ATTRIBUTE BASED DATA SHARING IN CLOUD COMPUTING

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

OF

MASTER OF TECHNOLOGY

IN

INFORMATION SYSTEM

Submitted By:

Shipra Saini

(Roll No. - 2K16/ISY/13)

Under the supervision of

Ms. Priyanka Meel

Assistant Professor

Department of Information Technology

Delhi Technological University

DEPARTMENT OF INFORMATION TECHNOLOGY

DELHI TECHNOLOGICAL UNIVERSITY

SESSION: 2016-2018

CERTIFICATE

This is to certify that Ms. **SHIPRA SAINI (2K16/ISY/13)** has carried out the major project titled **"Attribute based data sharing in cloud computing"** as a partial requirement for the award of **Master of Technology** degree in **Information System** by **Delhi Technological University, Delhi.**

The Major project is a bona fide piece of work carried out and completed under my supervision and guidance during the academic session 2016-2018. The Matter contained in this thesis has not been submitted elsewhere for the award of any other degree.

Date:

(Project Guide)

Ms. Priyanka Meel

Assistant Professor

Department Of Information Technology

Delhi Technological University

DECLARATION

We hereby declare that the thesis work entitled "**Attribute based data sharing in cloud computing**" which is being submitted to Delhi Technological University, in partial fulfilment of requirements for the award of degree of Master of Technology (Information System) is a bonafide report of thesis carried out me. The material contained in the report has not been submitted to any university or institution for the award of any degree.

Shipra Saini
2K16/ISY/13

ACKNOWLEDGEMENT

I express my gratitude to my major project guide **Ms. Priyanka Meel , Assistant Professor** in **Department of Information Technology** at **Delhi Technological University, Delhi** for the valuable support and guidance she provided in making this major project. It is my pleasure to record my sincere thanks to my respected guide for her constructive criticism and insight without which the project would not have shaped as it has.

I humbly extend my word of gratitude to Dr. Kapil Sharma, Head of Department and other faculty members of department of Information Technology for providing their valuable help and time whenever it was required.

SHIPRA SAINI

Roll No.: 2K16/ISY/13

M.Tech. (Information System)

Department of Information Technology

Delhi Technological University, Delhi

ABSTRACT

In cloud computing secure data sharing can be done through very promising encryption technique which is known as Ciphertext-policy attribute-based encryption (CP-ABE) . In this scheme the access policy related to the data to be shared is controlled by the owner of the data . There is a security risk known as key escrow problem in the CP-ABE scheme because a trusted key authority is responsible for the issuing of the secret keys of users. Attribute with arbitrary state are not supported by most of the CP-ABE schemes which are already existing.This thesis aim at the removal of key escrow issue from attribute-based data sharing scheme and improvement of the expressiveness of attribute. So to ensure that the whole secret key of a user can't get compromised by cloud service provider or by key authority the proposal of an improved two-party key issuing protocol is given.

In addition, we present the idea of attribute with weight, being given to improve the outflow of attribute, which can not just stretch out the articulation from binary to arbitrary state, yet in addition help to lighten the complexity of access policy.

Improvement regarding complexity of ciphertext encryption and storage cost are done in this scheme. The performance analysis and the security proofs are provided which implies that efficient and secure data sharing is achieved by this scheme.

LIST OF FIGURES

Title No. **Page**

CONTENTS

Chapter 1

INTRODUCTION

Betterments and advancement in the sector of computing and networking empowers numerous individuals to effortlessly proportion their information with remains utilizing media storage appliances available online . People share data , photos or messages with other individuals or companions through informal communities for sparing the cost and also for the simplicity of transferring .As individuals appreciate the upsides of these new innovations and services , their worries about information security and access control additionally emerge. Cloud server can utilize the data improperly or other users can illegally access the data these can be considered as potential risks for data sharing. Individuals might want to make their sensitive or private information as it were open to the approved individuals with accreditations specified by them. ABE can be used as a method for characterizing the access policy because it is a very optimistic approach for the cryptography. Particularly, ciphertext-policy attribute-based encryption.

Consequently, every client or user with an alternate set of attributes is permitted to decode diverse pieces of data per the security strategy. Need for the dependency on the data storage server for avoiding unapproved information access is eliminated,which is the approach for the conventional control access.Consequently, the significant advantage is to a great extent lessen the requirement for processing and storing certificates of public key under traditional public key infrastructure (PKI).

Along with the advantages, CP-ABE has an issue known as the problem of key escrow . Ciphertext can be decrypted by the KGC by creating the attribute keys of the user. This could be a potential danger to the information secrecy or protection in the information sharing frameworks. The key revocation is the another challenge. Since a few users may change their associate attributes eventually, or some private keys may be imperiled, key

revocation or refresh for each trait is important with a specific end goal to make frameworks secure. This issue is significantly more troublesome particularly in ABE, since each quality is possibly shared by various users .It might bring about bottleneck amid rekeying technique or security breach because of the absence of a firewall like entity.

1.1Motivation of Study

Due to high scale-in and scale-out behaviour, Cloud computing has captured the eye of researchers in order to exploit the depths of this domain. A standout amongst the most encouraging cloud computing applications is sharing information online, for example, photograph partaking in On-line Social Networks and online wellbeing or health record framework.

A data owner (DO) is typically ready to store a lot of information in cloud for sparing the cost local data management . With no mechanism to protect data, provider of the cloud service (CSP), notwithstanding, can completely access all information of the user.CSP may misuse the data of the user for the benefits of business.This is a potential security risk. One of the hardest difficulties in the situation of cloud computing is how to safely and efficiently share user data.

Ciphertext-policy attribute-based encryption (CP-ABE) is turned into an essential encryption innovation to handle secure information sharing challenge .In a CP-ABE, an attribute set portrays the secret key related to the user , and ciphertext is related with access structure. Owner of data defines the access structure over the universe of attributes .Ciphertext can be decrypted by user only if attribute set matches the access structure.

Some open issues are yielded when we use CP-ABE scheme in cloud applications straightforwardly. A completely confided key authority (KA) issues the security keys of all the users. This leads to a security issue. All the user's ciphertext can be unscrambled by KA because it knows the secret key of user which remains al together against to the will of the client. It is termed as key escrow .

Other concern is the expressiveness of the attribute set.CP-ABE schemes which are existed are mostly defined on binary state attributes. If attribute satisfy the given scenario

then attribute value ‘1’ is used otherwise ‘0’ is used if it not satisfies.They do not deal with arbitrary-state attribute. In our work the introduction to the weighted attribute is done which simplifies the access policy as well as attribute expression is extended from binary to arbitrary state.Then there is a relief in the cost of encryption and storage.

