Hybrid Solution for Privacy-Preserving Access Control for Healthcare Data

Mukalel Bhaskaran SMITHAMOL, Sridhar RAJESWARI
Department of Computer Science and Engineering, College of Engineering, Guindy, Anna University, Chennai, 600025, India
smithamolm@acm.org

***Abstract*—The booming in cloud and IoT technologies has accelerated the growth of healthcare system. The IoT devices monitor the patient's health, and upload collected data as Electronic Medical Records (EMRs) to the cloud for storage and sharing. Outsourcing EMRs to the cloud introduce new security and privacy challenges. In this paper, we proposed a novel architecture ensuring security and privacy for the outsourced health records. The proposed model uses partially ordered set (POSET) for constructing the group based access structure and Ciphertext-Policy Attribute-Based Encryption (CP-ABE) to provide fine-grained EMR access control. The modified group based CP-ABE (G-CP-ABE) minimizes the computational overhead by reducing the number of leaf nodes in the access tree. Also, the proposed G-CP-ABE framework merges symmetric encryption and CP-ABE scheme to minimize the overall encryption time. As a result, G-CP-ABE can be used to monitor health conditions even from a resource constrained IoT device. The performance analysis shows the efficiency of the proposed model, making it suitable for practical use.**

***Index Terms*—data privacy, electronic medical records, internet of things, cloud, access control.**

I. Introduction

The promising potential of cloud and IoT empowered to have interconnected medical devices and sensors to provide healthcare services. In a healthcare system, many devices and things (smartphone, tablet, RFID, sensors, implantable medical devices) will be connected to the Internet providing health information and services [1]. The information collected by these devices are recorded in the form of EMRs and uploaded to cloud for sharing.

The advantages of cloud computing [2] paradigm have attracted healthcare industry towards outsourcing medical database. In fact, when storing or processing data in the cloud, the data owner loses the control over the data and is exposed to unauthorized users including the cloud server [3]. As health records store very sensitive information, sharing and storage of EMRs in the cloud remains a great challenge. Development of middleware applications using the data for things and cloud provide exceptional health services [4]. Cloud and IoT together are expected to revolutionize the healthcare industry applications.

Rising to the security challenges in the cloud, researchers have proposed many schemes for data confidentiality and privacy [3], [5]. Some efforts are being made to provide secure health services in cloud following HIPPA (Health Insurance Portability and Accountability Act) privacy and security guidelines. However, the lack of effective access privacy policies prohibits the adoption of cloud for EMR outsourcing.

Information collected by IoT devices is highly sensitive and should be handled appropriately to ensure the privacy. Privacy involves access control indicating the claim of users to determine for themselves about the characteristics of communications done [6]. Though encryption provides data confidentiality, fine-grained data access is difficult with traditional encryption standards. Moreover, encrypted data should be open to sharing and access control policy.

To fill this gap, Sahai and Waters introduced the concept of Attribute-based Encryption (ABE) [7]. ABE permits to embed access policy in the shared data itself, and it follows one-to-many encryption mode. In the subsequent work, Goyal, Pandey, Sahai, and Waters [8] improved the concept of ABE by proposing two variants namely key-policy ABE (KP-ABE) and Ciphertext-Policy ABE (CP-ABE).

CP-ABE is extremely useful in EMR system as data owners (hospital) can make and enforce access policy in the encrypted text. Most of the existing CP-ABE scheme uses bilinear maps and generates secret keys and ciphertext of large size. The size of keys and ciphertext is linear to the attributes involved, and the number of bilinear pairings is directly proportional to attribute size. The use of ABE in IoT devices and healthcare monitoring applications is challenging due to the heavy computation of bilinear pairings.

The key objective of this work is to provide simple and computationally effective access control policy using CP-ABE with minimal bilinear pairings. In this study, a novel method is proposed to create group based access structure using POSET reducing the computational overhead of encryption and decryption considerably. The POSET based access structure reduces the number of bilinear pairings in encryption and decryption process. Also, the proposed model G-CP-ABE merges symmetric encryption and CP-ABE scheme to achieve data confidentiality and access privacy effectively.

The major contributions in this article can be summarized as follows:

- Proposed and implemented a novel cloud-based security architecture G-CP-ABE with reduced bilinear pairings for EMR database uploading that realizes secure and efficient access control.
- A novel method to construct the access tree is proposed. The proposed access control system uses POSET to create the group access structure which reduces the number of bilinear pairings.

This work was supported in part by the Quality Improvement Programme of All India Council for Technical Education, India.

Digital Object Identifier 10.4316/AECE.2017.02005

- The proposed model reduces overall encryption and decryption time by merging symmetric encryption and CP-ABE encryption.
- Provided security and performance analysis ensuring efficiency of the proposed architecture.

The structure of the paper is as follows. Section 2 briefly describes related works. Section 3 furnishes the preliminaries needed to explain the proposed security model. Section 4 gives the detailed system architecture and the generation of security solution G-CP-ABE. Section 5 and 6 provides the security and experimental analysis. Finally, Section 7 summarizes our findings and offers valuable insight into potential future improvements.

II. Related Works

In this section, we summarize the related work on ABE and the various flexible ABE schemes used in providing fine-grained access control of EMRs in the cloud.

ABE is an enhanced functional version of identity-based encryption. Since the introduction of ABE in implementing fine grained access control by Sahai and Waters, many research works have been published on ABE-based access control. Basically ABE has two variants: KP-ABE [8] and CP-ABE [9]. In CP-ABE, decryption keys are associated with the set of attributes whereas, in the case of KP-ABE, keys are associated with access structure. Fine-grained access control policy is essential in the scenario of outsourcing EMRs to the cloud [10].

After the introduction of ABE, several KP-ABE [11-12] security models and CP-ABE [13–18] security models are presented in the literature. However, CP-ABE is preferred than KP-ABE in EMR sharing since data owner can specify the access structure during encryption.

