International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org, editorijettcs@gmail.com
Volume 3, Issue 4, July-August 2014 **ISSN 2278-6856**

Hybrid Attribute Based Encryption

GD Makkar[1], Vivek Panwar[2]
[1,2]GRDIMT, Deptartment of CSE,
Dehradun 248001, India

Abstract

Now a day, cloud computing has become a significant technology trend. The cloud computing technology benefits include cost savings, high availability of resources, and easy scalability. The cloud computing provides many services to the users. These are Software as a Service, Platform as a Service and Infrastructure as a Service. For any enterprise, data plays a significant role in its growth. Cloud computing also provides the facility to the enterprises to store their valuable data remotely at cloud. One of the main hurdles that disallow the use of cloud computing is the security of data in the cloud as the users have no physical contact with the data. The only way to secure the data in the cloud is encrypting the data before storing it. The threat to the data stored in the cloud is not only from the intruders outside of the organization but also from the malicious users inside the cloud service provider who have inherently access to the databases. The only possible solution to protect our private and confidential data on the cloud is to encrypt it before storage. There are many schemes are proposed and used in the past ranging from Identity Based Encryption (IDE), Fuzzy Attribute Based Encryption (Fuzzy ABE), Key-Policy Attribute Based Encryption (KP-ABE) to Cipher-Policy Attribute Based Encryption (CP-ABE). One problem with KP-ABE and CP-ABE is that both use single trusted authority. Data owner is only trusted authority. It can be bottlenecked. In our approach, we used the hybrid Attribute Based Encryption (HbABE) where we combine scheme KP-ABE and CP-ABE and use the concept of access structure that defines which users can decrypt the data. In our scheme, we used two tree structures, one used as access structure and the other is used as a private key. HbABE is specifically designed for the company to be used internally purpose not for the public use. It provides the security at three levels, first, using symmetric data encryption, second KP-ABE & CP-ABE and third using the biometric identity. Its purpose is to fully secure the company's file both from outside as well as inside intruders.

Keywords: Cloud computing security, encryption, ABE, KP-ABE, CP-ABE.

1. INTRODUCTION

Cloud computing is a virtual pool of resources such as software, platform & infrastructure that is dynamically scalable and reconfigured at a very low cost to meet the need of the customer. All services of the cloud computing, such as storage, application development and access application access through Internet. The National Institute of Standards and Technology (NIST) [1] which is responsible for developing standards and guidelines for technologies defines the cloud computing as ". . . a pay-per-use model for enabling available, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." Cloud computing provides three kinds of services. These three services are Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) model called SPI model. The biggest problem with cloud computing is the security and privacy of the user data storage and management. All the user data is stored at the Cloud Service Provider's end. Although CSP takes all measures to provide better security, but still it is tough to have full faith on the CSP due to the state-of-the-art risks associated with the cloud. Virtualization which is the back bone of the cloud computing and is also a big threat to the security [2].

2. ORIGIN OF THREATS

There are three origins of threats:

- **Outsiders:** outsiders are the entities that exist outside of the organization who always have an eye on the sensitive data of the organization and try to break the security to get the access of the information server. This are the hackers who unauthorized access the user's data and misuse it.
- **Insider:** An insider is a representative of the CSP who exploit his/her position to get the customer's private data and abuse it for irreverent reason. It is continually stressing viewpoint for any enterprise that a CSP representative can have access to their valuable information and use their private information for their benefit [3].

Sometime CSP unintentionally can also be act as malicious. This insidious form of the malicious insider problem is through PaaS based services. If the service provider offers a platform that allows developers to develop an application that interact with users' data, i.e. Facebook Applications, users may unknowingly allow these developers access to all their data. For example, it is well known on the Facebook Platform that once a user adds an application, the application may have the ability to access all user information, if allowed to do so. Similarly, when a developer added his application in the Google play, store and user installed application in mobile, it unknowingly gives access to the user's private information such as phone status and identity, network access, mobile location, contacts, etc. Even if the application developers are not malicious this does not mean that the application cannot be hacked.

