immediate revocation of privileges," in *Proc. 12th Annu. IEEE Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2015, pp. 128–133.
[18] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 1, pp. 131–143, Jan. 2013.
[19] S. Pérez, J. L. Hernández-Ramos, D. Pedone, D. Rotondi, L. Straniero, and A. F. Skarmeta, "A digital envelope approach using attribute-based encryption for secure data exchange in IoT scenarios," in *Proc. Global Internet Things Summit (GIoTS)*, Geneva, Switzerland, 2017, pp. 1–6.
[20] J. Park, E. Kim, S. Park, and C. Kang, "Advanced attribute-based key management for mobile devices in hybrid clouds," in *Proc. 10th Int. Conf. Availability, Rel. Secur.*, Aug. 2015, pp. 566–575.
[21] T. Jung, X.-Y. Li, Z. Wan, and M. Wan, "Privacy preserving cloud data access with multi-authorities," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2625–2633.
[22] Z. Wang, D. Huang, Y. Zhu, B. Li, and C.-J. Chung, "Efficient attribute-based comparable data access control," *IEEE Trans. Comput.*, vol. 64, no. 12, pp. 3430–3443, Dec. 2015.
[23] Z. Guan, J. Li, Y. Zhang, R. Xu, Z. Wang, and T. Yang, "An efficient traceable access control scheme with reliable key delegation in mobile cloud computing," *EURASIP J. Wireless Commun. Netw.*, vol. 2016, no. 1, p. 208, 2016. doi: 10.1186/s13638-016-0705-2.
[24] K. Xue, Y. Xue, J. Hong, W. Li, H. Yue, D. S. L. Wei, and P. Hong, "RAAC: Robust and auditable access control with multiple attribute authorities for public cloud storage," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 953–967, Apr. 2017.
[25] M. Munsyi, A. Sudarsono, and M. U. H. A. Rasyid, "An implementation of data exchange using authenticated attribute-based encryption for environmental monitoring," in *Proc. Int. Electron. Symp. Knowl. Creation Intell. Comput. (IES-KCIC)*, Surabaya, Indonesia, 2017.
[26] T. V. X. Phuong, G. Yang, and W. Susilo, "Hidden ciphertext policy attribute-based encryption under standard assumptions," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 1, pp. 35–45, Jun. 2016.
[27] P. Akhila, M. H. L. Bhavani, and R. A. Canessane, "Role check: Protecting the user information using attribute-based encryption, dynamic key generation & user retraction system," in *Proc. 2nd Int. Conf. Sci. Technol. Eng. Manage. (ICONSTEM)*, Mar. 2016, pp. 125–129.
[28] S. Huda, A. Sudarsono, and T. Harsono, "Secure data exchange using authenticated ciphertext-policy attributed-based encryption," in *Proc. Int. Electron. Symp. (IES)*, Sep. 2015, pp. 134–139.
[29] Y. Rao, "A secure and efficient ciphertext-policy attribute-based signcryption for personal health records sharing in cloud computing," *Future Gener. Comput. Syst.*, vol. 67, pp. 133–151, Feb. 2017.
[30] I. Shree, K. Narmatha, and C. V. Joe, "An multi-authority attribute based encryption for personal health record in cloud computing," in *Proc. 10th Int. Conf. Intell. Syst. Control (ISCO)*, Coimbatore, India, Jan. 2016, pp. 1–5.
[31] A. Michalas and N. Weingarten, "Healthshare: Using attribute-based encryption for secure data sharing between multiple clouds," in *Proc. IEEE 30th Int. Symp. Comput.-Based Med. Syst. (CBMS)*, Jun. 2017, pp. 811–815.
[32] S. Maheswari and U. Gudla, "Secure sharing of personal health records in Jelastic cloud by attribute based encryption," in *Proc. 4th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS)*, Coimbatore, India, Jan. 2017, pp. 1–4.