The issues in cloud computing like key escrow gives the inspiration to learn more about the technology and to know how it can be removed and to do something about its enhancement . So it inspired me to study about CP-ABE and what has done so far and how it is used in cloud computing and what are its advantages.

1.2 *Our contribution*

Many schemes inspired to propose a data sharing scheme based on attributes for the applications of cloud computing which is represented by ciphertext-policy for removing escrow with ABE scheme with weighted attributes (CP-WABE-RE). It aims at the removal of two problems that are key escrow and expression of arbitrary state. The commitments of our work are as per the following:

Proposed a method that determines a key related guarantee, which ensures that KA does not perceive CSP’s secret master key and vice-versa. This enhancement finally results in independent behaviour of both entities and hence a complete key formation by either of the entity can be prevented and thence, security breaches can be diminished.

Show that the attribute weight enhanced the attribute expression. The weighted property can not just express arbitrary-state attribute , yet in addition the complexity of access policy is reduced. Therefore the capacity cost of ciphertext and calculation complexity in encryption can be decreased. In addition, larger attribute state can be expressed than ever under a similar condition.

The running of the scheme shows the efficiency in respect of the cost of storage and complexity computation.

1.3 *Organization of thesis*

The thesis is organized in various chapters as follows:

Chapter 2 gives an overview of the related work of the study that is what is the various research works have been done in this area and how all those work helped in evolution of our study.This includes the CP-ABE scheme and the related works done in past and their limitation.

Chapter 3 summarizes the research methods used in this thesis. This includes the model of system,distinctive phases in the proposed scheme and the procedures of the scheme.

Chapter 4 shows the results and their analysis. And at last the chapter 5 summarizes the research work under conclusion and suggests some future work.

Chapter 2

LITERATURE REVIEW

This chapter gives an overview of the research work done with relation to our thesis and is further subdivided into 2 sections i.e. KP-ABE and CP-ABE and limitation of these techniques.This section also contains the preliminaries like access structure, bilinear mapping and weighted access tree.

2.1 Overview of related research work done in past

Earlier Fuzzy identity-based encryption (IBE) was introduced by Sahai and Waters in 2005, which is the attribute-based encryption (ABE)'s seminal work. ABE comprises of 2 differing policies related to keys. If the policy is associated with ciphertext then its called CP-ABE and if it is associated with the key then its called KP-ABE.

Many approaches and schemes for CP-ABE were introduced later. Afterward, numerous CP-ABE plans with specific highlights have been introduced in the literature.
The computational overhead in the case of a user generated CP-ABE arrangement is improved by [1] from O(2N) to O(N) .Here , N denotes the number of attributes. Also efficient attribute and user revocation is presented in the scheme. Ciphertext 's size is around lessened to fifty-fifty as compared to the initial.

In most of the CP-ABE schemes an entity is required for issuing the the secret keys of users.It is fully trusted authority. This entity has its own master secret key which is used for the generation of User's secret keys.
The ciphertexts of system users can be decrypted by this authorized entity coz it has power to do it coz secret keys of user is generated by using its master key. Thus, it leads to the issue of key escrow problem, which is dealt by Chase and Chow [1].

The participants of key generation protocol are the non colluded authorities. They can't share their data belonging to the same user because they are not colluded with each other. Also can't link multiple attribute sets. For producing User's secret key , all authorities have to communicate with each others due to centralized authority absence. This leads to the degradation of performance[8], [4]. A communication overhead of O(N2) is resulted on rekeying phase as well as on system setup phase. N represents the number of authorities. Likewise, in addition to the attribute keys, O(N2) additional auxiliary key components are required to store by the user . Later on, a nameless exclusive key generation convention for IBE was proposed by Chow [6]. Here, private key can be issued by KA to a validated user in the dark of the rundown of the user's identification. It appears that this scheme could be appropriately utilized as a part of the setting of ABE if traits are dealt with the identities. For CP-ABE this plan won't work out because identity of person is a set of traits which isn't freely obscure.In 2013, [5] gave an enhanced security information sharing plan in view of the typical CP-ABE .

The concern of key escrow is conveyed by utilizing an escrow-relieved issuing key convention where the key formation authority and the data repository center cooperate to produce mystery key for user. In this way, the cost of estimation in producing user's private key increments in light of the fact that the convention needs intuitive estimation amidst the two parties. Additionally, Liu et al. brought in a finegrained passage control plan to manipulate it with hierarchy of attributes, where [10] and [11] are based over [12] and [13], individually. In the plans, the attributes are separated into numerous tiers to accomplish finegrained passage control plan for ranked attributes, yet the characteristics can simply express twofold state.

Afterward, an arbitrary-state ABE was proposed by Fan et al. [3] to comprehend the issue of the dynamic membership management. In the above paper, a traditional attribute is separated into two parts which are an attribute and its value.
Ex: we can denote traditional attributes as {"Nurse","Teacher","Scientist"} and the improved attributes can be described as {Career: "Nurse","Teacher","Scientist"} . Here "Career" is an attribute and "Nurse","Teacher" and "Scientist" denotes the attribute values. If there are

precise range of attributes then the cost of estimation of traditional scheme is less than attribute's cost .

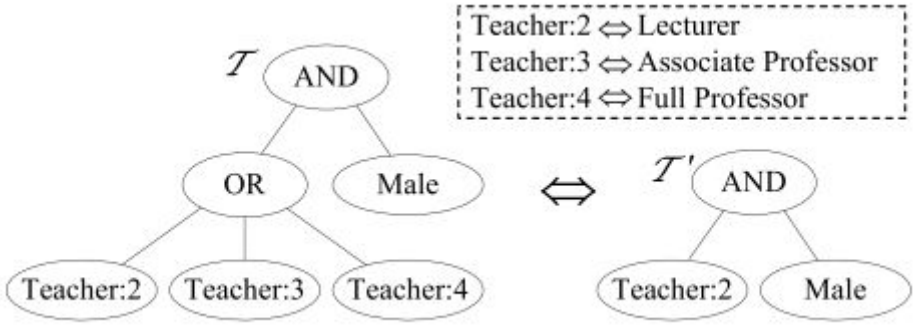


Fig. 2.1 : For a ciphertext the figure above shows the 2 equivalent access structures. A general access policy is represented by T in the current approach of CP-ABE . The enhanced accessing policy is denoted by T in the proposed scheme.

2.2 Overview of attribute-based encryption schemes

There are for the most part two kinds of attribute-based encryption schemes: A client can unscramble an inclined cryptotext just if the quality group complements the entrance architecture over the cryptotext.

