Survey on Attribute Based Encryption with Attribute Hiding in Cloud Storage

D. Vaduganathan,
Master of Engineering,
Department of CSE,
Angel College of Engineering And Technology,
Tamilnadu.

S. Ramasami,
Assistant Professor,
Department of CSE,
Angel college of Engineering and Technology,
Tamilnadu.

***Abstract*– Cloud computing is mainly used for storing the large amount of data. The data can be accessed anywhere from the world. Access Control is main issue in this scenario. One of the Access Control is Identity based encryption is used to provide the security for data access from the cloud. Identity may be E-mail id or mobile number. Thus IBE system is not efficient for larger organization. Identities are not enough for the sensitive data in cloud storage. To improve this identity based encryption we go for the attribute based encryption (ABE) system. It is used to improve the security on the cloud storage. Access policy is mainly used in the Attribute based encryption. Sensitive data can be encrypted under the set of policies. The access policy is defined with the set of attributes. The keys are generated with the set of attributes which is based on who can decrypt the data. The decryption can be done if and only if the set of attributes in the key matches with the access policy attributes. That is efficient encryption and decryption in cloud storage. But privacy of data is not fully secured in the ABE system. Cipher texts are revealing some useful information about the policy and the attributes. Through this cloud server can learn the access policy and the attributes. To hide this Access policy and attributes inner product encryption system are used. The inner products are encrypted by the predicate based encryption (PE). PE system is used to improve the security of the ABE in cloud storage. This paper has the survey on what are the systems used to hide the attributes and access policy in the cloud storage**.

***Keywords– cloud computing, data sharing, scheme*.**

I. INTRODUCTION

Cloud computing is widely used in the IT industry. It provides the on-demand services and scalability. Users can use services based on the pay-as-usage basis. Cloud storage also one of the services in the cloud computing. If user wants cloud storage, they need to pay for their usage. Cloud provider will give the space to store the data in the cloud storage. Data owner can upload the data in the encryption format by the public keys. Through this cloud cannot view the plaintext. The decryption can be done by the private key. The encryption and decryption is done based on the identity of the person that is called as the identity based encryption. It is one of the public-key crypto systems. IBE system has two disadvantages. One is to encrypt the plaintext data owner that gets all the person identity. i.e., public key wants to encrypt the data in cloud storage. Second disadvantage is storage of public key that takes more space in cloud. To overcome enhance IBE system as the attribute based encryption (ABE).Attribute based encryption is used for encryption and decryption in the cloud storage. An ABE system uses the access tree structure to define the policy for who can access the data. Each cipher texts contain the set of policies.

The policy satisfaction is identified by the threshold value. The threshold value is based on the number of attributes in the cipher texts. The expected threshold value is got which can decrypt the cipher texts. Attribute based encryption can be classified into two types based on the Access policy. One is attribute based encryption with cipher texts policy (ABE-CP) and second is attribute based encryption with key policy (ABE-KP). ABE-KP system has the policies with the private keys that are called as the attribute based encryption with the key policy. Here the data owner can not define the access policies, because of the policies in the private keys. Key generator is responsible to define the keys. Data owner cannot directly define the policies for the data. Data owner should depend on the key generator. ABE-CP system has the policies in each cipher texts that are called as Attribute based encryption based on cipher texts policy. The policy is defined with the cipher texts in the form of access structure. The decryption keys have set of attributes. User can do the decryption if and only if the key's attributes are satisfied with cipher texts. ABE-CP is not efficient for large amount of users. Single authority is not efficient for the large amount of users, since performance degrading and the bottle neck problem occurred. Access Structure (access policy) is classified as two types based on the Access structures. One is monotone access structure and another one is non-monotone access structure. Monotone access structure is defined as the only positive or possible attributes representing the access structure. ABE-KP's access structure is one of the examples of monotone access structure. To overcome the disadvantages we can go for the Hierarchical – Attribute based encryption (H-ABE). It is used to improve the scalability. H-ABE system avoids the performance degrading and bottleneck problem. KP-Attribute based encryption is also called the predicate based encryption. In ABE, system has the set of policies that are defined in the form Boolean formula. Boolean formula is called as the predicates. So all this attribute based encryptions comes under the predicate based encryption.