Alshehri, Radziszowski, and Raj suggest a secure access policy by the direct adaptation of CP-ABE in accessing healthcare data [19]. Authors presented the feasibility of using CP-ABE in EMR accessing from the cloud. Li, Lou and Ren [20] discuss the issue in the data security and privacy related to Wireless Body Area Network (WBAN). Authors addressed various challenges concerned with confidentiality and security of data collected inside WBAN and outsourcing to the cloud. A new prototype for EMR accessing on a mobile device is proposed using basic CP-ABE [21]. The work focuses on EMR availability even in offline but ignores the computation and storage overhead induced in mobile devices.

Fabian, Ermakova, and Junghanns propose collaborative and secure sharing of EMR across multiple clouds [22]. Yang, Li, and Niu [23] recommend vertical fragmentation of EMR database and introduce a hybrid search over plaintext and ciphertext. The method ignores integrity checking in a multi-user scenario. The authors [24], suggests a framework for fine-grained access control in EMR data sharing and search by employing CP-ABE scheme. Liu, Zhang, Ling, and Liu provide a scheme for EHR access control policy embedded in linear secret sharing codes [25]. However, their model incurs many expensive bilinear pairings.

Employing CP-ABE scheme in the access policy construction of EMR data outsourcing makes the computation efficiency of outsourced data a real challenge. CP-ABE encryption involves many expensive bilinear pairing operations, and the number of bilinear pairing linearly increases with the number of attributes.

The study reveals the opportunity to improve and optimize the existing solutions to enhance the performance of CP-ABE. There exists a wide variety of optimization techniques which has proven their power in optimization in various field of engineering applications [26–29]. We proposed a graphical method to optimize the result of existing ABE technique.

Most of the existing work focuses on either embedding access policy in the ciphertext by adapting CP-ABE directly or outsourcing computation to a third party. Both the approaches are not feasible for large EMR tables since the number of bilinear pairing operations proportional to the number of attributes and depth of the access tree. Given these challenges, the proposed security framework aims at reducing bilinear pairings using POSET based group access policy. Also, the EMR tables are encrypted with faster symmetric encryption, and the access policy is embedded in the key ciphertext which reduces overall encryption and decryption time significantly.

Before proceeding to the proposed framework, next section briefly outlines the preliminary concepts needed for CP-ABE.

III. Background

In this section, we briefly explain the bilinear maps, access structure, and general CP-ABE scheme. The definitions assume that a set of attributes describes a group access structure and the ciphertext embeds the access structure.

Bilinear Maps [30]. Consider G_1, G_2, of prime order p. We consider g a generator of G_1. Typically, G_1 is an elliptic curve and G_2 is a finite field. Let e describes a bilinear map, e: $G_1 \; X \; G_1 \rightarrow G_2$. The bilinear map e has the following useful properties:

1. Bilinearity: for all p, q in G_1 and a, b in Z_p , we have, $e(p^a, q^b) = e(p,q)^{ab}$,
2. Non-degeneracy: There exists p, q in G_1 such that $e(p,q) \neq 1$.
3. Computability: For all, $p, q \in G_1$, there is an efficient computation of $e(p, q)$.

The map e is symmetrical since $e(p^a, p^b) = e(p,p)^{ab} = e(p^b, p^a)$.

Access Structure Let {1, 2… n} be a set of parties. Then, a collection $A \subseteq 2^{\{1,2,\ldots,n\}}$ is a monotone if $\forall B, C$ $B \in A, B \subseteq C$ then $C \in A$. A monotone access structure is a collection A of non-empty subsets of $2^{\{1,2,\ldots,n\}}$. The sets in A are called authorized sets and not in A are called unauthorized sets. In this article parties are attributes and access structure is authorized set of attributes.

CP-ABE consists of four fundamental algorithms [9]: Setup, Key Generation (KeyGen), Encryption, and Decryption. Let U be the universal set of attributes describing data properties and user properties.

- **Setup**. It takes implicit security parameter as the input and produces the public key *PK* and master key *MK*.

- **KeyGen** $(MK, S) \rightarrow SK_S$: The key generation takes as input master key MK and a set of attributes S and outputs SK_S the secret key associated with S.
- **Encrypt**$(PK, M, A) \rightarrow C$: The data owner performs the encryption. The algorithm takes input as public key PK, message M and the access structure A defined over U. Encryption is carried out according to A and the ciphertext CT is the output. We assume that ciphertext implicitly contains A.
- **Decrypt** $(PK, C, SK_S) \rightarrow M$: The data user runs the decryption algorithm. The decryption algorithm takes public key PK, ciphertext C, and user secret key SK_S as the input. It outputs original message M if SK_S satisfies the access structure A.

The security model for CP-ABE. The challenger runs the setup algorithm and delivers the public key PK to the adversary.

- **Phase 1**: The adversary makes repeated secret keys corresponding to set of attributes $S_1, S_2, \ldots, S_{q1}$.
- **Challenge**: The adversary submits two messages M_1 and M_2 where $|M1| = |M2|$. Also, the adversary submits challenge access structure A^* where none of the sets S_i in phase1 satisfy A^*. The challenger guesses a random bit b and encrypts M_b using A^* and gives the ciphertext CT^* to the adversary.
- **Phase 2**: Phase 1 repeated with the restriction that S_i does not satisfy A^*.
- **Guess**: The adversary outputs guess bit $b' \in \{0,1\}$ of b and it wins if $b' = b$.

The model can be easily extended to manage chosen ciphertext attacks.

In the following section, based on the CP-ABE we formalize the security framework for cloud and IoT-enabled healthcare system.

IV. Proposed Security Framework G-CP-ABE

Privacy-aware security framework, Group CP-ABE (G-CP-ABE), is proposed to manage access control over EMRs outsourced to the cloud. The proposed architecture enables a healthcare system to handle the enormous amount of data generated by IoT devices to manage patient supervision. Since the collected data are very sensitive and private health information, the proposed model ensures confidentiality and access privacy. Figure 1 shows the architecture in detail. The major entities in the system are:

- ***Trusted Central Authority (TCA)*:** Initiates each group with unique group key based on the authentic set of attributes.
- ***EMR data owner***: Defines access structure for each group and performs the encryption before uploading to EMR cloud
- ***EMR Cloud*:** The cloud service provider is considered as a semi-trusted entity. It provides storage and other data transaction services.
- ***Data user*:** A data user wants to access EMR in the cloud. It can download the EMRs from the cloud and decrypts as per the satisfying group access structure.