2.2.1 Key -policy attribute-based encryption (KP-ABE):-

In KP-ABE, users' secret keys are produced in light of an access tree that characterizes the benefits extent of the concerned user, and information/data are scrambled over or encrypted over a set of attributes

2.2 .2 Ciphertext-policy attribute-based encryption (CP-ABE) :-

CP-ABE utilizes access trees for encrypting data and users' secret keys are created over a set of attributes. In a CP-ABE, user's secret key is portrayed by an attribute set, and ciphertext is related with an access structure. DO is permitted to characterize access structure over the

universe of properties or attributes. A user can unscramble or decrypt a given ciphertext only when user's attribute set matches the access structure over the ciphertext.

2.3 Access Structure

Let we denote a set comprises of parties by {P1,..., Pn}. Let 'A' be a collection defined as $A \subseteq 2\{P1,...,Pn\}$. It is monotone if $\forall B,C$ the following conditions are met : if $B \subseteq C$ and $B \in A$ then $C \in A$. An access structure is represented by the not empty of {P1,..., Pn } defined as $A \subseteq 2\{P1,...,Pn\} \setminus \{\varnothing\}$.

We consider the sets which are in A as the authorization sets .The sets which are not defined in A are unauthorized ones. Attributes takes the role of parties in our scheme.

Generally, except if expressed in other ways , the plan utilizes a monotone access architecture .

2.4 Bilinear Mapping

Let G0 and GT denotes the 2 multiplicative groups which are cyclic of prime order p. Let g be the generator of G0 .The expression $e\hat{} : G0 \times G0 \rightarrow GT$ denotes the bilinear mapping which satisfies the properties defined below :

• Bilinearity: For any a, b ∈ Zp and m, n ∈ G0, it has $e\hat{}(ma, nb) = \hat{}e(m,n)ab$.

• Non-degeneracy: There exists m, n ∈ G0 such that $e\hat{}(m,n) = 1$.

• Computability: For all m, n ∈ G0, there exists an algorithm with efficiency to compute $e\hat{}(m,n)$.

2.5 Weighted Access Tree

Let T be a weighted access tree, where root node of the tree is R. To facilitate description of the access tree, several functions and terms are defined as follows.

• **x** denotes a node of tree T .If x is a leaf node, it denotes an attribute with weight.

If x is a non-leaf node,it denotes a threshold gate, such as “AND”, “OR” and n-of-m (n<m)”.

• **numx** denotes the number of x’s children in T . For example, numR = 2 in Figure.

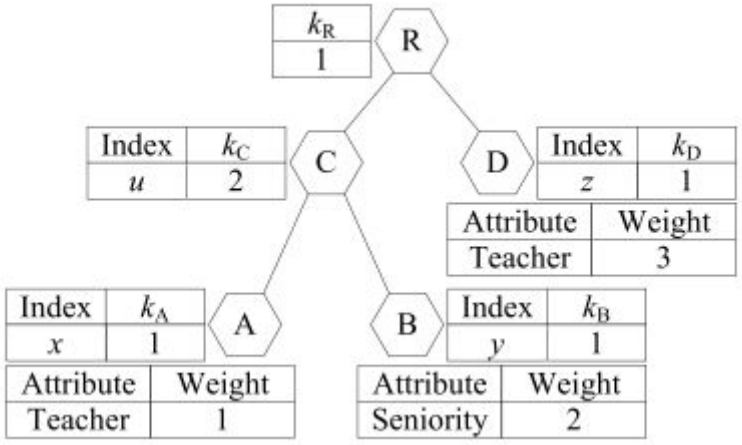


Figure 2.2 Shows an example of access tree structure with weights.

• **kx** denotes threshold value of node x, where numx >= kx >0 .

When kx = 1 and x is a non leaf node, it is an OR gate. When kx = numx and x is a non-leaf node, it is an AND gate. If x is a leaf node, kx = 1. For example, kR = 1 and kC = 2 denote an OR gate and an AND gate respectively in Fig. 2.

• **parent(x)** represents the node x’ s parent in T . For example, parent(A) = C in Fig. 2.

• **att(x)** denotes an attribute which is related to the leaf node x in T .

• **index(x)** returns an unique value which is related to the node . Here, in an arbitrary manner , the value is assigned to x in for a key given .

• **Tx** denotes the sub-tree of T which is rooted at the x node.

If a set of weighted attribute S assures the tree Tx , and is defined as Tx (S) = 1. Tx (S) is recursively computed . If x denotes a non-leaf node then ‘1’ is returned by Tx (S) if and

only if ‘1’ is returned by kx children at least. If x node is a leaf , then Tx (S) returns 1 if and only if the weight of attribute ωx from S must be equal to or greater than the weight of the leaf node. That is weight(ωx) ≥ weight(att(x)).

In addition, Morillo et al. proved that every weighted value of the threshold access structure can be defined as a natural number. Unless stated otherwise, the value of weight is a natural number in this paper. In Fig. 2, the access policy is denoted as: {(“Teacher:1” And “Seniority:2”) OR “Teacher:3”}. If one possesses attributes (“Teacher”, “Seniority”) with weight (“1”, “2”), he can satisfy the tree in Fig. 2; If the other one who possesses attribute (“Teacher”) with weight (“4”), he can also satisfy the access tree.

Chapter 3

RESEARCH METHODOLOGY

This chapter first describes the model of the system which is consist of four entities that are user , key authority, owner of data, cloud service provider . In the next segment, the distinctive phases of the scheme are shown .Then the algorithm of the scheme is represented.

3.1 Model of the system

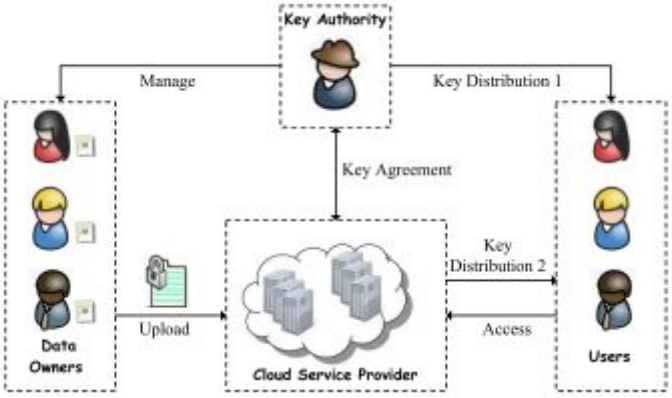


Figure 3.1 : This figure shows the model of the system with following four entities : KA , CSP ,User , Data owner .

This system is comprised of 4 types of individuals :

3.1.1 ***Key Authority (KA):*** KA is a half-faithful entity in the cloud structure which performs the tasks which are assigned and returns the correct results.Users' enrollment is the responsibility of KA. But this entity collects sensitive contents of the user. It generates maximum part of private key for every individual and also most part of system parameter.