- **Natural:** Although both insiders and outsiders can cause errors within the infrastructure, but there are some other errors that occur naturally due to software deficiencies itself or from hardware failure. Data in the cloud are usually stored through the application. Malfunctions and error in the software interface can lead to an intruder to get inside the software and have unauthorized access of the user data. For example, a flaw in Apache allowed an attacker to gain complete control over the web server [4]. These malfunctions are exists due to the poorly designed or implemented security measures. Software interface must be made fully secure against the accidental and malicious disclosure.

3.BACKGROUND

3.1 Fuzzy Identity Based Encryption

Sahai and Waters [5] introduced the Fuzzy Identity Based Encryption in 2005. They modify the original IDE given by Adi Shamir in [6] 1984. Identity based Encryption view the identities as a string of characters while in Fuzzy Identity Based Encryption view the identities as set of descriptive attributes. It is also called Attribute Based Encryption. In this scheme, every user is identified as a set of attribute. IBE allows the user to encrypt and decrypt the message without using public key certificate while Fuzzy IBE was the first paper to give the concept of attributes based encryption with public key cryptography. It is also different from the standard IBE in such a way it it allows for one-to-many encryption in which ciphertext is not necessarily encrypted for a particular user, it may be sent to multiple users. In Fuzzy IBE, a user with a set of attributes w is able to decrypt a ciphertext encrypted with the public key w', if and only if , the identities w and w' are close to each other as measured by the "set overlap" distance metric. Let U be a universe of attributes and w be an identity having set of attributes and is a subset of U i.e. $w \subseteq U$, decrypt a ciphertext encrypted by identity w' if and only if there exists an identity overlap d between w and w' such that $|w \cap w'| \geq d$. Fuzzy IBE is an attribute type encryption where set of attributes are used to represent an identity. In this scheme, a document is encrypted with a set of attributes and these attributes are decided according to the users to whom this encrypted document to be sent. For example, the chairperson of institute wants to encrypt a message and send to the HOD computer and mechanical and hiring committee for hiring new faculties. He would encrypt to the identity {"HOD", "computer", "mechanical", hiring committee"}. Any user whose identity contains all these attribute or set of identical attributes as set during encryption could decrypt the message. The benefit of using Fuzzy IBE is that a document or message can be stored on an untrusted storage server instead of trusted server to perform the authentication check before delivering the document.

3.2 Key-Policy Attribute Based Encryption

Fuzzy IBE restricts the user to share and access the information at a fine grained level. Sahai and Waters [5] introduced the concept of Attribute Based Encryption (ABE) and tried to share the encrypted data at a fine grained level. In ABE, set of descriptive attributes is used with a user's private key to get the ciphertext. A user can decrypt the ciphertext if and only his attributes match with the attributes of ciphertext. There should be at least d attributes overlapped between a private key and ciphertext. Vipul Goyal et al [7] improves the ABE and gave the new encryption scheme Key-Policy Attribute Based Encryption (KP-ABE). In KP-ABE, during encryption, user labeled the ciphertext with a set of descriptive attributes and an access structure is associated with each private key that specifies which user can decrypt the ciphertext (figure 1). Since the access structure is associated with private key that's why it is called the Key - Policy Attribute Based Encryption (KP-ABE). Access structure can be specified in the form of tree structure where the internal nodes consist of OR and AND gates and external node represents the different attributes that's used for encryption. A user can decrypt a ciphertext only if the attributes associated with the ciphertext satisfy the access structure of the user's private key. For example, a user encrypted a message M and produces ciphertext C with a set of descriptive attributes C1 (3,5,6,7). Every user private key is associated with an access structure. A user with access structure A(1 AND 2) cannot decrypt the message M as this access structure does not satisfy the set of attributes associated with the ciphertext. Another user with access structure A(3 OR 5) can decrypt the ciphertext as this access structure satisfy the attributes of ciphertext. Similar access structures A((1 AND 2) OR (3 OR 7)) and A (3 out of (1,2,3,4,5,6,7)) qualifies the ciphertext to decrypt while the access structure A (2 out of (1,2,5)) does not qualify.