The IoT devices (sensors, ECG monitor, breathing activity monitor, and other implantable medical devices) continuously monitor patient's health status and send theses information to the server. The internal server aggregates, processes, and uploads these data as EMRs.

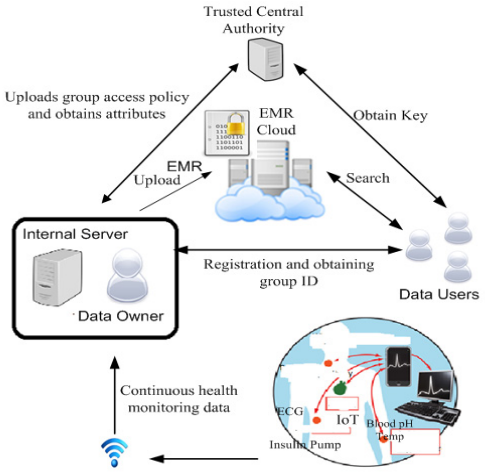

Figure 1. G-CP-ABE architecture

The data owner creates and manages a flexible small number of groups since for any healthcare system the users of the system can be predetermined easily. The architecture is assumed to follow centralized group management. We use $(C, \leq)$ to denote the hierarchical group organization, where $\leq$ is a partial order on C. Obviously $(C, \leq)$ is a partially ordered set (POSET). A group represents a collection of users, and any two groups are disjoint. Any POSET can be represented as an access graph $G = (V, E)$, in which vertices represents groups and edges represent connectivity from the predecessor to successor. G is a directed acyclic graph. The following section gives the details of the group access structure.

A. Proposed Group Access Policy

Consider an example of EMR table shown in Table I. There are five fields in the table. The access structure created based on AND gate for Table I is shown in Fig. 2.

TABLE I. EMRs in a Hospital

ID (A)	Name (B)	Age (C)	Disease (D)	Symptom(E)
A00125	Nirg Geng	38	varicella	Fever
A01345	Yi Zeng Li	21	Chikungunya	Fever, joint pain
A01267	Joseph V	56	Tuberculosis	Cough, fever

Each internal node in the access tree is threshold gate and the leaf node is an attribute [7]. We use *parent(x)* to denote parent of node x and *att(x)* to denote the attribute of x if x is a leaf node. The tree T is constructed according to the POSET defined for the group creation.

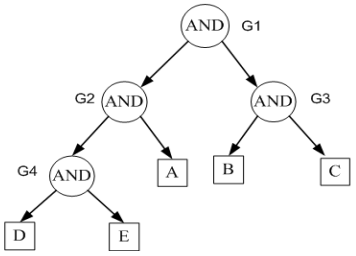

Figure 2. Access tree, T

For example, we define four groups, namely G_1, G_2, G_3, and G_4 for Table I. As per the tree T, each group has G_1={A,B,C,D,E}, G_2 ={A,D,E}, G3={B,C} and G4={D,E} set of attributes. According to the definition of POSET, $G2 \leq G1$ indicates G_1 has more access rights than G_2. Here, G_1 is the maximal root group which can view all attributes. Each group can view its attributes plus the attributes of its child groups.

Since the encryption of whole database using CP-ABE involves many expensive bilinear pairing operations inducing high computational overhead, we use symmetric encryption to encrypt the EMR database. The computational time required for symmetric encryption is less than that of CP-ABE. Hence, the proposed G-CP-ABE has two phases of encryption. In phase1, the EMR database is encrypted using a robust symmetric encryption (AES-256 bit key), and in phase 2 the keys used for symmetric encryption are encrypted using CP-ABE. Hence, the two level encryption reduces computation time and overhead significantly.

Based on the access tree *T*, random keys for symmetric encryption are chosen. Consider our example tree *T* in Fig. 2 and the keys were chosen are $G_4 \leftarrow k_1$, $G_3 \leftarrow k_2$, $G_2 \leftarrow (k_1,k_3)$, $G_4 \leftarrow (k_1,k_2,k_3)$. The assignment implies that fields {D, E} of database is encrypted using k_1, {B,C} using k_2 and {A} using k_1. Now, the key table is constructed as follows:

TABLE II. KEY TABLE K

G_1	G_2	G_3	G_4
$\{\{k_1\},\{k_2\},k_3\}$	$\{\{k_1\},k_3\}$	$\{k_2\}$	$\{k_1\}$

The key table indicates the partial ordering among groups. Group G_4 can decrypt only key k_1 and can decrypt and view attributes {D, E}, whereas group G_1 can decrypt all keys and can see all attributes of the database.

Now, as per the POSET construction, modified access tree is created and is shown in Fig.3.

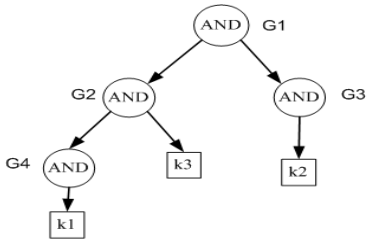

Figure 3. Modified access tree, T'

The access tree is drawn according to the key table in Table II. The modified access tree has optimized the number leaf nodes as compared to that of original access tree shown in Fig.2. The reduction of leaf nodes is significant achievement since the number of bilinear pairings is proportional to leaf nodes. In the real situation, the EMR table has many attributes, and the modified access tree construction qualify G-CP-ABE to perform encryption and decryption faster. The decryption time of CP-ABE significantly depends on the number of leaf nodes in the access tree [9]. The reduced leaf nodes provide gain in overall computation time significantly.

When a user wishes to search for one or more data fields, she should send the secret key SK_S corresponding to the group identity to the cloud server. If the key SK_S satisfy the access structure of the respective group then the server returns the encrypted database table. The data user runs the decryption algorithm to obtain the intended decryption keys.