3.1.2 ***Cloud Service Provider (CSP):*** This entity is accountable for managing servers in cloud and also it is semi-reliable . Data repository, calculation and transmission are the services given by CSP. It generates both parts of system parameter and secret key for every user for solving the problem of key escrow.

3.1.3 ***Data Owners (DO):*** These entities are the possessor of documents which are to be saved in cloud framework. They can define access structure of their own and encrypts the data by data encryption operation. They send ciphertext which is generated to CSP.

3.1.4 ***Users:*** They want to access the data files stored by the Data Owners in the system. They do so by accessing the ciphertext stored in cloud system. Ciphertext is downloaded by them and then corresponding decryption operation is executed in order to retrieve original file.

3.2 Phases of the scheme

(CP-WABE-RE): The **4 phases** of the scheme are defined as below **:**

3.2.1 *First Phase - Initialization of System :*

Two algorithms are executed in this phase : K.Set and C.Set.

1) **K.Set(1κ) → (P1, MK1)**.

KA runs this algorithm. The probabilistic operation which takes a parameter of security i.e κ as its input .A universal parameter P1 and a master key MK1 are generated as outputs .

2) **C.Set(1κ) → (P2, MK2)**.

This procedure is executed by CSP. A defence parameter κ is taken as input and P2 and MK2 are returned as outputs.

P = {P1, P2} and MK = {MK1, MK2} denotes the universal parameter and master key of method respectively. KA and CSP stores MK1 and MK2 respectively.

3.2.2 *Second Phase - Encryption of Data :*

For improving the efficiency of encryption, File F is encrypted by DO with content key ck . For this encryption, plain symmetric scrambling algorithm is used. Eck (M) denotes ciphertext of the file .

The following operation D.En(P, ck, A) → (CT) is used for the content key (ck) encryption. P, ck and an access policy A are taken as parameters by DO and it creates key ciphertext CT as an output which implicitly contains A and ck gets encrypted .Then, Eck (F) and CT are delivered by DO to CSP.

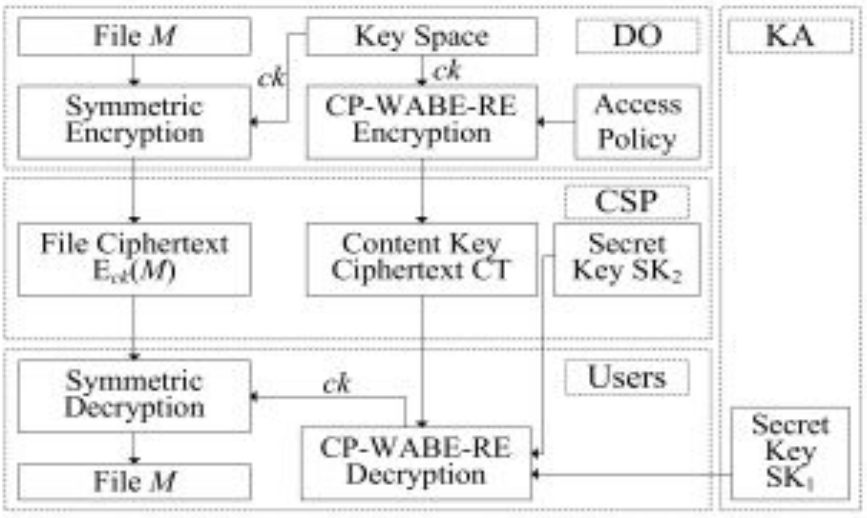


Figure 3.2 : The above figure shows the framework of the system

.

3.2.3 *Third Phase - Generation of User Key:*

This phase contains K.GenKey and C.GenKey algorithms.

(1) **K.GenKey(MSK1, S) → (SK1).**

KA inputs master secret key i.e MK1 and a set of attributes with weights S in key generating algorithm. The secret key SK1 is created as described by S.

(2) **C.GenKey**

In this part, an improved 2-party issuing of key protocol is proposed for removing key escrow problem . KA and CSP are two parties which perform this task with their master keys . Neither CSP nor KA can produce the whole group of secret keys of individual users .

KA does not collude with CSP ,this is taken as an assumption otherwise secret key of the user could be compromised by sharing master secret keys by them.

C.GenKey(MK2) → (SK2).

CSP inputs MK2 and results secret key SK2 by running the key generating protocol below.

- **KeyComKA↔CSP(MK1, I Dt,r, MK2) → (SK2)**. personalized secret r and a user identity I Dt .

Similarly, CSP inputs its MK2 and user identity I Dt .

Finally, a personalized key component SK2 is generated by CSP for the user. The user separately receives key components from KA and CSP and constructs the whole secret key SK as follows ,

SK = {SK1, SK2}.

3.2.4 *Fourth Phase (Data Decryption):*

This phase consists of U.Decrypt and D.Decrypt algorithms. Ciphertext Eck (F) and content key of the ciphertext CT is downloaded by the user first from CSP.Then, user have to satisfy some conditions in order to receive ck by running the U.Dec algorithm.

(1) **U.Dec(P, SK,CT) → (ck).**

User takes P, SK and CT as the parameters .CT includes access rule A. The content key can only be retrieved when the attribute set with weights i.e denoted by S equals the access approach A.

After getting content key , user uses it to further decrypt file F . This is done by utilizing D.Dec operation.

(2) **D.Dec(Eck (F), ck) → (F).**

Eck (F) and ck are taken as inputs by the User and outputs file F is generated based on symmetric decryption algorithm.

3.3 CP-WABE-RE Scheme proposal :

This section comprises of five procedures for CP-WABE-RE framework . The five processes are :-

- Initialization of system
- Construction of a new file (Encryption of Data)
- Authorization of recent user (Generation of user key)
- Accessing document file (Decryption of Data)
- Deletion of document file

3.3.1 *Initialization of system*

Let we define a bilinear group named G0 of the prime order p and the generator of the group be g. Let bilinear map be defined as $e\hat{} : G0 \times G0 \rightarrow GT$. A hash function is denoted by $H : \{0, 1\}* \rightarrow G0$. A set S is defined as $S = \{s1,s2,...,sm \in Zp\}$ for any $i \in Zp$.
An universe of attribute set A is defined as A = {a1,..., an}and W denotes a arrangement of weights which are characterized as $W = \{\omega1,...,\omega n\}(\omega1 \leq ... \leq \omega n)$ .Thus the framework contains n2 weighted attributes which are $A\tilde{} = \{a1 : \omega1,..., a1 : \omega n,..., an : \omega1,..., an : \omega n\}$, where the higher hierarchy of attributes is utilized, the greater weighted value is dispersed.