[33] R. S. Sharon and R. J. Manoj, "E-health care data sharing into the cloud based on deduplication and file hierarchical encryption," in *Proc. Int. Conf. Inf. Commun. Embedded Syst. (ICICES)*, Feb. 2017, pp. 1–6.
[34] A. Samydurai, K. Revathi, P. Prema, D. S. Arulmozhiarasi, J. Jency, and S. Hemapriya, "Secured health care information exchange on cloud using attribute based encryption," in *Proc. 3rd Int. Conf. Signal Process., Commun. Netw. (ICSCN)*, Mar. 2015, pp. 1–5.
[35] D. D. Stalin and A. Jeyachandran, "A comprehensive survey of security mechanisms in healthcare applications," in *Proc. Int. Conf. Commun. Electron. Syst. (ICCES)*, Oct. 2016, pp. 1–6.
[36] C.-W. Liu, W.-F. Hsien, C.-C. Yang, and M.-S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage," *Int. J. Netw. Secur.*, vol. 18, no. 5, pp. 900–916, Sep. 2016.
[37] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 7, pp. 1214–1221, Jul. 2011.
[38] J. Hur, "Improving security and efficiency in attribute-based data sharing," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 10, pp. 2271–2282, Oct. 2013.
[39] K. Yang and X. Jia, *Security for Cloud Storage Systems*. New York, NY, USA: Springer, 2014.
[40] L. Zu, Z. Liu, and J. Li, "New ciphertext-policy attribute-based encryption with efficient revocation," in *Proc. IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, Sep. 2014, pp. 281–287.
[41] S. Moffat, M. Hammoudeh, and R. Hegarty, "A survey on ciphertext-policy attribute-based encryption (CP-ABE) approaches to data security on mobile devices and its application to IoT," in *Proc. Int. Conf. Future Netw. Distrib. Syst.*, 2017, p. 34.
[42] L. Pang, J. Yang, and Z. Jiang, "A survey of research progress and development tendency of attribute-based encryption," *Sci. World J.*, vol. 2014, Jul. 2014, Art. no. 193426.
[43] R. R. Al-Dahhan, Q. Shi, G. M. Lee, and K. Kifayat, "Survey on revocation in ciphertext-policy attribute-based encryption," *Sensors*, vol. 19, no. 7, p. 1695, 2019.
[44] Z. Qiao, S. Liang, S. Davis, and H. Jiang, "Survey of attribute based encryption," in *Proc. 15th IEEE/ACIS Int. Conf. Softw. Eng., Artif. Intell., Netw. Parallel/Distrib. Comput. (SNPD)*, Las Vegas, NV, USA, Jun./Jul. 2014, pp. 1–6.
[45] C.-C. Lee, P.-S. Chung, and M.-S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *Int. J. Netw. Secur.*, vol. 15, no. 4, pp. 231–240, Jul. 2013.
[46] Y. Cheng, Z.-Y. Wang, J. Ma, J.-J. Wu, S.-Z. Mei, and J.-C. Ren, "Efficient revocation in ciphertext-policy attribute-based encryption based cryptographic cloud storage," *J. Zhejiang Univ. Sci. C*, vol. 14, no. 2, pp. 85–97, 2013.
[47] R. Tourani, S. Misra, T. Mick, and G. Panwar, "Security, privacy, and access control in information-centric networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 566–600, 1st Quart., 2018.
[48] J. Akinyele, C. Lehmann, M. Green, M. Pagano, Z. N. J. Peterson, and A. Rubin, "Self protecting electronic medical records using attribute-based encryption," IACR Cryptol. ePrint Arch., vol. 2010, 2010.
[49] H. Li, H. Duan, X. Lu, and Z. Huang, "A clinical document repository for CDA documents," in *Proc. 1st Int. Conf. Bioinf. Biomed. Eng.*, Jul. 2007, pp. 1084–1087.
[50] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, "Patient controlled encryption: Ensuring privacy of electronic medical records," in *Proc. ACM Workshop Cloud Comput. Secur. (CCSW)*, 2009, pp. 103–114.