3.3 Ciphertext-Policy Attribute Based Encryption (CP-ABE)

The disadvantage of KP-ABE is that access control of the encrypted data is controlled by the set of descriptive attributes not by the access tree and also while encrypting data; the user had no control over the attributes. This reason led to the design of the cipher policy attribute based encryption (CP-ABE). In CP-ABE, user can set his own desired set of attributes while associating the access structure with the ciphertext. Another restriction with the KP-ABE is that all attributes have to be made public as they are component of the public key.

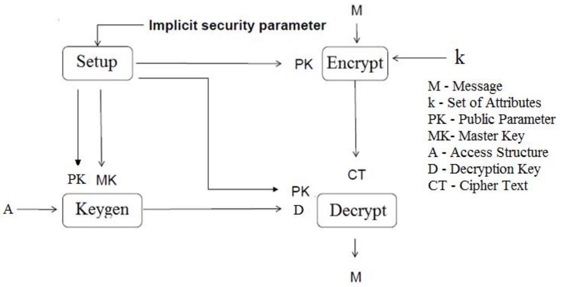

Figure 1 Key Policy Attribute Based Encryption

Bethencourt et al [8] introduced the Ciphertext-Policy Attribute Based Encryption (CP-ABE). It is also modified version of the ABE. CP-ABE is similar to the Key-Policy Attribute Based Encryption except that it works in the reverse order. In a key policy attribute based encryption, access structure is the user's private key and set of descriptive attributes are associated with the ciphertext during the encryption. To decrypt the ciphertext, the user matches his private key (access structure) with the attributes associated with the ciphertext. If it matches, then the ciphertext is decrypted otherwise not. Cipher-Policy Attribute Based Encryption (CP-ABE) works in the reverse order of KP-ABE. Instead of associating the set of attribute to the ciphertext, it associates the access structure and encrypt the message using the set of descriptive attributes (user's private key) and a user can decrypt it if and only if his private key (set of descriptive attribute) matches with the access structure associated with the ciphertext (figure 2). For example, If the access structure in encrypted data is (1 AND (2 OR 3). If a set of attributes in user's private key is (1 AND 2), then the user can decrypt the data.

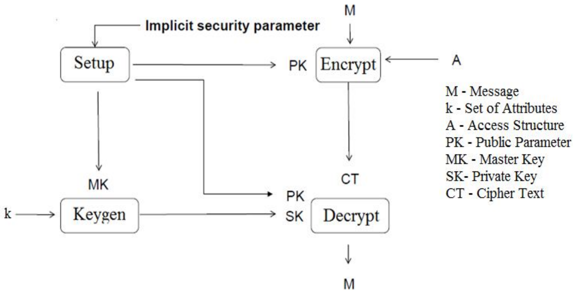

Figure 2 Cipher-Policy Attribute Based Encryption (CP-ABE)

4. HYBRID ATTRIBUTE BASED ENCRYPTION (HbABE)

Key-policy attribute based encryption (KP-ABE) [7] and cipher-policy attribute based encryption (CP-ABE) [8] are the two approaches that provides finer grained access control of encrypted data on the cloud. But KP-ABE lacks of flexibility in managing the attributes and scalability and finds a problem in dealing with multiple levels of attribute authorities and also it was not possible for Encryptor to decide who can decrypt the encrypted data. All consumers whose private key satisfies the access policy can decrypt it. As well, consumers have no choice except the trust on the key issuers. There is single trusted authority (TA) which is data owner. This scheme cannot be applied to the multiple data owners because each user would receive many keys from multiple owners. Key management is difficult in this scheme. There are some applications where KP-ABE is not suitable such as encryption in message broadcasting. For this CP-ABE found to be fully suitable where the user's private key is defined by a set of attributes and an access structure is associated with the ciphertext. A user can decrypt the message only if the private key attributes matches with the access structure associated with ciphertext during the encryption. HbABE is a hybrid approach of KP-ABE and CP-ABE. For the proper key management, we used the "A Hierarchical Attribute-Based Solution for Flexible and Scalable Access Control in Cloud Computing (HASBE) by Zhiguo Wan et al [9]. It was Wang et al [10], who firstly proposed hierarchical attribute-based encryption (HABE) to achieve fine-grained access control in cloud storage services by combining hierarchical identity-based encryption (HIBE) and CP-ABE. Zhigua Wan gave a system model for the proper distribution and management of the key Hybrid attribute based encryption is a cryptographic scheme which is ideal for the closed group of user such as executives in a company. This scheme is for the protection of the important files and data within the company so that only the authorized peoples in a company can access the data. A secure hierarchical key generation center is used for the generating and managing the keys. Each authorized user is assigned an access structure which consists of a set of attributes depicted in the form of tree structure. Headquarter of the company which is a trusted authority will be responsible for the evolution of the universal access structure of the tree. We use the similar system model as described by Zhiguo Wan et al [9] which defines a good hierarchical model for the generation and management of the key.

