Journal of Excellence in Computer Science and Engineering, Vol. 4(2) 2018, pp. 35-41

RESEARCH ARTICLE

Enriching Controlled Information Sharing in Healthcare Systems using Attribute based Encryption with Break-Glass Policy

*S.P Predeep Kumar[1]

[1]Information Technology, NIMS, Sharjah, UAE.

Received- 31 January 2018, Revised- 26 April 2018, Accepted- 10 May 2018, Published- 14 May 2018

ABSTRACT

During emergency situations in health care, information sharing is the most crucial task in emergency management. This paper proposes a control model that accesses the information sharing in controlled manner during critical situations. This single organization can be upgraded to cloud computing environment. But cloud computing environment faces some security issues like replay attacks, data modification etc. that are to be handled properly. To avoid these issues, Attribute Based Encryption (ABE) scheme is used. ABE encrypts the data on the basis of user attribute. The decryption of ciphertext can be done if the user key attribute coincides with the ciphertext attribute. Collision resistance is one of the main features of ABE. In order to support break-glass policies, this prototype can be extended by allowing the unauthorized users to gain access in emergency situations and then revoking after emergency.

Keywords: Access control, privacy, Cloud computing, Break glass policy, Attribute based encryption.

1. INTRODUCTION

In the last few decades, many terrorist attacks, accidents, natural calamities like floods, earthquakes, hurricanes etc. demands the need for an efficient emergency management. Timing and flexible information sharing play a vital role for an emergency management system. The main area that draws an attention in information sharing during emergency management is healthcare systems. Information sharing in controlled manner for health care system is considered as the main focus of this project. During emergency, emergency policies are triggered and are executed instead of regular policies. In precise, each emergency event consists of one or more policy templates, which describes the rights that are to be followed during critical emergency situations. Based on certain laws and regulations, the emergency policies are experimented by the experts, and the reports generated during the emergency preparedness phase are tested by the emergency managers through a risk assessment study [1, 2]. Cloud storage is becoming more popular to move this emergency management system to cloud environment. Cloud computing is considered as the central storage area where organization can share their information. Though moving the data into cloud is flexible, it also raises several security issues such as overflowing data, data interruption etc. The solution for this problem is to encrypt the data before it is passed to the source output. The ways to encrypt the file as well as the kind of users who can get access to the file are decided by the health record owner. The patient health record can be made available to the user provided with the decryption key, and hence confidentiality and integrity can be achieved easily. Rights have been given to them to provide as well as revoke access when it is necessary [3]. It is important to protect the data

*Corresponding author. Telephone: +919442761492

Email address: sppredeep@gmail.com (S.P.P.Kumar)

Double blind peer review under responsibility of DJ Publications

https://dx.doi.org/10.18831/djcse.in/2018021004

stored on the server [4]. For this purpose, ABE is proposed. On the basis of user or data attribute access policies, the user can send the data to a group of users without knowing the full list of complete users and hence ABE is used in dynamic environments such as grid computing and health care area. ABE faces challenges in providing an important feature such as break-glass control system, i.e., they are privileged accounts that are not assigned to the user according to the user role. However, the user can obtain the account password if the need arises. User-based access control policies and centralized policy decisions points are used to implement break-glass access control in a system. Also break glass policy is used in a situation when a person needs immediate access privilege to particular information. This paper proposes a scheme which integrates break-glass policy with ABE.

This model proposes fine-grained access control with granularity in attribute-level. There are many models (obtained from ABAC or the XACML standard) designed on the basis of fine-grained access control [5]. One of the most important models which supports fine-grained access control in a healthcare domain is Context aware Term Based Access Control (CTMAC) presented in [6]. It integrates contextual information and it also supports team-based access control. Role Based Access Control (RBAC)-A model can also be replaced by the above said model. Also in case of emergency detection, Complex Event Processing (CEP) technology should not be used to control the access system.