Then, the EMR owner encrypts the key table *K* using CP-ABE scheme and embeds the access policy in the ciphertext of the key table. The attributes of the key table are the random symmetric keys. The number of random keys is always finitely small as it directly depends on the number of groups. For Table I data, the number of groups is four, and the number of keys is three. Therefore, the number of keys is always less than or equal to the number of groups. As a result, the number of attributes is reduced. Recall that the number of bilinear pairing operations and length of encryption keys is dependent on the number of attributes in the table. Reduction of attributes always reduces the computations involved providing performance enhancement. Also, instead of encrypting the original table, the reduced small key table is encrypted using CP-ABE which minimizes the overall encryption and decryption time.

The decryption procedure is simple. The data user downloads the ciphertext file *C* from the EMR cloud. The file has two parts. First part stores the ciphertext of key table *K* and the user decrypts this part to obtain the respective symmetric decryption key provided user possess the required matching group attribute set. The construction of the framework is described below.

B. G-CP-ABE Construction

This section provides a detailed construction of the proposed G-CP-ABE scheme. Private keys of a user are identified with the given group attribute set. We will use a hash function $H: \{0,1\}^* \rightarrow G_0$, to map the binary representation of attribute to a random group element. A universe of attribute set is defined as $U= \{A_1,A_2,...,A_n\}$. In addition to the phases of the CP-ABE scheme, the proposed G-CP-ABE scheme has symmetric key setup, symmetric encryption, and symmetric decryption. When data volume is large, CP-ABE incurs more computational overhead. Therefore, symmetric encryption is the better choice for achieving computational efficiency.

***Setup*(1^m, *U*)→(*PK, MK*)**: TCA runs the setup algorithm. It chooses a bilinear group G_0 of prime order p and generator g. The security parameter is m and selects two random elements α, β in Z_p. The algorithm outputs the public key *PK* and master key *MK* as follows:

$$PK = \{G_0, g, h = g^{\beta}, e(g,g)^{\alpha}\} \quad (1)$$

$$MK = \{g^{\alpha}, \beta\} \quad (2)$$

***KeyGen*(*PK, MK, S*)→SK_S**: The key generation algorithm takes as input group attribute set *S*(*S in U*) and outputs secret key SK_S, identifies with that set. The algorithm randomly chooses $r \in Z_p$ and $r_i \in Z_p$ for each attribute *i in S*. The key is computed as follows:

$$SK_S = (D = g^{\alpha}.h^{r}, \forall i \in S: D_i = g^{r}.H(i)^{r_i}, D_i' = h^{r_i}) \quad (3)$$

Now, the benefit is each attribute set has minimum one attribute and maximum m attributes, where m gives the total number of random keys used. Less number of attributes reduces the computation and length of the CP-ABE secret key. G-CP-ABE takes less key generation time since the modified access tree has less number of attributes. The number of expensive exponentiations carried out is reduced significantly with modified access tree.

The data owner invokes subroutine *Encrypt(PK, K, A)*.

Here, *PK* is the public key, *K* is the symmetric key set and *A* is the group access structure. The algorithm outputs ciphertext C_2.

The data user can access the corresponding decryption key only if she has satisfying attribute set. The data owner selects *m* random numbers $x_1,x_2,...,x_m$ in Z_p. Then it computes C_i^1 and C_i^2 for $i = 1,2,...,m$.

$$C_i^1 = k_i.e(g,g)^{\alpha.x_i},\ C_i^2 = g^{x_i} \tag{4}$$

The number of exponentiation operation in G-CP-ABE is very few as *m* is finitely small. The algorithm selects polynomial q_y for each node *y* in the access tree. The node information is randomly chosen from top to bottom. For each node *y*, the degree of the polynomial is set as k_y-1 where k_y is the threshold value for that node. From the root node *R*, data owner sets $q_R(0)=x_1$, and chooses other points to define the polynomial q_R. The points in the polynomial are made up of two types of nodes, level nodes and randomly selected nodes from the access tree. It sets $q_y(0)=q_{parent}(index(y))$. Now consider the set of leaf nodes *L*, then the data owner computes C_i^1(z) and C_i^2(z) for each leaf node *z* in *L* as follows:

$$C_z^1 = h^{q_y(0)}, C_z^2 = H(att(y))q_y(0) \tag{5}$$

Now, the integrated key set ciphertext, $C_2 \leftarrow \{ C_i^1, C_i^2, C_z^1, C_z^2\}$. Now the symmetric encryption also executes iteratively with *m* keys for each subset of attributes and the final ciphertext uploaded is $\{C_1, C_2\}$.

***Decrypt(C, SK$_S$)*:** The data user invokes this subroutine iteratively. Similar to CP-ABE, we define a recursive function *DecryptNode(C_2, SK_S,y)* that takes C_2 (the *G-CP-ABE* ciphertext), users secret key SK_S, and a node *y*. If the node *y* is a leaf node, then *i=att(y)* and if *i* is in *S*, then

$$DecryptNode(C_2, SK_S, y) = \frac{e(D_i, C_y^1)}{e(D_i^{'}, C_y^2)} = \frac{e(g^r.H(a_i)^{r_i}, h^{q_y(0)})}{e(h^{r_i}, H(att(y))q_y(0))} = e(g,g)^{r.\beta.q_y(0)} \tag{6}$$

If *i* is not in *S* then *DecryptNode(C_2,SK_S,y)is set as null. If y* is not a leaf node then, for each node z, where z = child(y), it runs DecryptNode(C2, SKS, z) and stores the result in Fz. Let Sx be an arbitrary kx sized set of child nodes z such that Fz not null. If no such node exists, then it implies that the node was not satisfied and returns null. Otherwise, we compute,

$$F_y = \prod_{z \in S_y} F_z^{\Delta_i S^{'}_{y(0)}} = \prod_{z \in S_y} (e(g,g)^{r.\beta.q_z(0)})^{\Delta_i S^{'}_{y(0)}} = \prod_{z \in S_y} (e(g,g)^{r.\beta.q_y(i)})^{\Delta_i S^{'}_{y(0)}} = e(g,g)^{r.\beta.q_{y(0)}}, \tag{7}$$

where i = index(z), S'y ={index(z): z in Sy}.