4.1 Key Generation and Management

Zhiguo Wan gave a system model as shown in the figure 3 [9] which define a hierarchy of authorities instead of using single trust authority as in the case of KP-ABE and CP-ABE. The trusted authority is the root authority and responsible for managing top-level domain authorities. The trusted authority will be responsible for developing a universal access tree to define various sets of attributes that can satisfy the authorized users of the company. This tree is always kept confidential with the trusted authority only.

Under the trusted authority, there are multiple domain authorities. When any user wants to encrypt a file, he/she will get the encrypted key from the domain authority and the domain authority approach the trusted authority for the key. Trusted authority generates an access structure (key) from the universal access tree structure and provides it to the domain authority which further provides to the data owner.

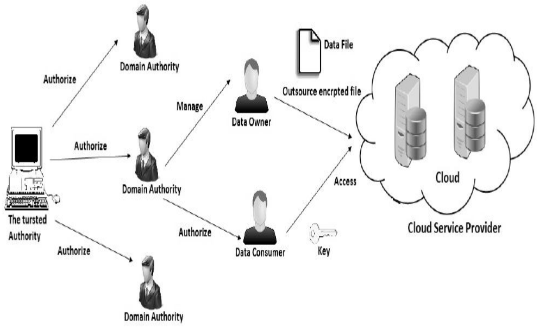

Figure 3 Hierarchical Attribute-Set-Based Encryption (HASBE)

Zhiguo Wan et al divides the hierarchical system model in to five parties that works at four levels. These five parties are:

(i) The trusted authority
(ii) Domain authority
(iii) Data owner
(iv) Data consumer
(v) Cloud service provider

4.2 Encryption and Decryption in HbABE

In our approach, we used a combination of KP-ABE and CP-ABE approaches. In KP-ABE, data is encrypted using the user's access structure (private key) and set of attributes are associated with the ciphertext. During decryption, user matches his access structure (private key) with the set of attributes with the ciphertext and if it matches then he/she can decrypt the data otherwise not.

CP-ABE uses the reverse order. User encrypts the data using the set of attributes (private key) and associates an access structure with the ciphertext. Only those users can decrypt the data whose set of attribute (private key) match with the access structure associated with the ciphertext.

We use three levels of encryptions in our approach to provide full security to the data and files of the organization.

i. Symmetric data encryption
ii. Access structure (tree based structure)
iii. Biometric authentication

4.2.1 Three levels of HbABE

In the first level, data or file is encrypted using the symmetric data encryption using the private key which is known to all authorized users of the company (figure 4).

In the second level (figure 5), we will the concept of KP-ABE and CP-ABE. Two access trees will be used for encrypting the data and no other attributes will be associated with the ciphertext. Every authorized user in the company has been assigned an access structure which also acts as his/her identity. When a user wants to encrypt the data, he/she uses his own access structure (data owner's private key) to encrypt it and also associate an access structure that he/she obtained from his domain authority.