1.1. ABE for fine-grained data access control

To analyze the fine grained access control, ABE technique is commonly used to produce source output [7-9]. This technique can also be used to protect Electronic Health Records (EHRs), but in this broadcast, variant of CP (Ciphertext Policy)-ABE [10] can be used to encrypt the file. Also based on the number of unrevoked users, the length of cipher text grows linearly.

1.2. Revocable ABE

It is considered as a challenging one to efficiently revoke users/attribute effectively. Normally it is a less efficient technique because it is done mainly by broadcasting periodic key which are updated to unrevoked users [11]. Hence backward/forward security cannot be achieved easily, thus resulting in less efficiency. In recent trends, to overcome these, two CP-ABE schemes can be proposed. These schemes have the capability to provide immediate revocation when compared to periodical revocation. But for Multi-Authority ABE (MA-ABE), this is not designed yet.

The remaining of the paper is organized as follows. Sections 2 and 3 present an overview of the model and ABE respectively. Section 4 presents ABE with break-glass. Section 5 presents prototype implementation, and section 7 concludes the paper.

2. INFORMATION SHARING IN EMERGENCY SITUATION

To improve the information sharing during emergency situation, two requirements must be satisfied, i.e. the system must provide user to access resources that are not normally authorized, and the other one is the action need to be taken in order to manage the emergency. For this purpose, emergency policies are supported by the model and these include the connection of Emergency and Control Policies (EACP). CEP system [12] consists of events which can be used to specify the emergencies and is considered as one of the main characteristics of this model. Core Event Specification Language (CESL) is used to describe the beginning and ending of events during emergency situation.

2.1. Emergency description

Tuple emergencies (init, end, time-out and identifier) are considered. The first and second i.e. init and end emergency events are described in CESL, whereas the init describes the triggering of the emergency and end is the event used to turn off the emergency and is considered as an optional one. The third one time-out describes the time taken by the event to expire and the last one identifier describes the scheme of both event init as well as end. The identifier plays an important role since it establishes the connection between init and end events. If the number of emergencies and its associated policies are large and if the granularity level of the policy is high, then this model will be considered to be a challenging one. [13] proposes how this works in such a challenging situation. This is not considered as a typical domain to manage emergency schemes but we have decided to select this it is provides opportunity to handle tedious examples of emergency policies.

2.1.1. Reference scenario model

Assume a hospitalized patient, where treatment is given to the patient using these structures by means of specialized equipment. This equipment is used to monitor the real time view of patient vital signs. The monitor collects the data from the monitoring equipment and thus the emergency situation is automatically detected. Let us consider an example to describe the heart rate emergency. The heart rate of the patient is given to the monitoring system by the sensor in the vital signs stream of tuples (heart rate…patient i) for every 30 seconds. Heart rate emergency can be described as,

```
HeartRateEmergency
    init: VS1 v1;
   VS1 σ(heartrate > 100)(VitalSigns);
     end: VS2 v2;
   VS2 σ(heartrate<60)(VitalSigns);
     timeout: ∞;
     identifier: patient_id;
   end;
```

The above description is explained as follows: If the heart rate of the patient is greater than 100 beats, then it indicates the initiation of emergency and this ends when the heart rate becomes less than or equal to 0 beat. Consider for patient 1, if the heart rate emergency is detected, the emergency instances are created as follows.

```
HeartRate Emergency Instance1
     emg: Heartrate Emergency;
          identifier: 1;
```

The above instance explains that if the heart rate ends for patient 1, then immediately HeartRate Emergency Instance 1 will be deleted. This proposes that information is shared in controlled manner by means of emergency policy. Different emergency access control policies are required by different instances of the same emergency case. In addition, it is also associated with an emergency template. Triggering of the emergency template is performed if an emergency is identified.

2.2. Eacptemplate

Consider an eacp (emergency access control policy) template which consists of tuple (sbj, obj, priv, ctx, obl): In this, ctx is expressed in Boolean and it defines the context, sbj and obj define the subject and object specifications respectively, obl is specified as obligation and privilege is expressed as priv. If the expression ctx becomes true, then resource identified by object specification exercise privilege by the authorized users, identified by the subject specification sbj. At this situation, if the condition obl is not equal to zero, then it indicates that some set of actions needs to be fulfilled when an authorized user exercise priv on obj [14].