(1) K.Set(1κ).

An algorithm is runned by KA in which k is taken as the security parameter . At that point, KA picks arbitrary $\alpha1, \beta \in Zp$ and processes $h = g\hat{}\beta$ and $u1 = (\hat{}e(g, g))\hat{}\alpha1$. Finally, it gets P1 and MK1 as defined by the equation (1):

P1 = {G0, g, h, u1}, MK1 = {α1, β} (1)

(2) C.Set(1κ).

An operation is executed by CSP which takes a parameter of security κ as an input. In light of the κ, CSP picks an arbitrary no. α2 ∈ Zp and ascertains u2 = (ˆe(g, g))^α2 . At that point, it sets P2 and MK2 as follows (2):

P2 = {u2}, MK2 = {α2} (2)

Finally, the master secret key of system is described as MK = {{α1, β},{α2}}. P = {G0, g, h, u = u1·u2 =(ˆe(g, g))^α} denotes the public parameter . Here α = α1 + α2.

3.3.2 *Construction of a new file (Encryption of Data)*

The file F is processed by the data owner before transferring it to the CSP. This is done by the steps given below :

(1) A distinctive ID is chosen for the file F by the data owner.

(2) Symmetric encryption method is used for the encryption of file F with content key ck. In a key space , ck is chosen . Eck(F) denotes the ciphertext of the file after encryption.Eck denotes the Symmetric encryption operation using the content key .

(3) An access structure T is defined by DO. Encryption of ck is done by using encryption operation which is improved . The ciphertext CT of the ck is returned.

D.En(P,ck,T).

DO executes the improved algorithm and P, ck and T are taken as inputs which returns CT as an output. For every node x (including the leaf nodes) in T the selection of a polynomial qx is done. The selection of the node's information of qx is randomly done in the top to bottom manner from the node at the root R.

Then, dx is set to kx −1 for each node in T ,where dx is the degree of the polynomial and the threshold value is denoted by kx. qR(0) = s (s ∈ Zp) is set by the Data owner beginning from the node at the root R. Selection of s is randomly done.

For defining the polynomial qR completely, dR other points is randomly chosen by DO. qx(0)=qparent(x)(index(x)) is set for each non-root node x. For defining qx completely, dx other points is randomly chosen.An attribute with weight is denoted by each node at the leaf. The set of leaf nodes is denoted by Y and the minimal weight of every leaf is denoted by ωi in the access tree T.DO sets the minimum weight ωi .

CT is computed by DO by using formula (3).Then, integrated ciphertext {ID,CT, Eck(F) } is sent by the DO to CSP.

CT = { T , C = g^s , C = ck·(ˆ e(g,g))^α·s

{ ∀y∈Y , i ∈[1,n] : C y =h^(qy(0)) · H(att(y))−ωis ,

∀j ∈(i,n] , Cy,j =(H(att(y))) ^(−(ωj−ωi)s) } }

3.3.3 *Authorization of recent user (Generation of user key)*

User's enrollment is first accepted by KA . Then KA authenticates the user and if he is legal , a set of weighted attributes S are assigned to the user in accordance with his identity or role. Then, the secret key of the user is generated with the cooperation of CSP and KA. This phase is consisting of 2 algorithms C.GenKey and K.GenKey.

(1) **C.GenKey**.

Here for executing the work of CSP, an improved key issuing protocol is provided between KA and CSP.

KeyComKA↔CSP(MK1, ID t,r,MK2)

When a secret key is needed by a user then a protected 2-party (2PC) protocol is performed by KA and CSP. MK1 is input by KA and MSK2 is input by CSP. MK1 and MK2 are described by MK1 ={ α1,β} and MSK2 ={ α2} respectively. Also number r is randomly

chosen by KA. MK1 is not known by CSP and MK2 is not known by KA during this protocol.

x = (α1 + α2)βmod p is generated after the 2PC protocol execution and CSP gets it. After this , for the user the personalized key component is generated and for this KA and CSP goes through an interactive protocol.

1) A random no. ρ1 ∈ Zp is selected by CSP. X 1 =g^(x/ρ1) =g^((α1+α2)*β/ρ1) is calculated and {X1,PoK(ρ1,x)} is send to KA. Proof of knowledge is represented by Pok.

2) θ ∈ Zp is chosen by KA and Y1 = X1 ^(θ/β) =g ^((α1+α2)*θ/ρ1) and Y2 = h^(r*θ) is computed. {Y1,Y2,PoK(θ,β,r)} is transmitted to CSP.

3) ρ2 ∈ Zp is randomly selected by CSP and X2 =(Y1^ρ1) *(Y2^ρ2) = (g^((α1+α2)θ)*(h^rθ))^ρ2 is computed. {X2,PoK(ρ2)} is transferred to KA.

4) Y3 = X2^(1/θ) = (g^(α1+α2)*h^r)^ρ2 is calculated by KA and {Y3,PoK(θ)} is transmitted to CSP.

5) D = Y3^(1/ρ2) = (g^(α1+α2))*(h^r) = (g^α)*(h^r) is calculated CSP and a SK 2 ={D = g^α*h^r} which is personalized key component is transferred to the corresponding user t.

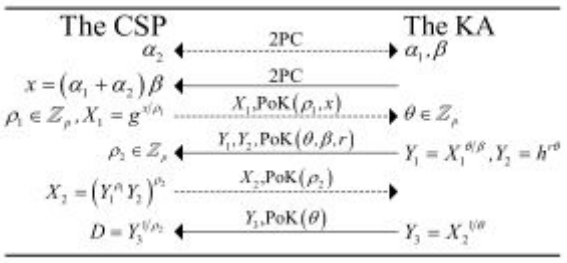


Figure 3.3 : Above figure shows the description of the protocol defined .

(2) **KA.KeyGen(MSK1,r, S).**

This algorithm is executed by KA which takes MK1 , a random number r (generated in C.GenKey) and S which is a a set of weighted attributes as inputs.

A weighted value ωj(ωj ∈ W) is defined for each weighted attribute j which belongs to S i.e j ∈ S.

SK1 is computed as follows :

SK1 = {L = g^r, ∀ j ∈ S : Dj = H(j)^ (r*ω*j)} (4)

So the secret key of the user can be constructed by receiving the components of the key from CSP and KA separately. SK is created as follows:

SK = {D = gαhr, L = gr , ∀ j ∈ S : Dj = H(j) rωj} (5)

3.3.4 *Accessing document file (Decryption of Data)*

In cloud framework, lawful users can openly question the scrambled text. At the point when a user demands CSP to get a cryptotext, it transfers the relating scrambled text {I D,CT, Eck (F)} to the user. Then when a user calls the improved Users.Decrypt algorithm he can acquire the content key ck. At that point, he utilizes ck to additionally unscramble the record file M utilizing Data.Decrypt task.