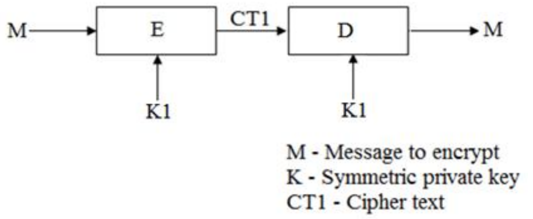

Figure 4 First level of HbABE, Symmetric encryption

Access structure is a tree structure which defines which users qualifies for the decryption of the data. When a user wants to decrypt the data, he/she will use his access structure (consumer's private key) and match it with the access structure stored with the ciphertext. If his/her private key is a part of the tree or subtree, then he/she can decrypt the data.

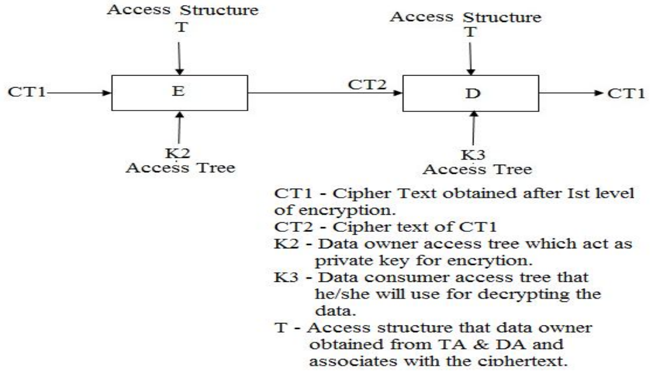
Access Structure
T
Access Structure
T
CT1
E
CT2
D
CT1
K2
Access Tree
K3
Access Tree
CT1 - Cipher Text obtained after Ist level of encryption.
CT2 - Cipher text of CT1
K2 - Data owner access tree which act as private key for encrytion.
K3 - Data consumer access tree that he/she will use for decrypting the data.
T - Access structure that data owner obtained from TA & DA and associates with the ciphertext.

Figure 5 Second level of HbABE, access structure encryption

Third level (figure 6) is authentication level. After obtaining the ciphertext CT2, data owner will apply the biometric encryption. All authorized users in the company will obtain his/her biometric identity through their DA and TA. This level is required to ensure that data is being decrypted only by the valid user of the company so that no outsider can have access to the company confidential data. Data owner now will apply his biometric identity on the ciphertext CT2. Only those users can access the CT2 who has their biometric identity available with trusted authority (TA). Since it is likewise possible to produce invalid biometric identity of a user. One way to avoid this situation, we can include user identity attribute with his/her biometric identity. If intruder able to get a user biometric identity, he cannot even access the ciphertext CT2 as biometric identity is combined with the user's attribute identity. When any user will authenticate, his/her credentials will get stored in the log file which is accessible to the trusted authority only.

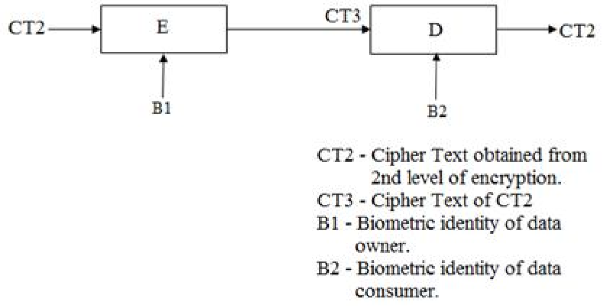
CT2
E
CT3
D
CT2
B1
B2
CT2 - Cipher Text obtained from 2nd level of encryption.
CT3 - Cipher Text of CT2
B1 - Biometric identity of data owner.
B2 - Biometric identity of data consumer.

Figure 6 Third level of HbABE, Biometric encryption for authentication

All the three levels of HbABE are shown in the figure 7.

4.2.2 Second Level of HbABE

In our construction, we use data owner access tree K2 as a private key. A consumer that wants to decrypt a message will match his access tree K3 with the access structure T associated with the ciphertext. Each internal node of the tree is threshold gate and the external node contains the attributes.