If the tree is satisfied by S then we set

$$A_i = DecryptNode(C_2, SK_S, y_i) = e(g,g)^{r.\beta.q_{y_i}(0)} = e(g,g)^{r.\beta.S_i}, (i = 1,2,...,k) \tag{8}$$

Next, e(g,g)αSi can be computed as follows

$$F_i = \frac{e(C_i^1, D)}{A_i} = \frac{e(g^{S_i}.g^{\alpha}.g^{\beta r})}{e(g,g)^{r.\beta.S_i}} = e(g,g)^{\alpha S_i} \tag{9}$$

The decryption time of CP-ABE depends on the number of attributes a user possesses. Here, in G-CP-ABE the number of attributes assigned to a user is less since the access tree is created using POSET group structure. Therefore, G-CP-ABE has a comparatively better decryption performance as discussed in section 6.

Based on the hierarchical structure, a node can determine the keys of successor nodes (children), but the reverse is not true. If S_i is the current node then it determines $e(g,g)^{\alpha Si}$, $e(g,g)^{\alpha Si+1}$,...,$e(g,g)^{\alpha Si+m}$ sequentially and obtains the keys $\{k_i,k_{i+1},...,k_m\}$. Now the intended group user can decrypt and view attributes $\{A_i, A_{i+1},..., A_m\}$in the key table.

$$\frac{C_i^1}{F_i} = \frac{k_i.e(g,g)^{\alpha S_i}}{e(g,g)^{\alpha S_i}} = k_i, i = 1,2,...,m. \tag{10}$$

C. *Emergency Access*

The emergency situation is handled by adding a default group policy as shown in Fig.4.

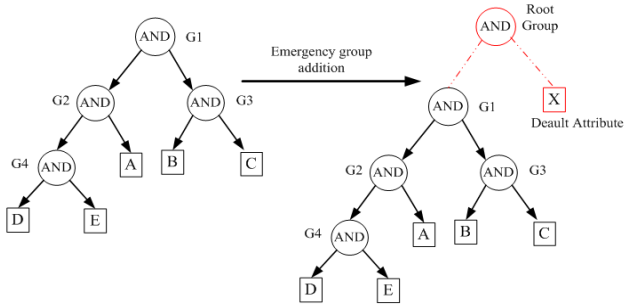

Figure 4. Access tree update for an emergency situation

A dummy attribute is added and the emergency group is made as the root group of the access tree. Now, the root group has the right to view all attributes. The root group is created with a time parameter and is invalid after the period. Then, the root is removed and the original access tree is retained. The modified key table is shown in Table III.

TABLE III. MODIFIED KEY TABLE K[1]

Root Group	G_1	G_2	G_3	G_4
$\{\{k_1\},\{k_2\},k_3\}$	$\{\{k_1\},\{k_2\},k_3\}$	$\{\{k_1\},k_3\}$	$\{k_2\}$	$\{k_1\}$

The group creation and management are done by the data owner itself as part of the access structure creation. We assume simple group management since collusion attacks are prevented by CP-ABE [9].

V. SECURITY ANALYSIS

The proposed scheme ensures privacy and better access control over the medical records outsourced to the cloud. Each user can decrypt the records if and only if their attributes match with the access policy embedded in the second ciphertext. We assume that end-to-end communications are securely encrypted via SSL, TSL or any other security protocols.

Theorem1 The proposed EMR access control policy is secure in the proposed group access policy security model under Decisional Bilinear Diffie-Hellman (DBDH) assumption.

Proof: The proposed EMR access control policy is group based, and secret key of the user depends on the group attribute set. The access control policy is embedded in the encrypted key table like basic CP-ABE scheme. The model has reduced the number of attributes and thereby the length of CP-ABE secret keys. The proof follows from [9],

[15]since the proposed model G-CP-ABE is an improved version of CP-ABE scheme [9].

We denote the security model of CP-ABE as $Ɛ_1$= *(Setup*$_1$*, KeyGen*$_1$*, Encrypt*$_1$*, Decrypt*$_1$*)* and the security model of our model G-CP-ABE as $Ɛ_2$= *(Setup*$_2$*, KeyGen*$_2$*, Encrypt*$_2$*, Decrypt*$_2$*)*. Now, we will show that security model $Ɛ_2$ can be reduced to $Ɛ_1$, which provides a proof for our claim.

Suppose that there exists an adversary Á who tries to attack the group based access policy in $Ɛ_2$ with non-negligible advantage ϵ. We build a simulation program Ò that can run the security model for CP-ABE under DBDH assumption. We assume that Ò can break the security model of $Ɛ_1$ with an advantage of ϵ. We assume that universal set of attributes *U* is defined. The design of Ò is as follows:

- **Initiation**: The adversary Á outputs the group attribute set *Y* that it wants to be challenged upon. Adversary Á forwards the attribute set *Y* to the simulator Ò of security model $Ɛ_1$. Now, Ò forwards it to the challenger of $Ɛ_1$.
- **Setup**: The challenger of $Ɛ_1$ sets the public parameter by using equation (1) and forwards it to Á.
- **Phase 1**: The adversary Á generates repeated requests for the secret keys for an attribute set S such that S does not satisfy the challenge attribute set *Y,* with the advantage of ϵ and forwards S to Ò. Because the secret keys of both security models are same, Ò passes S to the challenger of $Ɛ_1$. Challenger of $Ɛ_1$ finds the secret key by running polynomial function for nodes in the access tree using equation (3). Now, Ò gets secret key SK_S from the challenger of $Ɛ_1$ and forwards it to Á.
- **Challenge**: The adversary Á submits two messages M_1 and M_2 where $|M_1| = |M_2|$. The simulator Ò sends them to the challenger $Ɛ_1$. The challenger $Ɛ_1$ flips a random coin *b* and encrypts M_b and gives the ciphertext CT^* to Ò. Obviously, CT^* is a challenge ciphertext of $Ɛ_2$ and Ò forwards it to Á.
- **Phase 2**: The simulator acts same as in Phase 1.
- **Guess**: The adversary Á outputs guess bit $b' \in \{0,1\}$ of *b*. The adversary Ò also outputs its guess bit as b' *(*Ò*)* = b' *(*Á*)*.

Thus, if Á can break the security of access control system of G-CP-ABE with non-negligible attack ϵ, then Ò can also break the security of CP-ABE with the same probability. Hence, the security of $Ɛ_2$ follows from $Ɛ_1$.