A high level definition related to subject/object specification and context condition is developed, similar to attribute-centric RBAC-A [15] based on role and Attribute-Based Access Control (ABAC). In this model, it is necessary to specify the role as well as the attribute based condition in order to identify the user. Hence the tuples in this model are expressed in pair. Here sbj is expressed as (roles, cond), where the first one denotes the set of authorized roles performed and the second is a condition based on the attribute of user profile. An object specification is expressed as a pair (object, cond), where object indicates a target object and cond is a condition based on the attribute of the object. The context is designed as a set C of pairs (att, val), where att is a context attribute (e.g., time, location, session information, and so on) and val is the corresponding value.

2.2.1. Model

Assume the heart rate given in model 2.1.1 during emergency. If the health record (HR) of a patient (object condition) is accessed, it should be informed to the subject who takes care of the patient through email. If the subject does not give any authorized access to health record of the patient, then immediately it is informed to the corresponding patient through email (obligation). To achieve this, eacp template defines an attribute named as att.

```
HeartRatePolicy
sbj: (paramedic, param_id = call.param_id);
obj: (HR patient_id = emg.patient_id);
priv: read;
ctx: -;
obl: mailto(patient_mail);
```

end;

cond is expressed in Boolean of the form α, β and θ, where α indicates the attribute of user profile (object, respectively), θ is used as a matching operator in (≥, ≤) and β is a value of constant or an attribute expressed as att.

3. ROLE OF ABE IN CLOUD COMPUTING

In cloud computing, identity based encryption is mostly used but it has the disadvantage that it does not provide secured data. [16]. Also, cloud computing is mainly used to store and access the data. When symmetric and asymmetric schemes are used, it may also produce some control issues. Hence to avoid these limitations, ABE can be used as an encryption technique [17]. There are different types of model in cloud computing (private, public, community and hybrid cloud). In private cloud computing, single user is used, whereas in public, many consumers are used. If same kind of consumers uses this service, then community cloud is used. Hybrid model uses any two of the above model [18]. The most two important target of this ABE framework is to provide secure health care data and efficient key management. By including some of the parameters, this scheme eliminates some of the computation tasks. According to the requirements of different users' data, it is the responsibility of this architecture to divide the system into multiple security domains as public (PUD) and personal domains (PSD). The users belonging to PUD perform access based on professional roles such as doctors, nurses, and medical researchers. In addition to it, the PUDs are connected to the independent sector. In practice, a PUD can be mapped to an independent sector in the society. In PSD, each user consists of owner to access the data on the basis of rights given to them. In these two security domains, it is necessary to utilize ABE in order to access the cryptographically enforced health care data. PUD consists of MA-ABE that consists of multiple Attribute Authorities (AAs). Each attribute has the capability to govern disjoint subset of attributes. Role attribute defines the professional role of each public user and is defined for all PUDs. Without interacting with the owner, PUD user can obtain secret keys from AAs.

During encryption, role-based fine-grained access policies are specified by the owners in order to control the access by the public user. Also for encryption, complete list of authorized users is not required. This reduces the management overhead of user and the owner. In both these domains, data owner is considered as the trusted authority and s/he uses Key-Policy Attribute Based Encryption (KP-ABE) system in order to manage the keys as well as the right to access the user. In PSD, the intrinsic properties of the data must be defined in a data attribute. For this purpose, each file is associated with data attribute and the key size specifies the number of files to be accessed. In PSD, the number of user is small and hence it degrades the burden of the owner. The main advantage of ABE is that even if the data is stored in trusted server, it can be accessed only by the authorized users. Hence it is considered as a protective one.