[51] K. D. Mandl, W. W. Simons, W. C. Crawford, and J. M. Abbett, "Indivo: A personally controlled health record for health information exchange and communication," *BMC Med. Inform. Decis. Making*, vol. 7, no. 1, p. 25, Sep. 2007.
[52] K. D. Mandl, P. Szolovits, and I. Kohane, "Public standards and patients' control: How to keep electronic medical records accessible but private," *Brit. Med. J.*, vol. 322, no. 7281, pp. 283–287, Feb. 2001.
[53] P. Kumar, S. Kumar, and P. J. A. Alphonse, "Attribute based encryption in cloud computing: A survey, gap analysis, and future directions," *J. Netw. Comput. Appl.*, vol. 108, pp. 37–52, Apr. 2018.
[54] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Proc. 21st Annu. Int. Cryptol. Conf. Adv. Cryptol. (CRYPTO)*. London, U.K.: Springer-Verlag, 2001, pp. 213–229.
[55] G. R. Blakley, "Safeguarding cryptographic keys," in *Proc. AFIPS Conf.*, vol. 48. 1979, pp. 313–317.
[56] M. Ito, A. Saito, and T. Nishizeki, "Secret sharing schemes realizing general access structure," in *Proc. IEEE Global Telecommun. Conf.*, vol. 87. Tokyo, Japan: IEEE Press, 1987, pp. 99–102.
[57] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.
[58] L. Li, T. Gu, L. Chang, Z. Xu, Y. Liu, and J. Qian, "A ciphertext-policy attribute-based encryption based on an ordered binary decision diagram," *IEEE Access*, vol. 5, pp. 1137–1145, 2017.
[59] H. S. G. Pussewalage and V. A. Oleshchuk, "A patient-centric attribute based access control scheme for secure sharing of personal health records using cloud computing," in *Proc. IEEE 2nd Int. Conf. Collaboration Internet Comput.*, Nov. 2016, pp. 46–53.
[60] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute based data sharing with attribute revocation," in *Proc. 5th ACM Symp. Inf., Comput. Commun. Secur. (ASIACCS)*, 2010, pp. 1–10.
[61] S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Over-encryption: Management of access control evolution on outsourced data," in *Proc. 33rd Int. Conf. Very Large Data Bases (VLDB)*, 2007, pp. 123–134.
[62] V. Božović, D. Socek, R. Steinwandt, and V. I. Villányi, "Multi-authority attribute-based encryption with honest-but-curious central authority," *Int. J. Comput. Math.*, vol. 89, no. 3, pp. 268–283, 2012.
[63] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.
[64] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proc. ACM Conf. Comput. Commun. Secur.*, New York, NY, USA, 2007, pp. 195–203.
[65] N. Attrapadung, J. Herranz, F. Laguillaumie, B. Libert, E. de Panafieu, and C. Ràfols, "Attribute-based encryption schemes with constant-size ciphertexts," *Theor. Comput. Sci.*, vol. 422, pp. 15–38, Mar. 2012.
[66] N. Oualha and K. T. Nguyen, "Lightweight attribute-based encryption for the Internet of Things," in *Proc. 25th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Aug. 2016, pp. 1–6.
[67] D. Ding, M. Conti, and A. Solanas, "A smart health application and its related privacy issues," in *Proc. Smart City Secur. Privacy Workshop (SCSP-W)*, Vienna, Austria, Apr. 2016, pp. 1–5.
[68] J. Venkatesh, B. Aksanli, C. S. Chan, A. S. Akyurek, and T. S. Rosing, "Modular and personalized smart health application design in a smart city environment," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 614–623, Apr. 2018.
[69] F. Casino, P. López-Iturri, E. Aguirre, L. Azpilicueta, A. Solanas, and F. Falcone, "Dense wireless sensor network design for the implementation of smart health environments," in *Proc. Int. Conf. Electromagn. Adv. Appl. (ICEAA)*, Turin, Italy, Sep. 2015, pp. 752–754.
[70] C. Patsakis, R. Venanzio, P. Bellavista, A. Solanas, and M. Bouroche, "Personalized medical services using smart cities' infrastructures," in *Proc. IEEE Int. Symp. Med. Meas. Appl. (MeMeA)*, Lisboa, Portugal, Jun. 2014, pp. 1–5.
[71] X. Yao, Z. Chen, and Y. Tian, "A lightweight attribute-based encryption scheme for the Internet of Things," *Future Generat. Comput. Syst.*, vol. 49, pp. 104–112, Aug. 2015.
[72] K. D. Raghuwanshi and S. Tamrakar, "An effective access from cloud data using attribute based encryption," in *Proc. Int. Conf. Futuristic Trends Comput. Anal. Knowl. Manage.*, Feb. 2015, pp. 212–218.
[73] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy*, Berkeley, CA, USA, 2007, pp. 321–334.
[74] D. F. Ferraiolo and D. R. Kuhn, "Role-based access controls," Mar. 2009, *arXiv:0903.2171*. [Online]. Available: https://arxiv.org/abs/0903.2171