(1) **U.Dec(P,CT, SK).**

P, CT, and SK which is depicted by S are taken as inputs by user. If the access policy T is satisfied by the weighted attributes S which is possessed by the user.Then the content key ck can be acquired by the user.This procedure is defined as follows which is a recursive calculation.

(2) **D.Dec(Eck (F), ck)**

Ciphertext of file i.e. Eck (F) is taken as input by the user and content key ck too. File M can be decrypted using the symmetric decryption algorithm like AES or DES using the formula below

Dck [Eck (F)] = F (10)

here Dck represents a symmetric operation for decryption with the content key .

3.3.5 *Deletion of Data File*

Two types of data file deletion are performed .

1) **Discretionary Deletion:**

In the cloud system , ciphertext can be deleted by the data owners freely. If an encrypted file is to be deleted by DO then it is to be done by this procedure . Any secure signature scheme can be used.

(1) A request is sent to CSO by DO . ID of the file and KA's signature on it are included.

(2) Information request is verified by CSP. Ciphertext corresponding to the file is validated by CA . And if it's true ciphertext is deleted by CSP.

2) **Mandatory Blocking:**

A new function for providing file sharing is proposed .

(1) When a file gets accessed, the user evaluates how well the file is accessed.

(2) Assessments are synthesized for each file by CSP. CSP will mandatory block the files which are not consistent with the standards of evaluation. Private messages are received by the owner of data.

Chapter 4

RESULTS AND ANALYSIS

4.1 Theoretical Analysis

4.1.1 *Key Escrow and Attribute with weights:*

Cloud computing application , key escrow and characteristic of attribute with weights are shown in the table below for some schemes.

SCHEME	KEY ESCROW	WEIGHTED ATTRIBUTE	CLOUD SYSTEM
CP-WABE-RE	No	Yes	Yes
[3]	Yes	Yes	No
[5]	No	No	Yes
[7]	Yes	No	No

Table 4.1 - The above table shows the comparison of the schemes.

The problem of key escrow is removed from CP-WABE-RE scheme . For this purpose enhanced key assigning protocol is used. Key escrow problem is not solved by [3] and [7] both. But Hur [5] solved it by using escrow relieved key assigning protocol .

CP-WABE-RE provides the weighted attribute feature and also supports arbitrary-state attribute. Access policy which is associated with ciphertext is also simplified. [5] and [7] do not provide the feature of weighted attribute. Access structure is not simplified by [3] .Only arbitrary-state attribute is expressed.

All the three functions are supported by CP-WABE-RE scheme. Key escrow was solved Hur [5] so it satisfies cloud system environment.[3] and [7] can't be applied in cloud system because issue of key escrow is not resolved by them.

4.1.2 *Efficiency:*

The above four schemes are compared on the basis of effectiveness in Table 4.2 and Table 4.3. The utilized representation are characterized in Table 4.4. in the accompanying analysis of access rule, re-encryption of information of [5] and [7], and changing association management (that is, user adding, remnants, and attribute upgrading) of [3] are excluded to disentangle the comparisons.

Likewise,the cost of transferring is excluded while executing the inter dependant protocols in both [5] and scheme proposed .In Table 4.2, the plans are looked at regarding the size of CT , SK , P and MK .

The repository overhead is represented by size of CT in cloud computing and furthermore suggests the interaction cost from CSP to users, and similarly from DO to CSP. Size of SK indicates the necessary cost of storage for every user. Sizes of MK and P shows the repository overhead of KA and CSP regarding master key and open parameter.

Efficiency Comparisons : Storage Cost

Scheme	Size of CT	Size of SK	Size of PP	Size of MSK
CP-WABE-RE	$[\Sigma_{i=1}^{\lvert\mathbb{A}_C\rvert}(\omega_i - \omega_{i_1} + 1) + 1]L_{\mathbb{G}_0} + L_{\mathbb{G}_T}$	$(\lvert\mathbb{A}_u\rvert + 2)L_{\mathbb{G}_0}$	$3L_{\mathbb{G}_0} + L_{\mathbb{G}_T}$	$3L_{\mathbb{Z}_p}$
5	$(2\lvert\mathbb{A}_C\rvert + 1)L_{\mathbb{G}_0} + L_{\mathbb{G}_T}$	$(2\lvert\mathbb{A}_u\rvert + 1)L_{\mathbb{G}_0}$	$3L_{\mathbb{G}_0} + L_{\mathbb{G}_T}$	$L_{\mathbb{Z}_p} + L_{\mathbb{G}_0}$
7	$(\lvert\mathbb{A}_C\rvert + 1)L_{\mathbb{G}_0} + L_{\mathbb{G}_T}$	$(\lvert\mathbb{A}_u\rvert + 2)L_{\mathbb{G}_0}$	$3L_{\mathbb{G}_0} + L_{\mathbb{G}_T}$	$L_{\mathbb{G}_0}$
3	$2(\lvert\mathbb{A}_C\rvert + 1)L_{\mathbb{G}_0} + 2L_{\mathbb{G}_T}$	$(3\lvert\mathbb{A}_u\rvert + 1)L_{\mathbb{G}_0}$	$(n+2)L_{\mathbb{G}_0} + 2L_{\mathbb{G}_T} + knL_{\mathbb{Z}_p}$	$L_{\mathbb{G}_0}$

Table 4.2 - This table shows the efficiency comparisons of the storage cost.

As appeared in Table II, when ωi = ωi1 , all traits have equivalent weights in CP-WABE-RE plot. In this way our plan is identical to [5] and [7]. In the interim, CT measure is decreased as (|AC|+1)LG0 + LGT in CP-WABE-RE plot, which is equivalent to

[7]'s. Contrasting with [3] and [5], the CT estimate in our proposed plan and [7] is lessened by about half. At the point when ωi = ωi1 , CP-WABE-RE plan can utilize an attribute to express (ωi − ωi1 + 1) properties which have distinctive weights. In this manner, it requires littler capacity cost in CT than the others. In addition, we can find that the SK measure in CP-WABE-RE plot is equivalent to [7]'s, which is littler than [3]'s and [5]'s.

Besides, when |Au|→∞, the capacity overhead in our plan is diminished by about half contrasting with [5]'s. Furthermore, the storage cost in our plan is diminished about by 66.67% contrasting with [3]'s in principle.

Likewise, we can watch that the PP estimate is equivalent among [5], [7] and CP-WABE-RE plot. What's more, the size of PP in [3] is the longest since it is identified with the number of system attributes n and the number of system users k.. About the size of MSK, we can find that the parameter in CP-WABE-RE scheme doesn't give off an impression of being vastly different from the others.