4. BREAK-GLASS ACCESS POLICY

Break control policy enables subjects to override or break the control policies to access the information system in a controlled manner [19]. When a patient is unable to change his policy, then medical staff should access the patient temporarily. For this purpose, medical staff should have temporary authorization i.e. emergency key to decrypt the data. In this framework, it is normally done by moving the patient to department of emergency attribute and then moved into PSD of cipher text where the access is done by means of break glass policies. In this break glass policy, first the emergency key sk_{EM} is generated by means of single node key-policy and then delegating to the ED, where the data is stored in a database of patient directory. In case of emergency, medical staff needs to authenticate to the ED in order to obtain the emergency key and finally decrypt it. If the patient is recovering from this emergency situation, then the break glass policy can be revoked through computing rekey: rk_{EM}, and then finally it is submitted to the ED and the server for updating sk_{EM} and cipher text to the newest versions.

5. PROTOTYPE IMPLEMENTATION

In this prototype, many numbers of SDs, owners, AAs, and users are involved. Due to this, two ABE systems are used which consists of PSD and PUD. In PSD, YWRL's revocable KP-ABE scheme [20] is proposed. In PUD, revocable MA-

ABE scheme is proposed. This protocol framework is explained in figure 1. Read and write access is termed as data readers and contributors.

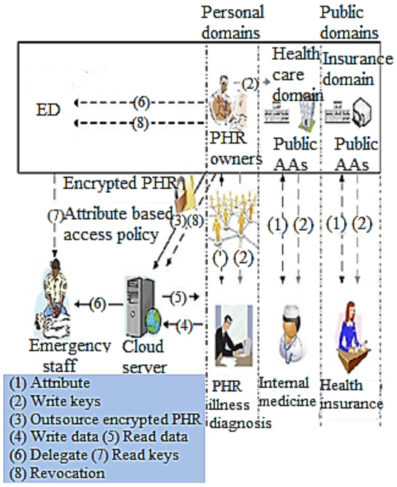

Figure 1.The proposed framework for patient-centric, secure and scalable PHR sharing on semi-trusted storage under multi owner settings

5.1. System setup and key distribution

Basic data attributes such as basic profile, medical history, allergies and prescriptions are shared by PSD in a system. In addition to data attribute, emergency attribute can also be defined in break-glass policy access. Public and master keys are generated for each and every owner client application. Pubic keys are mainly distributed through user profile in an online Healthcare Social-Network (HSN) e.g., the Indivo system [21]. Commonly two services are used to distribute the secret keys. In PHR service, access privilege of the data reader is specified by the owner in the PSD, and later the application is generated and its key is distributed. In the second service, first the PSD sends a request to the owner through HSN to obtain the secret key. In a short interval of time, the user provides the subset of data type. Depending on this, the access structure is derived by the application and it runs the keygen of KP-ABE, and thus secret keys are generated which includes the access structure.

In case of PSD, role attributes are defined by the system, and thus secret key is obtained from AAs by the reader (1, 2). In some cases there exist multiple AAs which govern different role attributes. This is illustrated in figure 1. Data encryption is done by means of MA-ABE. Contributors in PUD can write to patients health record, if write keys are distributed by AAs.

5.2. PHR encryption and access

The encrypted Personal Health Record (PHR) files are uploaded to the server by the owner (3). In PUD, encryption of PHR file is done under fine grained and role based policies whereas in PSD, it is done under a selected set of data attribute. Decryption of PHR files is done by authorized server. PHR maintains the privacy of the patient. Hence it should be encrypted before producing source output [22]. Only if the data reader has suitable attribute based keys (5), it can decrypt by downloading the file from the server. Write access will be granted by data contributor to PHR, if suitable write keys are presented (4).

5.3. User revocation

Revocation of a data or attributes is done in several possible ways which are described as follows:

- In PUD user, revocation is mainly based on role attributes;
- Revoking user's attributes is equivalent to revocation of a PUD. This is done mainly by AA and is given to the server, so that efficiency can be improved (8).
- In PSD, revocation is based on user's access privileges, and it can be done by means of PHR owner's client application.

5.4. Policy updates

In PHR document, the sharing policy is updated through updating the attributes (or access policy) in the cipher text. The operations that support this update are as follows: add/ delete/modify, which are mainly done by the server.