[75] A. Alrawais, A. Alhothaily, C. Hu, X. Xing, and X. Cheng, "An attribute-based encryption scheme to secure fog communications," *IEEE Access*, vol. 5, pp. 9131–9138, 2017.
[76] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute based encryption," in *Automata, Languages and Programming*, L. Aceto, I. Damgård, L. A. Goldberg, M. M. Halldórsson, A. Ingólfsdóttir, and I. Walukiewicz, Eds. Berlin, Germany: Springer, 2008, pp. 579–591.
[77] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Public Key Cryptography—PKC*, D. Catalano, N. Fazio, R. Gennaro, and A. Nicolosi, Eds. Berlin, Germany: Springer, 2011, pp. 53–70.
[78] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption," in *Advances in Cryptology—EUROCRYPT*, H. Gilbert, Ed. Berlin, Germany: Springer, 2010, pp. 62–91.
[79] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proc. ACM Conf. Comput. Commun. Secur. (CCS)*, 2007, pp. 456–465.
[80] J. Herranz, F. Laguillaumie, and C. Ràfols, "Constant size ciphertexts in threshold attribute-based encryption," in *Public Key Cryptography*, vol. 6056, P. Q. Nguyen and D. Pointcheval, Eds. Berlin, Germany: Springer, 2010, pp. 19–34.
[81] Z. Zhou, D. Huang, and Z. Wang, "Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption," *IEEE Trans. Comput.*, vol. 64, no. 1, pp. 126–138, Jan. 2015.
[82] M. Chase, "Multi-authority attribute based encryption," in *Theory of Cryptography* (Lecture Notes in Computer Science), vol. 4392, S. P. Vadhan, Ed. Berlin, Germany: Springer, 2007, pp. 515–534.
[83] M. Chase and S. S. M. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in *Proc. 16th ACM Conf. Comput. Commun. Secur. (CCS)*, Mar. 2009, pp. 121–130.
[84] H. Lin, Z. Cao, X. Liang, and J. Shao, "Secure threshold multi authority attribute based encryption without a central authority," *Inf. Sci.*, vol. 180, no. 13, pp. 2618–2632, 2010.
[85] J. Han, W. Susilo, Y. Mu, and J. Yan, "Privacy-preserving decentralized key-policy attribute-based encryption," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 11, pp. 2150–2162, Nov. 2012.
[86] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, "Secure distributed key generation for discrete-log based cryptosystems," *J. Cryptol.*, vol. 20, no. 1, pp. 51–83, 2007.
[87] X. Li and M. Zhang, "Cloud storage access control policy based on mabe," *J. Lanzhou Univ. Technol.*, vol. 42, no. 10, pp. 133–140, 2015.
[88] W. Wang, F. Qi, X. Wu, and Z. Tang, "Distributed multi-authority attribute-based encryption scheme for friend discovery in mobile social networks," *Procedia Comput. Sci.*, vol. 80, pp. 617–626, Jun. 2016.
[89] J. Wei, W. Liu, and X. Hu, "Secure and efficient attribute-based access control for multiauthority cloud storage," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1731–1742, Jun. 2018.
[90] X. Liang, R. Lu, X. Lin, and X. Shen, "Ciphertext policy attribute based encryption with efficient revocation," Univ. Waterloo, Tech. Rep., 2010, vol. 2.
[91] H. Yiliang, J. Di, and Y. Xiaoyuan, "The revocable attribute based encryption scheme for social networks," in *Proc. Int. Symp. Secur. Privacy Social Netw. Big Data*, Nov. 2015, pp. 44–51.
[92] M. Pirretti, P. Traynor, P. McDaniel, "Secure attribute-based systems," *J. Comput. Secur.*, vol. 18, no. 5, pp. 799–837, 2010.
[93] Z. Liu and D. S. Wong, "Practical attribute based encryption: Traitor tracing, revocation, and large universe," in *Proc. 13th Int. Conf. Appl. Cryptogr. Netw. Secur. (ACNS)*, 2015, pp. 1–54.
[94] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed access control in clouds," in *Proc. IEEE 10th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Nov. 2011, pp. 91–98.
[95] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute based data sharing with attribute revocation," in *Proc. 5th ACM Symp. Inf., Comput. Commun. Secur.*, Beijing, China, 2010, pp. 261–270.
[96] H. Qian, J. Li, Y. Zhang, and J. Han, "Privacy-preserving personal health record using multi-authority attribute-based encryption with revocation," *Int. J. Inf. Secur.*, vol. 14, no. 6, pp. 487–497, Nov. 2015.
[97] K. Fan, Q. Tian, J. Wang, H. Li, and Y. Yang, "Privacy protection based access control scheme in cloud-based services," *China Commun.*, vol. 14, no. 1, pp. 61–71, 2017.
[98] X. Xu, J. Zhou, X. Wang, and Y. Zhang, "Multi-authority proxy re-encryption based on CPABE for cloud storage systems," *J. Syst. Eng. Electron.*, vol. 27, no. 1, pp. 211–223, Feb. 2016.