VI. Experimental Analysis

To validate the security analysis, we implemented G-CP-ABE using CP-ABE toolkit (Stanford University) and Java Pairing-Based Cryptography library. The experiment used Type A pairing with 1024-bit discrete log security [9]. The EMR table is encrypted using Advanced Encryption Standard (AES) and the ABE ciphertext is the encrypted AES keys. We performed our experiments on a 3.60 GHz Intel Core i7 processor running Ubuntu 14.04 with 14 GB RAM.

During the initial phase of simulation, the model created a POSET based access structure and grouped the attributes accordingly (see Fig. 2 and 3). The grouping of attributes creates a sort of logical vertical partitioning of the database. Each logical partition is encrypted using a random symmetric key. Instead of using a single symmetric key, G-CP-ABE uses multiple keys to encrypt each logical fragment of the database providing more security. The encryption time of CP-ABE is proportional to the number of leaf nodes in the access tree.

The numbers of attributes used in the simulation are {5, 10, 15, 20, 25, 30, 35, 40, 45, and 50}. Also, the experiment is carried out by varying the number of records in the EMR table. Our access structure has only AND gates ensuring every attribute and hence leaf node participates in the policy creation.

The keys used for AES encryption are stored in the key table according to the group access structure (see Table II). The key table is then encrypted using CP-ABE scheme. The key generation time is measured against the number of subsets in the key table for the groups created. The results shown in Fig. 5 indicate that key generation time is linear to the number of subsets in the key structure for the groups. The proposed method has less key generation time due to the reduced leaf nodes in the modified access tree.

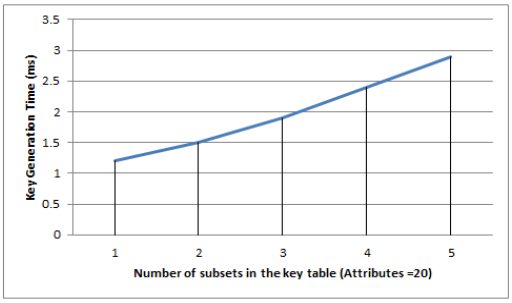

Figure 5. Key generation time versus the subsets in the key table

The access structure considers only AND gates for ensuring that all attributes participate in the access policy creation. The effect of levels in the access tree on the encryption time is measured and the result is shown in Fig. 6 indicates that encryption time linearly increases with the number of levels in the access tree.

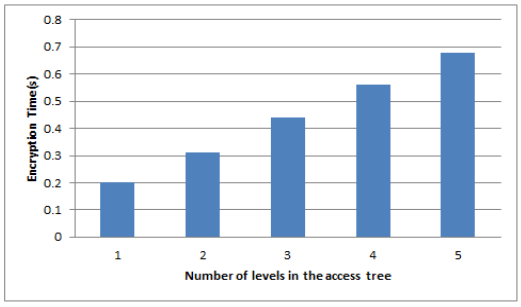

Figure 6. Encryption time versus the depth of the access tree

The proposed model reduces the depth of the access tree by using POSET based group structure which significantly reduced the overall encryption time. Simulation is carried out with a varying number of records and the overall encryption time increases linearly with increase in record number. The results are shown in Fig. 7. The original access tree is optimized using POSET based group access structure. The optimization process has reduced the number of leaf

nodes in the access tree leading to improved key generation, encryption and decryption performance.

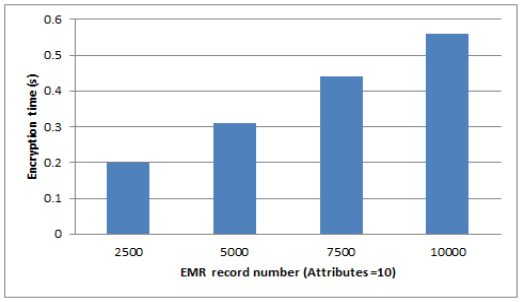

Figure 7. Encryption time versus the number of EMR records

We evaluate and compare the performance of G-CP-ABE with that of fine-grained access control schemes [24-25]. The simulation is carried out with varying the number of attributes, and the experiment is repeated ten times for each access policy. A fine-grained access control system for the e-healthcare cloud using CP-ABE is proposed in [24]. Authors adapt CP-ABE to embed access control in the ciphertext itself. A secure framework for access control in mobile cloud computing using CP-ABE is recommended in [25]. Since the schemes proposed in [24-25] use CP-ABE to provide fine-grained access control, we compare our method with proposals given in [24-25].

Figure 8 shows that overall encryption time of G-CP-ABE is less compared to that of the fine-grained access control schemes proposed in [24-25]. The experiment is carried out by varying the number of attributes. The encryption time increases linearly with increase in the number of attributes in [24-25]. But, in the proposed G-CP-ABE scheme, the increase is less than linear because of the modified access tree and the merging of CP-ABE and AES symmetric encryption. The depth of the access tree has significantly reduced compared with that of [24-25]. Also, the G-CP-ABE employs AES encryption to encrypt the EMR tables, whereas the schemes in [24-25] use CP-ABE to encrypt the EMR tables. Symmetric AES encryption is faster than CP-ABE, and the access policy is embedded in the ciphertext of the key table using CP-ABE. The size of the key table is very less as compared to EMR table. Therefore, G-CP-ABE achieves faster encryption time compared with that of [24-25].

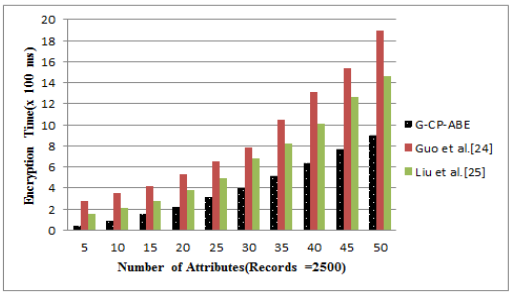

Figure 8. Encryption time versus the number of attributes

The decryption time usually depends on the number of attributes a user possesses. In the case of G-CP-ABE, only the key table is encrypted using the CP-ABE scheme. Also, the number of attributes is less as the model uses modified access tree. A group has a set of authenticated attributes, but the secret key of a group depends on the subset structure of the key table. The subsets in the key table are less compared to that of the original attribute assignment. Hence, the overall decryption time is less compared to that of the methods [24-25] and is shown in Fig. 9. From our results, we found a faster retrieval of health records from the cloud.