The only difference in between predicate and attribute based encryption is attribute hiding. i.e., in the ABE system, from cipher texts cloud, one can learn the useful information like attribute and access policy and is not hidden in the ABE systems. Instead of attribute revealing, in the Predicate encryption use inner product encryption that hide all information about the attributes and access policies from the cloud server. Predicate encryption is one of the functional encryption (FE). PE system comes under the functional encryption FE system. Functional encryption is used to define f(x, y). Where f is functionality and x is the set of attributes, y is the key. Functionality is applied to cipher texts. Through this we can decide what user can learn from the cipher texts encrypted with attributes x. Functionality encryption is used to hide the attributes and the policies. An ABE and PE system both comes under the functional encryption systems. In PE system define predicates P, with the set of attributes x, then decryption can be done only when it satisfies the condition P(x) =1, where predicates P are called as the Boolean formulas. The Boolean formulas are set by the data owner sets the policy that defines who can read the data.

II. FUNCTIONAL ENCRYPTION[1]

Functional Encryption (FE) is used to provide the functions to hide the access policy and the attributes in the following manner. Through this FE cannot gain any information about attributes and the access policy from cipher texts. Here access structure is used to define the access policy. FE encryption consists of the following phases. One Keysetup second is Encryption and last one is decryption.

A Keysetup:

Keysetup (T,B)→(SecK1,SecK2,SecK3…..SecKn, MATRIX(B)).

The keysetup will take input as tree (T) and Matrix (B) then it produces the output as the secret key for each attribute and produces the matrix for access tree. The private keys are generated from the Keysetup. Public key can be generated from the polynomial coefficients a_j. polynomial function can defined as,

$$f(x) = \{\prod{}^{m}{}_{j=1}\left(x - SecK_j\right) = \sum{}^{m}{}_{j=1} a_j x^j$$

Secret keys are find by,

$$SecK_i = h(B^I s_i a_i)$$

Here h is a generator

From the polynomial coefficients we can find the public following,

$$\text{Pub (k)} = (g^{aj}, g^{aj+1} \ldots\ldots)$$

Here the policies are hidden. Secret keys are generated based on each attribute.

B Encryption:

Encryption (Pub(k),M)→Cipher text (C)

Here encryption takes input as public key, original message and output as the cipher texts by the following,

a,b are two random numbers,

$$c = \begin{cases} c1 \\ c2 \end{cases}$$

$$c1 = (b^a.g^a{}_j.....g_{j+1}{}^a)$$
$$c2 = M.b^a$$

C Decryption:

Decryption(C,Seck$_j$)→M

Here the decryption takes input as cipher text and secret key and then it produces the output as Original message.

$$C/b^a = M$$

b^a can be computed by following,

$$b^a = (b^a.g_j{}^{SecKa}{}_{........}g_{j+1}{}^{Secka})$$

Here policies are hidden. However the policies are defined in the Keyset up phase itself. So it is always a static one. The policies cannot be change dynamically. Policies will be same for all of the data stored in the cloud.

III. CLOUD MASK FOR ATTRIBUTE HIDING[1]

Cloud mask is used to hide the attributes using protocol called OCBE protocol used to provide the message to the user only if satisfies the access policy. In the cloud computing attribute revealing is important issue in attribute based encryption. That issue can be avoided by the cloud mask. Through the cloud mask user cannot learn anything about the attributes from the cipher texts. This scheme is used to improve the confidentiality on the data in the cloud.

Cloud mask consists of following components,

A) Data Manager
B) Storage service
C) Users

A) Data Manager:
Data manager handle encryption under the access policy.

B) Storage service:
Storage service is defined as a cloud service that is used to store the document.

C) Users:
One who gets the storage services are called users

To hide the attribute following steps are followed:

Access control Vector and Broadcast group key management is combined to create the algorithm to hide the attributes. Access control vector is created by the concept of null space matrix. Null space is solved by the linear algebra methods. Attributes has the set of users. For each user here, create the secret key K_i. Here the matrix is created by the data manager. That matrix M contains each row represents the set of users for attributes. Access control vector is computed by matrix M‘s null space and distinct group key G_k and the random vector. If user of any attributes wants to do the decryption, then it is done by giving the valid key, thereby valid key with Access control matrix and vector produces the group key G_k.

A *Setup*

Group of size can be set in the setup phase.

B *Secret key generation:*

K_i € Secret keys it is choose randomly for each user.

C *Group key generation:*

Create key for each attribute of group of users. Each attribute represent the row of matrix. Null space is identified in this matrix and the group key is embedded with null space that is called as the Access control matrix.