[99] X.-d. Yang, M.-m. Yang, P. Yang, and Q. Leng, "A multi-authority attribute-based encryption access control for social network," in *Proc. 3rd IEEE Int. Conf. Control Sci. Syst. Eng.*, Aug. 2017, pp. 671–674.

[100] K. Yang, X. Jia, K. Ren, B. Zhang, and R. Xie, "DAC-MACS: Effective data access control for multi-authority cloud storage systems," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 11, pp. 1790–1801, 2013.

[101] K. Yang and X. Jia, "Expressive, efficient, and revocable data access control for multi-authority cloud storage," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 7, pp. 1735–1744, Jul. 2014.

[102] K. Yang and X. Jia, "Attributed-based access control for multi-authority systems in cloud storage," in *Proc. IEEE 32nd Int. Conf. Distrib. Comput. Syst.*, Jun. 2012, pp. 536–545.

[103] Y. Imine, A. Lounis, and A. Bouabdallah, "ABR: A new efficient attribute based revocation on access control system," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Valencia, Spain, Jun. 2017, pp. 735–740.

[104] C.-I. Fan, C.-N. Wu, C.-H. Chen, Y.-F. Tseng, and C.-C. Feng, "Attribute-based proxy re-encryption with dynamic membership," in *Proc. 10th Asia Joint Conf. Inf. Secur.*, May 2015, pp. 26–32.

[105] Y. Yasumura, H. Imabayashi, and H. Yamana, "Attribute-based proxy re-encryption method for revocation in cloud storage: Reduction of communication cost at re-encryption," in *Proc. IEEE 3rd Int. Conf. Big Data Anal.*, Mar. 2018, pp. 312–318.

[106] Y. Yasumura, H. Imabayashi, and H. Yamana, "Attribute-based proxy re-encryption method for revocation in cloud data storage," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 4858–4860.

[107] L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, and W. Jonker, "Mediated ciphertext-policy attribute-based encryption and its application," in *Information Security Applications*. Berlin, Germany: Springer, 2009, pp. 309–323.

[108] S. Jahid, P. Mittal, and N. Borisov, "EASiER: Encryption-based access control in social networks with efficient revocation," in *Proc. ACM Symp. Inf., Comput. Commun. Secur. (ASIACCS)*, Mar. 2011, pp. 411–415.

[109] J. Li, W. Yao, J. Han, Y. Zhang, and J. Shen, "User collusion avoidance CP-ABE with efficient attribute revocation for cloud storage," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1767–1777, Jun. 2018.

[110] J. Li, W. Yao, Y. Zhang, H. Qian, and J. Han, "Flexible and fine-grained attribute-based data storage in cloud computing," *IEEE Trans. Service Comput.*, vol. 10, no. 5, pp. 785–796, Sep./Oct. 2017.

[111] Z. Liu, Z. L. Jiang, X. Wang, and S. M. Yiu, "Practical attribute-based encryption: Outsourcing decryption, attribute revocation and policy updating," *J. Netw. Comput. Appl.*, vol. 108, pp. 112–123, Apr. 2018.

[112] G. Wang, Q. Liu, J. Wu, and M. Guo, "Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers," *Comput. Secur.*, vol. 30, no. 5, pp. 320–331, 2011.