Efficiency Comparisons : Computation Cost

Scheme	New File Creation (Data Encryption)	Data File Access (Data Decryption)	New User Authorization (User Key Generation)
CP-WABE-RE	$\{[\Sigma_{i=1}^{\lvert\mathbb{A}_C\rvert}(\omega_i - \omega_{i_1} + 2)] + 1\}\mathbb{G}_0 + 2\mathbb{G}_T$	$(2\lvert\mathbb{A}_u\rvert + 1)C_{\hat{e}} + (2\lvert S\rvert + 2)\mathbb{G}_T$	$(\lvert\mathbb{A}_u\rvert + 9)\mathbb{G}_0$
5	$(2\lvert\mathbb{A}_C\rvert + 1)\mathbb{G}_0 + 2\mathbb{G}_T$	$(2\lvert\mathbb{A}_u\rvert + 1)C_{\hat{e}} + (2\lvert S\rvert + 2)\mathbb{G}_T$	$(2\lvert\mathbb{A}_u\rvert + 4)\mathbb{G}_0$
7	$(2\lvert\mathbb{A}_C\rvert + 1)\mathbb{G}_0 + 2\mathbb{G}_T$	$(2\lvert\mathbb{A}_u\rvert + 1)C_{\hat{e}} + (2\lvert S\rvert + 2)\mathbb{G}_T$	$(\lvert\mathbb{A}_u\rvert + 3)\mathbb{G}_0$
3	$(2\lvert\mathbb{A}_C\rvert + 2)\mathbb{G}_0 + 3\mathbb{G}_T$	$(3\lvert\mathbb{A}_u\rvert + 1)C_{\hat{e}} + (2\lvert S\rvert + 3)\mathbb{G}_T$	$(4\lvert\mathbb{A}_u\rvert + 2)\mathbb{G}_0$

Table 4.3 - This table shows the efficiency comparisons for computation cost.

Notation	Definition
$\mathbb{G}_i$	exponentiation or multiplication in $group$ $(i = 0, T)$
$C_{\hat{e}}$	$\hat{e}$ operation, $\hat{e}$ denotes bilinear pairing
$\mathbb{Z}_p$	$Group$ $\{0, 1, \ldots, p-1\}$ under multiplication modulo p
S	Least interior nodes satisfying an access structure
$\mathbb{A}_C$	Attributes appeared in ciphertext CT
$\mathbb{A}_u$	Attributes of user u
ω_i	Maximum weight of attribute i in system
ω_{i_1}	Weight of attribute i in ciphertext CT
n	Number of attributes in system
k	Number of users in system
L_*	Bit-Length of element in *
$\lvert * \rvert$	Number of elements in *

Table 4.4 - The table defines the notations for the comparisons of the efficiency.

In Table III, we assess the calculation cost of encryption, decryption and user key generation. In the period of new document creation (information encryption), the calculation cost in CP-WABE-RE plan can be diminished as (2|AC|+1)G0+2GT when ωi = ωi1 , which is generally equivalent to [3]'s, [5]'s, and [7]'s.

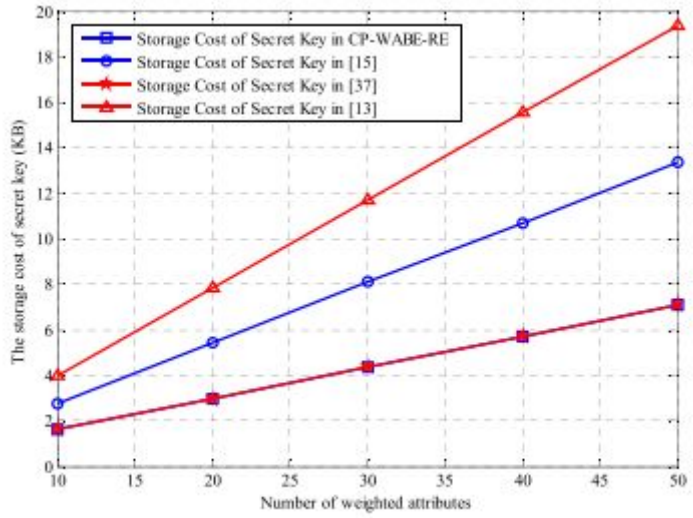
Like Table II, when ωi = ωi1 , CP-WABE-RE plot processes an attribute to represent numerous attributes which have distinctive weights. Then, access structure related with a ciphertext can be simplified by it. The scheme [3] is without expressing arbitrary-state attribute feature and is thus different from our scheme. It doesn't have the element of our scheme.Along these lines, the cost of encryption is saved in CP-WABE-RE when ωi = ωi1.In the period of accessing file i.e decryption of data , the parameter length is equivalent among [5], [7] and CP-WABE-RE scheme. The cost of computation on decoding in [3] is greater than the others.

What's more, in the period of new user authorization i.e generation of user's key, our proposed scheme just expends extra 6G0 of calculation cost in tackling key escrow issue, comparing with that in [7]. In the mean time, the calculation cost on generation of key in

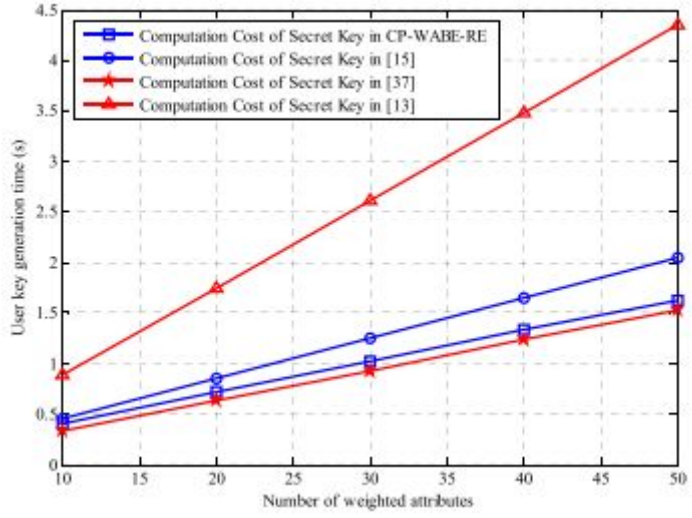
CP-WABE-RE plot is littler than [3]'s and [5]'s. Moreover, when $|Au|\rightarrow\infty$, the calculation cost in ours is diminished almost by half i.e 50% in principle than [5]'s, where the cost for transmission isn't engaged with both the two plans. In the meantime,the cost in ours is diminished by about 75% contrasting with [3]'s in principle.