5.5. Break-glass

The regular access policies will not be applicable for long time in an emergency situation. In such a situation, break glass policy can be used to access PHR. In this prototype, access right is given to an Emergency Department (ED, (6)). In case of emergency situation, the staffs contact ED to obtain temporary keys to verify the identity. If the emergency situation ends, then immediately the patient can revoke from it.

Consider an example to describe the working of this framework. If the owner Alice is assumed to be a patient in the hospital, then first she should create a file F1. After the creation of file, it should be encrypted. Encryption is done under YWRL KP-ABE and revocable MA-ABE which are briefly explained as follows.

Assume P1: = ‘‘(profession = physician) ^ (specialty = internal Medicine) ^ (organization = hospital A)’’

After encryption process, the break-glass key is given to the ED by Alice. After sending the key, Alice searches for the user access rights in PUD either online or offline. Consider an example where Bob send request with labels {personal info} or {medical history} in order to access the file. If this request is approved by Alice, then it immediately sends the secret key with structure {personal info v medical history} to Bob. By using “medical history” attribute, Bob can also access and decrypt another file F2 with labels “PHR—medical history—medications”. Let us consider a user Charlie. He is a physician and is concentrated mainly on the internal medicine in the PUD. By using American Medical Association (AMA), the American Board of Medical Specialties (ABMS) and the American Hospital Association (AHA), the user obtains the secret key but cannot able to decrypt it, since his attribute of role cannot able to satisfy the policy. Hence another staff, Dorothy can access it by obtaining the break glass key from ED temporarily. Dorothy can gain access because of the emergency attribute present in that key.

7. CONCLUSIONS

The emergency control model can also be extended using administrative policies and security is provided by ABE in cloud computing environment. This prototype can also be extended to use break glass policy and hence violations can be traced. Encryption can be done mainly by ABE, so that patient can access both in PSD and PUD. By evaluating the time in administrative policies, this prototype can also be extended to little brief in future. Many risk assessment tools [23] have been analyzed to extract policies obligation of emergency from its scenario.

REFERENCES

[1] T.F.E.M.A (FEMA), Emergency Response Plan Implementation, 2012.

[2] S.Umamaheswaran, K.Senthil and R.Rajaram, An Algorithm for Encrypting/Decrypting Textual Messages, Journal of Excellence in Computer Science and Engineering, Vol. 1, No. 1, 2015, pp. 1-10, http://dx.doi.org/10.18831/djcse.in/2015011001.

[3] K.D.Mandl, P.Szolovits and I.S.Kohane, Public Standards and Patients Control: How to Keep Electronic Medical Records Accessible but Private, BMJ, Vol. 322, No. 7281, 2001, pp. 283-287, https://dx.doi.org/10.1136/bmj.322.7281.283.

[4] C.Murugamani, Authenticating and Securing Ad-Hoc Networks using Gateway Selection Algorithm, Journal of Excellence in Computer Science and Engineering, Vol. 3, No. 2, 2017, pp. 27-33, http://dx.doi.org/10.18831/djcse.in/2017021003.

[5] H.L.Bill Parducci, eXtensible Access Control Markup Language (XACML) Specification 3.0, 2010.

[6] C.K.Georgiadis, I.Mavridis, G.Pangalos, and R.K.Thomas, Flexible Team-Based Access Control using Contexts, Proceedings ACM symposium on Access Control Models and Technologies, USA, 2001, pp. 21-27, https://dx.doi.org/10.1145/373256.373259.

[7] A.Boldyreva, V.Goyal and V.Kumar, Identity-Based Encryption with Efficient Revocation, Proceedings ACM Conference on Computer and Communication Security, USA, 2008, pp. 417-426, https://dx.doi.org/10.1145/1455770.1455823.

[8] S.Yu, C.Wang, K.Ren and W.Lou, Achieving Secure, Scalable and Fine-Grained Data Access Control in Cloud Computing, Proceedings IEEE on INFOCOM, USA, 2010, https://dx.doi.org/10.1109/INFCOM.2010

.5462174.