[113] C. Gentry and A. Silverberg, "Hierarchical ID-based cryptography," in *Advances in Cryptology—ASIACRYPT* (Lecture Notes in Computer Science), vol. 2501, 2002, pp. 548–566.

[114] Q. Huang, Y. Yanga, and M. Shen, "Secure and efficient data collaboration with hierarchical attribute-based encryption in cloud computing," *Future Gener. Comput. Syst.*, vol. 72, pp. 239–249, Jul. 2016.

[115] Y. Xie, H. Wen, B. Wu, Y. Jiang, and J. Meng, "A modified hierarchical attribute-based encryption access control method for mobile cloud computing," *IEEE Trans. Cloud Comput.*, vol. 7, no. 2, pp. 383–391, Apr./Jun. 2019.

[116] J. S. Su, D. Cao, X. F. Wang, Y. P. Sun, and Q. L. Hu, "Attribute-based encryption schemes," *J. Softw.*, vol. 2, no. 6, pp. 1299–1315, 2011.

[117] E. Luo, Q. Liu, and G. Wang, "Hierarchical multi-authority and attribute-based encryption friend discovery scheme in mobile social networks," *IEEE Commun. Lett.*, vol. 14, no. 8, pp. 1772–1775, Aug. 2016.

[118] L. You and L. Wang, "Hierarchical authority key-policy attribute-based encryption," in *Proc. ICCT*, Oct. 2015, pp. 868–872.

[119] J. Shuci, G. Weibin, and F. Guisheng, "Hierarchy attribute-based encryption scheme to support direct revocation in cloud storage," in *Proc. IEEE/ACIS 16th Int. Conf. Comput. Inf. Sci. (ICIS)*, May 2017, pp. 869–874.

[120] B. K. Gowda and R. Sumathi, "Hierarchy attribute-based encryption with timing enabled privacy preserving keyword search mechanism for e-health clouds," in *Proc. 2nd IEEE Int. Conf. Recent Trends Electron., Inf. Commun. Technol. (RTEICT)*, Bangalore, India, May 2017, pp. 425–429.

[121] F. Qi, K. Li, and Z. Tang, "A multi-authority attribute-based encryption scheme with attribute hierarchy," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. Appl. IEEE Int. Conf. Ubiquitous Comput. Commun. (ISPA/IUCC)*, Dec. 2017, pp. 607–613.

[122] J. G. Sun, J. Liu, and L. Y. Zhao, "Clustering algorithms research," *J. Softw.*, vol. 19, no. 1, pp. 48–61, 2008.

[123] T. Nishide, K. Yoneyama, and K. Ohta, "Attribute-based encryption with partially hidden encryptor-specified access structures," in *Applied Cryptography and Network Security* (Lecture Notes in Computer Science), vol. 5037, S. M. Bellovin, R. Gennaro A. Keromytis, and M. Yung, Eds. 2008, pp. 111–129.

[124] Y. Zhang and D. Zheng, "Anonymous attribute-based encryption with large universe and threshold access structures," in *Proc. IEEE Int. Conf. Comput. Sci. Eng. (CSE) IEEE Int. Conf. Embedded Ubiquitous Comput. (EUC)*, Jul. 2017, pp. 870–874.

[125] A. Kapadia, P. P. Tsang, and S. W. Smith, "Attribute-based publishing with hidden credentials and hidden policies," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*. Reston, VA, USA: The Internet Soc., 2007, pp. 179–192.

[126] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Advances in Cryptology–EUROCRYPT* (Lecture Notes in Computer Science), vol. 4965, N. Smart, Ed. Berlin, Germany: Springer, 2008, pp. 146–162.

[127] Y. Zhang, J. Li, X. Chen, and H. Li, "Anonymous attribute-based proxy re-encryption for access control in cloud computing," *Secur. Commun. Netw.*, vol. 9, no. 14, pp. 2397–2411, 2016.

[128] D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in *Proc. 4th Theory Cryptogr. Conf. (TCC)* Lecture Notes in Computer Science, vol. 4392, V. Salil, Ed. Berlin, Germany: Springer, 2007, pp. 535–554.