D *Key derivation:*

This phase comes under the decryption phase. While decryption is done, each user must give his valid secret key that means through the Access control vector through which he can get the decryption key.

Here the attributes are hidden. However the data manager is one of the entities to do the encryption under the policies. Data manager is responsible for managing the all encryption. Here the data manger only defines the policies with the cipher texts. So the data owner must trust the data manager.

IV. PREDICATE ENCRYPTION [2]

Attribute encryption with attribute hiding is called as the predicates encryption. It is similar to the Attribute based encryption. Here predicates are defined as the access structure with the set of attributes. The secret key‘s attributes are satisfied with the access tree‘s attributes to do the decryption. That condition is defined by the formula, P(X) = 1, where P is defined as the predicates and X is defined as the attributes. It’ s more advantage is not just providing the attribute based access control but also to provide attribute hiding. But its main disadvantage is that decryption can be limited by its capacity in cloud server. To improve this, decryption based predicate encryption is proposed, though predicates encryption cannot gather any knowledge from the cipher texts.

A Set up:

In the key setup $G= G_p \times G_q \times G_r$ and it chooses random $R_{1,i}$,$R_{2,i \text{ € } G}$ hash function H_1,H_2 from which we can setup the public key and private keys for each attributes.

B Encryption:

Each attribute vector can be defined as $x=(x_1,x_2,x_3\ldots.)$, Message Me € $\{0,1\}^k$,H1,H2 be hash function, computes s , v from the hash function

s = H1(V,M)

v =H2 (V) here V € G_T $R_{3,i}$ $R_{4,i}$ € G_r

$\{C =V.P^s$,$C1= g^s_p$, C2= M XOR V $C_{1,i}=H_{1,i}\ s.Q^{axi}\ .\ R_{3,i}$ $C_{2,i}=H_{2,i}\ s.Q^{bxi}\ .\ R_{3,i}\ \}$

C Key generation:

For each attribute creating the key from the predicates vector (v)

$$SK_v = (R_5\ Q_6\ h^{-y} \prod h_{1,i}{}^{r1,j}\ h_{2,i}{}^{r2,j}, \{\ k_{1.i} = g_p{}^{r1,j}\ g_q{}^{f1,vi},\ k_{2.i} = g_p{}^{r2,j}\ g_q{}^{f2s,vi}\ \}\ )$$

Here choose random z. it is a secret key for each user and public key is called partial transformation keys PTK are k1.i, $k_{2.i}$.

Partial Transformation:

In this phase the cipher texts can be converted to the partial decryption form, if Predicates satisfy the condition with set of attributes. If it does not satisfy the predicates, then the transformation returns the empty. If it satisfies the condition, its partial decryption can be taken place.

Decryption

If the decryption takes secret key as the input with the partial transformation, then it will produce the fully decryption format which means plaintext can get from this phase. To get the original message we have to perform XOR operation between the hash function of random number V and the partial cipher texts.

However it produces the partial decryption by the cloud server itself. Cloud server has done the partial decryption means that it cannot assure the data validity, because there is chance of server that may learn useful information from the partial decryption.

V. BLIND EXTRACTION ALGORITHM [5]

Blind extraction is one of the protocols used to hide the useful information from the cloud and to find unauthorized persons. It uses following steps to hide the attribute. Predicate encryption and attribute based encryption system’s disadvantages are overcome by the blind Extraction algorithm. These Blind Extraction algorithms are mainly used in the data base cipher texts query searches. While searching the cipher texts unauthorized persons may get a chance to gain information from the cipher texts. ABE and PE are not applicable for searching on the cipher texts. Blind Extraction algorithm produces the tokens for cipher texts searching the authorized persons alone. This type token cannot be created by the predicate based encryption. Following steps are performed by the Blind Extraction algorithm.

Algorithm:

A Setup

In the algorithm setup all the users of public key can be defined by choosing the random numbers for each user. Here Authorized person involves to produce the tokens.

B Encryption

To encrypt the message, the random numbers are produced to get the cipher texts from the original message. In the cipher texts in each attributes the generators like g^x are created.

C Policies submitting

The policies are defined to access structure and to send the Authority A that will find the Access structure of each node's polynomial values and produces the each attributes secret key values.

D Decryption:

The secret key for each attribute and cipher texts attribute satisfy means that it produces the decryption from cipher texts.