[9] S.Yu, C.Wang, K.Ren and W.Lou, Attribute Based Data Sharing with Attribute Revocation, Proceedings ACM Symposium on Information, Computer and Communication Security, China, 2010, https://dx.doi.org/10.1145/1755688.1755720.

[10] S.Narayan, M.Gagne and R.Safavi-Naini, Privacy Preserving EHR System using Attribute-Based Infrastructure, Proceedings ACM Cloud Computing Security Workshop, USA, 2010, pp. 47-52, https://dx.doi.org/10.1145/1866835.1866845.

[11] X.Liang, R.Lu, X.Lin and X.S.Shen, Ciphertext Policy Attribute Based Encryption with Efficient Revocation, University of Waterloo, Canada, 2010, pp. 1-9.

[12] A.Margara and G.Cugola, Processing Flows of Information: From Data Stream to Complex Event Processing, ACM Computing Surveys, USA, Vol. 44, No. 3, 2012, pp. 15, https://dx.doi.org/10.1145/2187671.2187677.

[13] B.Carminati, E.Ferrari and M.Guglielmi, Secure Information Sharing on Support of Emergency Management, IEEE International Conference Privacy, Security, Risk and Trust and IEEE International Conference on Social Computing, USA, 2011, pp. 988-995, https://dx.doi.org/10.1109/PASSAT/SocialCom.2011.69.

[14] Barbara Carminati, Elena Ferrari and Michele Guglielmi, A System for Timely and Controlled Information Sharing in Emergency Situations, IEEE Transaction on Dependable and Secure Computing, Vol. 10, No. 3, 2013, pp. 129-142, https://dx.doi.org/10.1109/TDSC.2013.11.

[15] D.R.Kuhn, E.J.Coyne and T.R.Weil, Adding Attributes to Role-Based Access Control, Computer, Vol. 43, No. 6, 2010, pp. 79-81.

[16] G.Spyra, W.J.Buchanan and E.Ekonomou, Sticky Policies Approach within Cloud Computing, Computers & Security, Vol. 70, 2017, pp. 366-375, https://dx.doi.org/10.1016/j.cose.2017.07.005.

[17] P.P.Kumar, P.S.Kumar and P.J.A.Alphonse, Attribute Based Encryption in Cloud Computing: A Survey, Gap Analysis, and Future Directions, Journal of Network and Computer Applications, Vol. 108, 2018, pp. 37-52, https://dx.doi.org/10.1016/j.jnca.2018.02.009.

[18] J.Li, Y.Zhang, X.Chen and Y.Xiang, Secure Attribute-Based Data Sharing for Resource-Limited Users in Cloud Computing, Computers & Security, Vol. 72, 2018, pp. 1-2, https://dx.doi.org/10.1016/j.cose.2017.08.007.

[19] S.Schefer-Wenzl and M.Strembeck, Model-Driven Specification and Enforcement of RBAC Break-Glass Policies for Process-Aware Information Systems, Information and Software Technology, Vol. 56, No. 10, 2014, 1289-1308, https://dx.doi.org/10.1016/j.infsof.2014.04.010.

[20] S.Yu, C.Wang, K.Ren and W.Lou, Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing, Proceedings IEEE on INFOCOM, USA, 2010, https://dx.doi.org/10.1109/INFCOM.2010.5462174.

[21] Indivo, 2012, http://indivohealth.org/.

[22] X.Liu, Q.Liu, T.Peng and J.Wu, Dynamic Access Policy in Cloud-Based Personal Health Record (PHR) System, Information Sciences, Vol. 379, 2017, pp. 62-81, https://dx.doi.org/10.1016/j.ins.2016.06.035.

[23] A.Lewko and B.Waters, Decentralizing Attribute-Based Encryption, Annual International Conference on the Theory and Applications of Cryptographic Techniques, Heidelberg, 2011, pp. 568-588, https://dx.doi.org/10.1007/978-3-642-20465-4_31.