[129] J. Lai, R. H. Deng, and Y. Li, "Fully secure cipertext-policy hiding CP-ABE," in *Proc. 7th Int. Conf. Inf. Secur. Pract. Exper. (ISPEC)* Lecture Notes in Computer Science, vol. 6672, F. Bao J. Weng, Eds. Berlin, Germany: Springer, 2011, pp. 24–39.

[130] Y. Zhang, X. Chen, D. S. Wong, and H. Li, "Anonymous attribute-based encryption supporting efficient decryption test," in *Proc. 8th ACM SIGSAC Symp. Inf., Comput. Commun. Secur. (ASIACCS)*. New York, NY, USA: ACM, 2013, pp. 511–516.

[131] Y. Zhang, X. Chen, J. Li, D. S. Wong, H. Li, and I. You, "Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing," *Inf. Sci.*, vol. 379, pp. 42–61, Feb. 2017.

[132] L. Sun and C. Xu, "Hidden policy ciphertext-policy attribute based encryption with conjunctive keyword search," in *Proc. 3rd IEEE Int. Conf. Comput. Commun.*, Dec. 2017, pp. 1439–1443.

[133] J. Lai, R. H. Deng, and Y. Li, "Expressive CP-ABE with partially hidden access structures," in *Proc. 7th ACM Symp. Inf., Comput. Commun. Secur. (AsiaCCS)*, Seoul, Korea, May 2012, pp. 1–12.

[134] H. Cui, R. H. Deng, G. Wu, and J. Lai, "An efficient and expressive ciphertext-policy attribute-based encryption scheme with partially hidden access structures," in *Proc. Int. Conf. Provable Secur.* (Lecture Notes in Computer Science), vol. 10005, L. Chen and J. Han, Eds. 2016, pp. 19–38.

[135] J. Li, K. Ren, B. Zhu, and Z. Wan, "Privacy-aware attribute-based encryption with user accountability," in *Information Security* (Lecture Notes in Computer Science), vol. 5735, P. Samarati, M. Yung, F. Martinelli, and C. A. Ardagna, Eds. Berlin, Germany: Springer, 2009, pp. 347–362.

[136] R. Xu and J. B. D. Joshi, "An integrated privacy preserving attribute based access control framework," in *Proc. IEEE 9th Int. Conf. Cloud Comput.*, Jun./Jul. 2016, pp. 68–76.

[137] O. Goldreich and R. Ostrovsky, "Software protection and simulation on oblivious RAMs," *J. ACM*, vol. 43, no. 3, pp. 431–473, 1996.

[138] M. S. Islam, M. Kuzu, and M. Kantarcioglu, "Access pattern disclosure on searchable encryption: Ramification, attack and mitigation," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2012, pp. 1–15.

[139] G. Lin, H. Hong, and Z. Sun, "A collaborative key management protocol in ciphertext policy attribute-based encryption for cloud data sharing," *IEEE Access*, vol. 5, pp. 9464–9475, 2017.

[140] D. Chen, L. Wan, C. Wang, S. Pan, and Y. Ji, "A multi-authority attribute-based encryption scheme with pre-decryption," in *Proc. 7th Int. Symp. Parallel Architectures, Algorithms Program.*, Dec. 2015, pp. 223–228.

[141] S. M. Sedaghat, M. H. Ameri, J. Mohajeri, and M. R. Aref, "An efficient and secure data sharing in smart grid: Ciphertext-policy attribute-based signcryption," in *Proc. 25th Iranian Conf. Electr. Eng. (ICEE)*, May 2017, pp. 2003–2008.

[142] G. Martin, S. Narayan, and R. Safavi-Naini, "Threshold attribute-based signcryption," in *Proc. Int. Conf. Secur. Cryptogr. Netw.* Berlin, Germany: Springer, 2010, pp. 154–171.