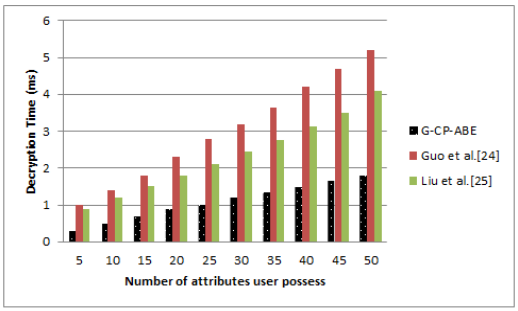

Figure 9. Decryption time versus the number of attributes

The main advantage of the proposed scheme is reduced computational overhead as shown in Fig.10. The proposed model G-CP-ABE significantly reduces the expensive bilinear pairings. The decryption time of CP-ABE depends on the number user attributes. Since G-CP-ABE uses POSET based group access structure, the number of user attribute is less compared to that of CP-ABE. Also, G-CP-ABE merges symmetric encryption and CP-ABE. The above benefits lead to the reduction in overall computation time.

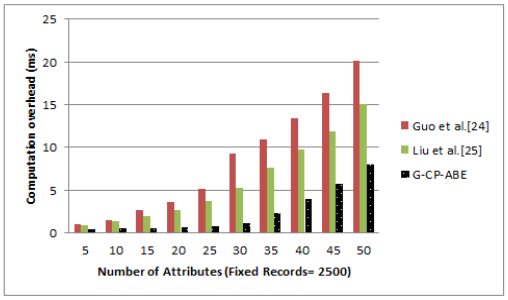

Figure 10. Computation overhead versus the number of attributes

The computational overhead almost doubles as the number of attributes increases as shown in Fig.10. However, the increase of overhead is very less in G-CP_ABE compared to that of access control schemes [24-25]. The reason is POSET based group access structure has reduced the number of leaf nodes in the access tree.

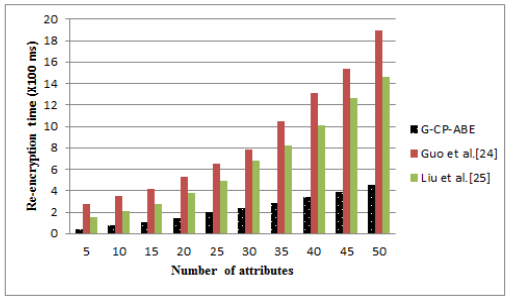

Figure 11. Re-encryption time versus number of attribute

The access control system G-CP-ABE is computationally effective since it handles the scenario of emergency group addition with minimum overhead. The access tree is modified by making the emergency group as modified root group. The key table is maintained as a dynamic list and is also modified as shown in Table III. Now, instead of re-encrypting the entire database, G-CP-ABE re-encrypts only the modified key table. As a result, G-CP-ABE has a better re-encryption time as shown in Fig. 11.

Compared to the existing works, the proposed model reduced the overall computation overhead using symmetric encryption and POSET based access structure. The number of bilinear pairing is directly proportional to the number of attributes in [24-25] whereas in our scheme it is proportional to the number of group access structure. The pairing cost is much more than the exponentiation and multiplication. Also, the proposed G-CP-ABE merges the benefits of symmetric encryption and CP-ABE encryption. In our model, ABE ciphertext is the simple key table whereas in [24-25] the ABE ciphertext is the entire EMR table. Hence, the overall encryption and decryption time is reduced enabling faster uploading and downloading of health records.

The reduced computational overhead makes G-CP-ABE suitable for health services in mobile devices. The performance of G-CP-ABE can be further improved by using lattices for group creation.

VII. Conclusion

IoT-enabled healthcare monitoring services need privacy-aware data access control in the cloud. Healthcare systems using cloud and IoT are aimed at managing EMRs in a secure and private manner offering better services to doctors, and patients. In this paper, we propose a novel fine-grained access structure G-CP-ABE with significant computation gain by integrating AES and CP-ABE encryption standards. POSET creates the access structure by grouping users and attributes are assigned to groups rather than individuals. The main advantage of G-CP-ABE is the minimal bilinear pairings compared to that of existing works. Security analysis and experimental analysis shows that the improved performance of proposed G-CP-ABE makes the scheme suitable for cloud and resource constrained IoT devices. In the future work, we will verify and validate the scope of G-CP-ABE in IoT devices. Also, we will implement G-CP-ABE in health record sharing across multiple health institutions.

References

[1] I. Ungurean, N.C. Gaitan, and V. G. Gaitan, "An IoT architecture for things from industrial environment," in In Communications (COMM), 2014 10th International Conference on, 2014, pp. 1–4. doi: 10.1109/ICComm.2014.6866713.

[2] P.M. Mell and T. Grance, "The NIST Definition of Cloud Computing," National Institute of Standards and Technology, Gaithersburg, MD, NIST SP 800-145, 2011.

[3] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud Computing: State-of-the-art and Research Challenges," J. Internet Serv. Appl., vol. 1, no. 1, pp. 7–18, May 2010. doi:10.1007/s13174-010-0007-6

[4] G. Corotinschi and V. G. Gaitan, "Smart cities become possible thanks to the Internet of Things," In System Theory, Control and Computing (ICSTCC), 19th International Conference on, IEEE, 2015, pp. 291–296. doi: 10.1109/ICSTCC.2015.7321308.

[5] Y. Lee, P. Kim, and Y. Park, "Secure Multi-Keyword Search with User/Owner-side Efficiency in the Cloud," Adv. Electr. Comput. Eng.,vol. 16, no. 2, pp. 11–18, 2016. doi:10.4316/AECE.2016.02002.

[6] J. L. Fernández-Alemán, I. C. Señor, P. Á. O. Lozoya, and A. Toval, "Security and Privacy in Electronic Health Records: A systematic literature review," J. Biomed. Inform., vol. 46, no. 3, pp. 541–562, June 2013. doi:10.1016/j.jbi.2012.12.003.