4.2.2.1 Access Structure [11]

Suppose {P1, . . . , Pn} are the set of parties i.e. attributes. A collection A2 $^{\{P1,...,Pn\}}$ is monotone if B, C : if B A and B C then C A. An access structure is [2] a collection A of non-empty subsets of {P1, . . . , Pn}, i.e., A $2^{\{P1,...,Pn\}}\backslash\{\varnothing\}$. The sets in A are called the authorized sets, and the sets not in A are called the unauthorized sets. Access tree is the one of most important part of the KP-ABE algorithm. A user encrypted a message using the access tree as a key to produce ciphertext that are labeled with access structure T. Internal nodes of the access tree consist of threshold gates, it can be AND and OR gates while the external nodes represent the attributes.

4.2.2.2 Access Tree [8]

Let T be a tree representing an access structure. Each internal node represents a threshold gate described by its children and threshold value. If num(x) is the number of children of node x and k(x) is its threshold value, then $0 \le k(x) \le num(x)$. When k(x) = 1 then the threshold gate is an OR gate and when k(x)=num(x) then it is an AND gate. Each leaf node ***x*** of the tree represents an attribute and a threshold value the leaf node is k(x) = 1.

- Num(x) = number of children of X
- K(x) = num(x) => AND gate
- K(x) = 1 => OR gate
- Leaf: k(x) = 1

4.2.2.3 Satisfying an Access Tree

Let R be a root of the access tree T. T_X represents the subtree of T rooted at node x. If a set of attributes k satisfies the access tree T_X then it is denoted as $T_X(k)=1$. If x is an internal node then evaluate $T_{X'}(k)$ for all children x' of node x. $T_X(k)$ returns 1 if and only if at least kx children returns 1. If x is a leaf node, then T_X (k) returns 1 if and only if att(x) ∈ k where att(x) is a function used only when x is a leaf node and denotes the attributes associated with the leaf node x of the tree.

Y set of attributes => Tx(Y) = 1
Tx(Y) = 1 iff at least k(x)== 1
x is a leaf node => then Tx(Y)== 1 iff att(x) ∈Y

Where att(x) denotes the attribute associated with the leaf node x.

4.2.2.4 Algorithms in second level of HbABE

Second level of Hybrid Attribute Based Encryption (HbABE) consist four algorithms: Setup, KeyGen, Encrypt, and Decrypt.

(i)Setup: This algorithm takes as input a security parameter and returns the public key PK and master secret key MK. PK is used by message senders for encryption. MK is used to generate user secret keys and is known only to the trusted and domain authority.

(ii) KeyGen (MK, K2) →(SK)

This algorithm takes as input an access tree K2 associated with the data owner and the master secret key MK. It outputs a secret key SK that enables the user to decrypt a message encrypted under an access tree structure T if and only if matches T.

(iii) Encryption (PK, CT1, T) → CT2

This algorithm takes as an input message CT1, public parameter PK, an access structure T and encrypt the message CT, produces the ciphertext CT2 such that only the user who have an access tree and it satisfies the access structure associated with the ciphertext during the encryption can only decrypt the message.

(iv) Decryption (CT2, PK, SK) → CT1

The decryption algorithm takes as an input private key SK for access control K2, public parameter PK and a cipher text CT1 which contains the access policy T. It outputs CT1 if K3 is a subset of T. Encryption and decryption at the second level is shown in the figure 8.