[143] E. Keita, A. Miyaji, and M. S. Rahman, "Dynamic attribute-based signcryption without random oracles," *Int. J. Appl. Cryptogr.*, vol. 2, no. 3, pp. 199–211, 2012.
[144] K. Yang, X. Jia, and K. Ren, "Secure and verifiable policy update outsourcing for big data access control in the cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 12, pp. 3461–3470, Dec. 2015.
[145] C. Ge, W. Susilo, J. Wang, Z. Huang, L. Fang, and Y. Ren, "A key-policy attribute-based proxy re-encryption without random oracles," *Comput. J.*, vol. 59, pp. 970–982, Nov. 2015.
[146] H. Hong and Z. Sun, "Towards secure data sharing in cloud computing using attribute based proxy re-encryption with keyword search," in *Proc. IEEE 2nd Int. Conf. Cloud Comput. Big Data Anal.*, Apr. 2017, pp. 218–223.
[147] X. Liang, Z. Cao, H. Lin, and J. Shao, "Attribute based proxy re-encryption with delegating capabilities," in *Proc. 4th Int. Symp. Inf., Comput. Commun. Secur. (ASIACCS)*, 2009, pp. 276–286.
[148] S. Luo, J. Hu, and Z. Chen, "Ciphertext policy attribute-based proxy re-encryption," in *Information and Communications Security*, M. Soriano, S. Qing, and J. López, Eds. Berlin, Germany: Springer, 2010, pp. 401–415.
[149] T. Mizuno and H. Doi, "Hybrid proxy re-encryption scheme for attribute-based encryption," in *Information Security and Cryptology*, F. Bao, M. Yung, D. Lin, and J. Jing, Eds. Berlin, Germany: Springer, 2010, pp. 288–302.
[150] W. Susilo, P. Jiang, F. Guo, G. Yang, Y. Yu, and Y. Mu, "EACSIP: Extendable access control system with integrity protection for enhancing collaboration in the cloud," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 12, pp. 3110–3122, Dec. 2017.
[151] J. Li, Q. Yu, and Y. Zhang, "Hierarchical attribute based encryption with continuous leakage-resilience," *Inf. Sci.*, vol. 484, pp. 113–134, May 2019.
[152] J. Li, Q. Yu, and Y. Zhang, "Key-policy attribute-based encryption against continual auxiliary input leakage," *Inf. Sci.*, vol. 470, pp. 175–188, Jan. 2019.
[153] L. Zhang, J. Zhang, and Y. Mu, "Novel leakage-resilient attribute-based encryption from hash proof system," *Comput. J.*, vol. 60, no. 4, pp. 541–554, 2017.
[154] J.-X. Zhang and L.-Y. Zhang, "Anonymous CP-ABE against side-channel attacks in cloud computing," *J. Inf. Sci. Eng.*, vol. 33, no. 3, pp. 789–805, 2017.
[155] J. Li, Q. Wang, C. Wang, and K. Ren, "Enhancing attribute-based encryption with attribute hierarchy," *Mobile Netw. Appl.*, vol. 16, no. 5, pp. 553–561, 2011.
[156] T. Tassa, "Hierarchical threshold secret sharing," in *Theory of Cryptography*, 2004, pp. 473–490.
[157] X. Liu, J. Ma, J. Xiong, and G. Liu, "Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data," *Int. J. Netw. Secur.*, vol. 16, no. 6, pp. 437–443, Nov. 2014.

KENNEDY EDEMACU received the B.S. degree in computer science from Gulu University, in 2011, and the M.S degree in data communication and software engineering from Makerere University, in 2014. He is currently pursuing the Ph.D. degree in computer science with Sangmyung University. He was an Assistant Lecturer with Muni University, from 2013 to 2016. His current research interests include cloud privacy, cryptography, and artificial intelligence.

HUNG KOOK PARK received the Ph.D. degree in information systems from the Claremont Graduate School, CA, USA. He is currently a Professor with the Department of Computer Science, Sangmyung University. He has made several publications in reputable journals and conferences.

BEAKCHEOL JANG received the B.S. degree from Yonsei University, in 2001, the M.S. degree from the Korea Advanced Institute of Science and Technology, in 2002, and the Ph.D. degree from North Carolina State University, in 2009, all in computer science. He is currently an Associate Professor with the Department of Computer Science, Sangmyung University. His current research interests include wireless networking, big data, the Internet of Things, and artificial intelligence.

JONG WOOK KIM received the Ph.D. degree from the Computer Science Department, Arizona State University, in 2009. He was a Software Engineer with the Query Optimization Group, Teradata, from 2010 to 2013. He is currently an Assistant Professor of computer science with Sangmyung University. His current research interests include data privacy, distributed databases, and query optimization.