[7] A. Sahai and B. Waters, "Fuzzy Identity-Based Encryption," in Advances in Cryptology – EUROCRYPT 2005, vol. 3494, Springer Berlin Heidelberg, 2005, pp. 457–473. doi: 10.1007/11426639_27.

[8] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based Encryption for Fine-grained Access Control of Encrypted Data," In Proceedings of the 13th ACM conference on Computer and communications security,2006, pp.89 doi:10.1145/1180405.1180418.

[9] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-Policy Attribute-Based Encryption," In Security and Privacy, 2007. SP'07. IEEE Symposium on, 2007, pp. 321–334. doi: 10.1109/SP.2007.11.

[10] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing," Proceedings,IEEE2010,pp.1–9. doi:10.1109/INFCOM.2010.5462174.

[11] F. Han, J. Qin, H. Zhao, and J. Hu, "A general Transformation from KP-ABE to Searchable Encryption," Future Gener. Comput. Syst., vol. 30, pp. 107–115, Jan. 2014. doi:10.1016/j.future.2013.09.013.

[12] L. Touati and Y. Challal, "Collaborative KP-ABE for cloud-based Internet of Things applications,"In Communications(ICC), IEEE 2016, pp. 1–7. doi: 10.1109/ICC.2016.7510836.

[13] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," Proceedings of 14th ACM on Computer and communications Security,2007,pp.456-465.doi: 10.1145/1315245.1315302.

[14] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded Ciphertext Policy Attribute Based Encryption," in Automata, Languages and Programming, vol. 5126, Springer Berlin Heidelberg, 2008, pp. 579–591. doi: 10.1007/978-3-540-70583-3_47.

[15] G. Wang, Q. Liu, and J. Wu, "Hierarchical attribute-based encryption for fine-grained access control in cloud storage services," In Proceedings of 17th ACM on Computer and communications security, 2010. pp.735-737, doi: 10.1145/1866307.1866414.

[16] B. Waters, "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization," in Public Key Cryptography – PKC 2011, vol. 6571, Springer Berlin Heidelberg, 2011, pp. 53–70. doi:. 10.1007/978-3-642-19379-8_4.

[17] Fuchun Guo, Yi Mu, W. Susilo, D. S. Wong, and V. Varadharajan, "CP-ABE With Constant-Size Keys for Lightweight Devices," IEEE Trans. Inf. Forensics Secur., vol. 9, no. 5, pp. 763–771, May 2014. doi: 10.1109/TIFS.2014.2309858.

[18] 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. Journal., pp. 1–11, 2017. doi:10.1109/JSYST.2017.2667679.

[19] S. Alshehri, S. P. Radziszowski, and R. K. Raj, "Secure Access for Healthcare Data in the Cloud Using Ciphertext-Policy Attribute-Based Encryption," In Data Engineering Workshops (ICDEW), IEEE, 2012., pp. 143–146. doi: 10.1109/ICDEW.2012.68.

[20] M. Li, W. Lou, and K. Ren, "Data security and privacy in wireless body area networks," IEEE Wirel. Commun., vol. 17, no. 1, pp. 51–58, Feb. 2010. doi: 10.1109/MWC.2010.5416350.

[21] J. A. Akinyele, M. W. Pagano, M. D. Green, C. U. Lehmann, Z. N. J. Peterson, and A. D. Rubin, "Securing Electronic Medical Records using Attribute-based Encryption on Mobile Devices," Proceedings of ACM workshop on Security and privacy in smartphones and mobile devices, 2011, pp. 75-86. doi: 10.1145/2046614.2046628.

[22] B. Fabian, T. Ermakova, and P. Junghanns, "Collaborative and Secure Sharing of Healthcare Data in Multi-Clouds," Inf. Syst., vol. 48, pp. 132–150, Mar. 2015. doi:10.1016/j.is.2014.05.004.

[23] J. J. Yang, J. Q. Li, and Y. Niu, "A hybrid Solution for Privacy Preserving Medical Data Sharing in the Cloud Environment," Future Gener. Comput. Syst., vol. 43–44, pp. 74–86, Feb. 2015. doi.org/10.1016/j.future.2014.06.004.

[24] C. Guo, R. Zhuang, Y. Jie, Y. Ren, T. Wu, and K.-K. R. Choo, "Fine-grained Database Field Search Using Attribute-Based Encryption for E-Healthcare Clouds," J. Med. Syst., vol. 40, no. 11, Nov. 2016. doi:10.1007/s10916-016-0588-0.

[25] Y. Liu, Y. Zhang, J. Ling, and Z. Liu, "Secure and Fine-grained Access Control on e-healthcare Records in Mobile Cloud Computing," Future Gener. Comput. Syst., Jan. 2017. doi:10.1016/j.future.2016.12.027.

[26] E. A. Bossanyi, "Wind Turbine Control for Load Reduction," Wind Energy, vol. 6, no. 3, pp. 229–244, Jul. 2003 doi: 10.1002/we.95.

[27] R. E. Precup and S. Preitl, "Optimisation criteria in development of fuzzy controllers with dynamics," Eng. Appl. Artif. Intell., vol. 17, no. 6, pp. 661–674, Sep. 2004. doi:10.1016/j.engappai.2004.08.004

[28] M. A. Ramírez-Ortegón, V. Märgner, E. Cuevas, and R. Rojas, "An optimization for binarization methods by removing binary artifacts," Pattern Recognit. Lett., vol. 34, no. 11, pp. 1299–1306, Aug. 2013 doi: 10.1016/j.patrec.2013.04.007.

[29] S. B. Ghosn, F. Drouby, and H. M. Harmanani, "A Parallel Genetic Algorithm for the Open-Shop Scheduling Problem Using Deterministic and Random Moves," Int. J. Artif. Intell., vol. 14, no. 1, pp. 130–144, 2016.

[30] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," in Advances in Cryptology — CRYPTO 2001, vol. 2139, Springer Berlin Heidelberg, 2001, pp. 213–229. doi:10.1007/3-540-44647-8_13.