However blind extraction algorithm uses the interaction between the user and the authority. So it is not applicable for the single point of failure and it is not applicable for the larger application. Because of single authority, its performance will degrade and is only applicable for the database based cipher texts search by the token.

VI. HIERARCHICAL REDICATE ENCRYPTION[3]

To improve the scalability, the hierarchy predicate encryption is used. Predicate based encryption has the single authority. In cloud, users cannot manage all the users with single authority. To improve this Predicate Encryption (PE) we go for HPE. The HPE trusted authority is the single to manage all the local authority (LA). Local authority is responsible for managing all the users. Here attributes are split by integer range are semantics range. Integer range for example is like age 30 to 60. Semantic range is for instant like Tamilnadu which has semantically all the cities. In the hierarchical tree, child nodes are combined to create the parent node. Following algorithm shows the steps in the Hierarchical Predicate Encryption (HPE).

A Set up the public keys:

In this phase, creating the public keys and master secret keys are based on the Seconds. Hence the public key and master secret keys are highly secured.

B Generating the index:

In this phase, the index for the each encryption is generated. Encryption can be done through the public key and the attributes.

C Generating the Capacity:

In this phase, keys are generated in an hierarchical manner. Here the Boolean formulas are converted to the polynomial vector, through which the vector can produce the secrets.

To manage all the local Authority (LA), single authority is responsible. So single point of failure problem occurred here also.

VII. PRIVACY PRESERVING ATTRIBUTE BASED ENCRYPTION (PP-ABE)[6]

PP-ABE system is used to hide the access policies and the attributes from the cipher texts. These systems hidden the attribute and policies from the cloud server as well as users. Here the attribute are classified as two types. One is application level attributes. Second is algorithm level attribute. Application level attribute is like human begins or roles of the human begins. Algorithm level attribute is defining the possible occurrences of the application level attribute which is positive or negative. The attribute is occurred with the access structure then the attribute is positive otherwise it is negative. Here the attribute define by the binary strings. A binary bit value 0 represents the negative occurrence of the attribute and the 1 represents the positive occurrence of the attribute. Every ABE system represents the access structure by different format. Here the access structure are defined by the AND logics. For Example Access structure define by the product term (positive attributes should be 3) AND (negative attributes should be 1). Thus Access structure can define by the attribute set $\overline{X}_1 X_3$. In this example rank is the application level attribute. Here the Access structure is the rank > lieutenant. To set the access structure by the AND logic we have to do mapping between the algorithm level and the application level attribute. Then define the algorithm level attribute (positive occurrence of attribute) by define the rank‘s are {…., rank, lieutenant, captain}. Here less than the rank attributes are negative attributes. To hide the access structure from the cipher texts, the access structures are omitted to hide from users and cloud servers. Here the access structures are does define implicitly. Through this we cannot learn anything about attributes defined in the access structures. Even authorized person also cannot learn anything from the access structures after the decryption. The algorithm consists of the following steps:

A Setup the master key:

This phase is used to generate the master key for the each attribute. Each attribute is mapping to the each element in the group. Here a_i and b_i are the two random numbers generated for each attribute. Group's element each represents the attributes.

Master key = {{a_i,b_i},α_i,β_i }

B Key generation:

The set of attributes can define as the X_{n-1},X_{n-2},X0 generate secret key for each attribute by following,

Secret key {D_i=$h^{r}_{i,}$ for each attribute X_i}

Here r is the random number.

C Encryption:

Encryption can be done by the following, here the access structure are in the product form. The message is hidden by the message authentication code (MAC). In the product form each attribute is represents the X called as the literal. attribute can represents either positive or negative. The cipher texts are constructed by the following steps:

(i)choose the random numbers
(ii)cipher texts computation Ci
(iii) find vectors for the product term
(iv)cipher texts generation with the Mac codes.

C = (M||MAC)

D Decryption:

It takes input as the cipher texts and the attributes and outputs the message if access structure satisfy the Access structures.

Here can hide the attributes and policies from the user and the cloud. However attributes are classified by the algorithm level.i.e positive occurrences of the attributes and the negative occurrences. To manage both positive and negative occurrences is difficult. Positive and negative occurrences define by the AND logics or product terms. To find the vectors for product terms are difficult.