4.2 Results of the comparison of key escrow

(a) The score of storage of secret key.

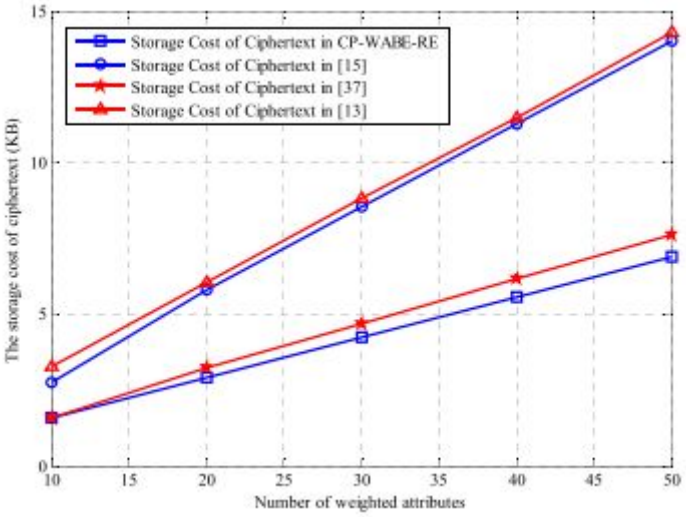


(b) The score of time of creating key of the individual . The coordinate is cost of time or storage overhead of user's secret key.The abscissa is the quantity of traits with weights in secret key of user.

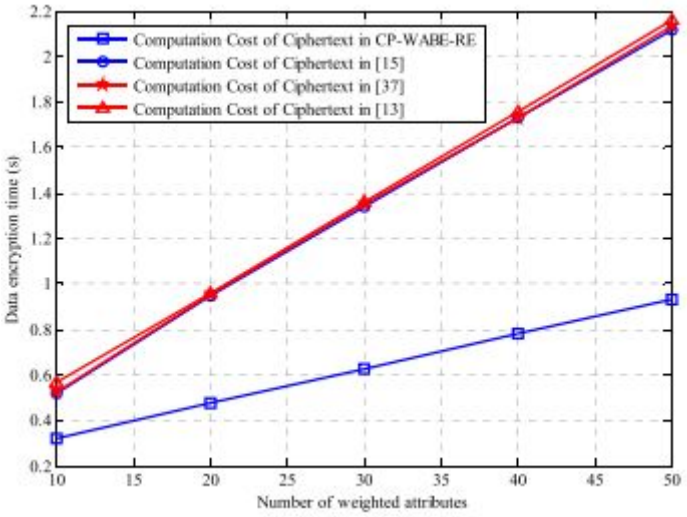


Results of comparison of attribute with weights.

(a) The score of storage of encrypted text.



(b) The score of time of encryption of data. The no. of attributes or traits showed up in encrypted text is the abscissa. Overhead of repository or time cost of encryption at DO is the coordinate .



Chapter 5

CONCLUSION AND FUTURE WORK

6.1 *Conclusion*

In cloud computing , attribute based data sharing is redesigned which improves the key escrow problem. Data secretiveness in cloud computing is enhanced .CSP and KA are semi trusted entities and they cannot bargain the secret key of the user individually until they collude with each other , which they cannot do because they are honest by assumption. The attribute with weights is delivered for improving the interpretation of attribute which describes arbitrary state attributes. Complexity of access policy,time cost as well as storage cost of cipher text is reduced. Security proof is presented which ensures the efficiency and security of the above scheme.

6.2 Future Work

And as a part of the future work, the CP-ABE technique can be further utilised . In the future, our work can be continued in several directions. Securely forwarding the revocation related computations to the CSP (or even to the user), as we mentioned in a remark, could allow immediate banning of a user, disallowing the decryption of all previously (and later) encrypted ciphertexts. Steps in this direction, without assuming trusted CSP, would be useful. The method of identity-based user revocation can be the foundation of a future method that allows non monotonic access structures in multi-authority setting. However our scheme cannot be applied directly for this purpose, it may be used to develop ideas in this field.

REFERENCES

[1] M. Chase and S. S. M. Chow, “Improving privacy and security in multi-authority attribute-based encryption,” in Proc. 16th ACM Conf. Comput. Commun. Secur., 2009, pp. 121–130.

[2].Shulan Wang ; Kaitai Liang ; Joseph K. Liu ; Jianyong Chen ; Jianping Yu ; Weixin Xie, "Attribute-Based Data Sharing Scheme Revisited in Cloud Computing," IEEE Transactions on Information Forensics and Security ,April 2016 .

[3] C.-I. Fan, V. S.-M. Huang, and H.-M. Ruan, “Arbitrary-state attribute based encryption with dynamic membership,” IEEE Trans. Comput., vol. 63, no. 8, pp. 1951–1961, Aug. 2014.

[4] S. Müller, S. Katzenbeisser, and C. Eckert, “Distributed attribute based encryption,” in Proc. 11th Int. Conf. Inf. Secur. Cryptol., 2009, pp. 20–36.

[5] J. Hur, “Improving security and efficiency in attribute-based data sharing,” IEEE Trans. Knowl. Data Eng., vol. 25, no. 10, pp. 2271–2282, Oct. 2013.

[6] S. S. M. Chow, “Removing escrow from identity-based encryption,” in Proc. 12th Int. Conf. Pract. Theory Public Key Cryptogr., 2009, pp. 256–276.

[7] X. Xie, H. Ma, J. Li, and X. Chen, “An efficient ciphertext-policy attribute-based access control towards revocation in cloud computing,” J. Universal Comput. Sci., vol. 19, no. 16, pp. 2349–2367, Oct. 2013.

[8] M. Chase, “Multi-authority attribute based encryption,” in Proc. 4th Conf. Theory Cryptogr., 2007, pp. 515–534.

[9] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attributebased encryption,” in Proc. IEEE Symp. Secur. Privacy, May 2007, pp. 321–334.

[10] X. Liu, J. Ma, J. Xiong, Q. Li, and J. Ma, “Ciphertext-policy weighted attribute based encryption for fine-grained access control,” in Proc. 5th Int. Conf. Intell. Netw. Collaborative Syst., Sep. 2013, pp. 51–57.

[11] X. Liu, J. Ma, J. Xiong, and G. Liu, “Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data,” Int. J. Netw. Secur., vol. 16, no. 6, pp. 437–443, Nov. 2014

[12] L. Cheung and C. Newport, “Provably secure ciphertext policy ABE,” in Proc. 14th ACM Conf. Comput. Commun. Secur., 2007, pp. 456–465.

[13] B. Waters, “Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization,” in Proc. 14th Int. Conf. Pract. Theory Public Key Cryptogr., 2011, pp. 53–70.