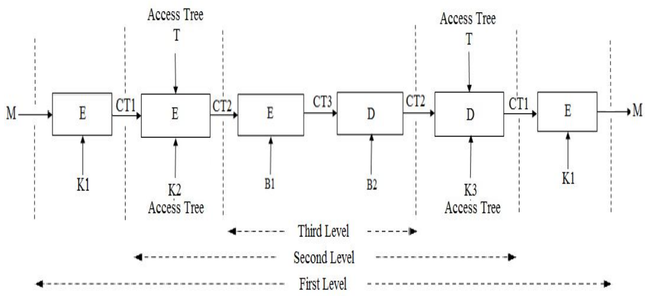

Figure 7 Three levels of HbABE Encryption

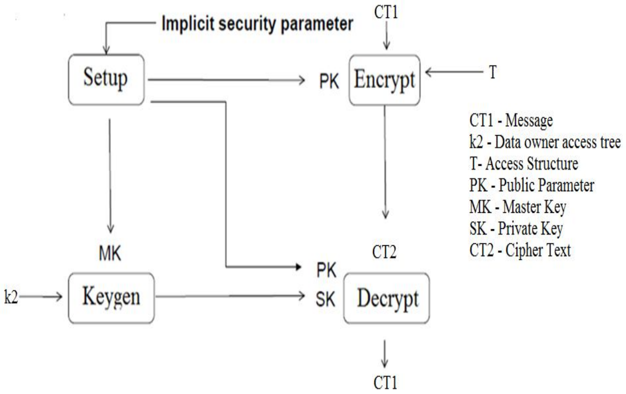

Figure 8 Encryption & Decryption in Hybrid Attribute Based Encryption

5. CONCLUSIONS

In this thesis, we present HbABE scheme which is hybrid of KP-ABE and CP-ABE. We tried to secure confidential data and files of an organization. When data are outsourced and stored in the cloud, main threat to an organization is from intruders both from outside of the their system as well as malicious users of the cloud service provider that they don't get inside of their private and confidential information. The only way to secure data is to encrypt the data before storing in the cloud. In our approach we use symmetric encryption, hybrid of encryption schemes KP-ABE and CP-ABE and biometric identity for the authentication purpose, so that only the employees of the organization have access to the encrypted file and they can decrypt it. Although out scheme HbABE does not provide so much fine access of data as in case of CP-ABE and also slow because of involvement of encryption at three levels, our aim was to fully secure the organization data from outside and inside intruders. We also used multiple authorities for implementing and managing the keys. HbABE is just an idea of securing the data; it needs to be implemented yet. It might have some problems, but every big thing starts with a little idea.

References

[1] P. Mell and T. Grance, "The NIST Definition of Cloud Computing", National Institute of Standards and Technology, Information Technology Laboratory, Technical Report Version September 2012

[2] T. Ristenpart and E. Tromer et al., "Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds", ACM Conference on

Computer and Communications Security CCS'09. Nov. 2009.

[3] Wong, P., “Conversations About the Internet #5: Anonymous Facebook Employee”, The Rumpus. Jan. 2010.url: http://therumpus.net/2010/01/conversations-aboutthe-internet-5-anonymous-facebook-employee.

[4] Ho., C., “Apache aw opens systems up to attack” ZDNet UK. Mar. 2010. url: http://www.zdnet.co.uk/news/securitythreats/2010/03/08/apache-flaw-opens-systems-up-toattack-40077943/

[5] Sahai, A., Waters, B., “Fuzzy identity based encryption,” in Proc. Advances in Cryptology—Eurocrypt, 2005, vol. 3494, LNCS, pp. 457–473.

[6] Shamir, A., “Identity Based Cryptosystems and Signature schemes”

[7] Goyal, V., Pandey, O., Sahai, A., Waters, B., “Attribute Based Encryption for Fine-Grained Access Conrol of Encrypted Data” ACM conference on Computer and Communications Security (ACM CCS), 2006.

[8] Bethencourt, J., Sahai, A., Waters, B., “Ciphertext-policy attribute based encryption,” in Proc. IEEE Symp. Security and Privacy, Oakland, CA, 2007.

[9] Wan, Z., Liu, J., Deng, R.H., “HASBE: A Hierarchical Attribute-Based Solution for Flexible and Scalable Access Control in Cloud Computing”, IEEE Transactions on Information Forensics And Security, Vol. 7, No. 2, April 2012

[10] Wang, G., Liu, Q., Wu, J., “Hierachical attibute-based encryption for fine-grained access control in cloud storage services,” in Proc. ACM Conf. Computer and Communications Security (ACM CCS), Chicago, IL, 2010.

[11] Beimel. A., “Secure schemes for Secret Sharing and Key Distribution”, PhD thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996.