VIII. SECURE DATA SHARING ABE (SS-ABE)[6]

In the ABE system users may be chance of transfer the decryption keys to unauthorized users. To avoid this malicious key distribution SS-ABE scheme is used. It is also used to hide the user's attributes from the cloud and the users. Tracing is done for cipher texts to avoid the malicious key distribution. In these scheme tracing is more challenging one. To improve this key policy-attribute based encryption (KP-ABE),

it produces the abuse free attribute based encryption.

Here n-bits user identity spaces are defined. Each bit represents the attributes. For tracing each attribute users has the unique identity. These identity and user's attributes are hiding from the users. Through this cannot learn anything from the cipher texts about the attributes matching or mismatching. The attributes are classified as the hidden normal attributes (HN) and the hidden identity attributes (HID).

A Setup the key:

This phase outputs the public key and the master key.

B Encryption:

Encrypt the message M with the set of attributes X, but the attributes are X_{hide} hidden.

C Key generation:

Key generation can be done by access structure as input and produces the output.

D Decryption:

Decryption can be done with decryption keys for each attributes of users.

The above SS-ABE system is used to hide the attributes from the authorized and the unauthorized persons. However here using one of the types ABE systems is Key policy-Attribute based Encryption (KP-ABE). Through this KP-ABE system's authority only can set the policies for the access structure. Because in KP-ABE's keys only having the policies. Here data owner trust the authority.

IX. SUMMARY

Summary of ABE	**Attribute based encryptions**		
	ABE	***Motivation***	***Limitation***
1	Functional Encryption[1][4]	Attribute hiding in cloud	Dynamic policies are not allowed
2	Cloud mask for attribute hiding[1]	Attribute hiding in cloud	Trust the data manager.
3	Predicate Encryption[2]	Attribute hiding in cloud	Cloud server itself doing
4	Hierarchical Predicate Encryption[3]	Attribute hidng,scalability in cloud	Single point of failure
5	Blind Extraction[5]	Attribute hiding in cloud	Authority failure
6	PP-ABE[6]	Attribute hiding in cloud	Difficult to manage both positive and negative attributes
7	SS-ABE[6]	Attribute hiding in cloud	To hide attribute KP-ABE system is using. In KP-ABE authority only can set the policies

X. CONCLUSION

The above related work shows many schemes that are used for the attributes, hiding with attribute based encryption in cloud computing. Each schemes have some short comes with that. But still specifically, there are no efficient papers for both attribute hiding and the policy hiding. There are papers for Query searching with the attribute hiding techniques using the predicate based encryption. Cloud masks are used to hide the papers. But here, lot of protocols is used in these schemes. These schemes go out of attribute based encryptions. Also these types of protocols are not efficient for large amount of users. To improve this scalability, finally hierarchical predicate based encryption (HPE) is used. Even though HPE is used for the scalability with attribute hiding there are some short comes like single point failure as in trusted authority (TA).

REFERENCES

[1] Hongjiao LI, Shan WANG, Xiuxia TIAN, Weimin WEI, Chaochao SUN,Daming LIU, “A Survey of Privacy-preserving Access Control in Cloud Computing, "in Journal of Computational Information Systems 10: 13 (2014) 5829-5846 july 1 2014.

[2] Fugeng Zeng, Chunxiang Xu, Wanpeng Li, Jia Mo, “Predicate Encryption for Inner Product in Cloud Computing,” in International Journal of Advancements in Computing Technology(IJACT) Volume4, Number13, July2012.

[3] Ming Li, Shucheng Yu, Ning Cao and Wenjing Lou “Authorized Private Keyword Search over Encrypted Data in Cloud Computing,” in. IEEE.

[4] Dan Boneh,and Amit Sahai and Brent Waters, “Functional Encryption: Definitions and Challenges,”.

[5] Yanbin Lu and Gene Tsudik, “Enhancing Data Privacy in the Cloud,”

[6] Shucheng Yu “Data Sharing on Untrusted Storage with Attribute-Based Encryption,”

[7] Apu Kapadia, Patrick P. Tsang†, Sean W. Smith “Attribute-Based Publishing with Hidden Credentials and Hidden Policies,” in *NDSS’07*. LNCS 5037, 2007, pp. 179–192.

[8] Shucheng Yu, Kui Ren, and Wenjing Lou “Attribute-Based Content Distribution with Hidden Policy,” In *Proc. of NPSEC’08,* Orlando, Florida, USA, 2008.

[9] A.Balu, K.Kuppusamy,“Ciphertext policy Attribute based Encryption with anonymous access policy,”