Decentralized Access Control Technique with Anonymous Authentication

Kavya Ratna A.,   Mr. K. C. Thiagarajan, M.E.
Assistant Professor(Cse Department).
SRS College Of Engineering And Technology.

Abstract: **In this paper, a new decentralized access control scheme for secure data storage in clouds that supports anonymous authentication is proposed. In the proposed scheme, the cloud verifies the authenticity of the series without knowing the user's identity before storing data. This scheme also has the added feature of access control in which only valid users are able to decrypt the stored information. This system also prevents replay attacks and supports creating, modifying, and reading of data stored in the cloud. This paper also address user revocation. Our authentication and access control scheme is more robust and decentralized, unlike other access control schemes designed for clouds which are centralized. When compared to decentralized approaches the computation, communication and storage overheads are improved.**

Index Terms — Access control, authentication, attribute-based signatures, attribute-based encryption, cloud storage.

I INTRODUCTION :

Research in cloud computing is receiving a lot of attention from both academic and industrial worlds. In cloud computing, users can outsource their computation and storage to servers using Internet. This frees users from the hassles of maintaining resources on-site. Clouds can provide several types of services like applications , infrastructures , and platforms to help developers write applications . Much of the data stored in clouds is highly sensitive, for example, medical records and social networks. Security and privacy are, thus, very important issues in cloud computing. In one hand, the user should authenticate itself before initiating any transaction, and on the other hand, it must be ensured that the cloud does not tamper with the data that is outsourced. User privacy is also required so that the cloud or other users do not know the identity of the user. The cloud can hold the user accountable for the data it outsources, and likewise, the cloud is itself accountable for the services it provides. The validity of the user who stores the data is also verified. Apart from the technical solutions to ensure security and privacy, there is also a need for law enforcement.

Cloud servers prone to Byzantinefailure, where a storage server can fail in arbitrary ways. The cloud is also prone to data modification and servercolluding attacks. In server colluding attack, the adversarycan compromise storage servers, so that it can modify datafiles as long as they are internally consistent. To providesecure data storage, the data needs to be encrypted.However, the data is often modified and this dynamicproperty needs to be taken into account while designingefficient secure storage techniques.Efficient search

on encrypted data is also an importantconcern in clouds. The clouds should not know the querybut should be able to return the records that satisfy thequery. This is achieved by means of searchable encryption. The keywords are sent to the cloud encrypted, andthe cloud returns the result without knowing the actualkeyword for the search. The problem here is that the datarecords should have keywords associated with them toenable the search. The correct records are returned onlywhen searched with the exact keywords.

Security and privacy protection in clouds are beingexplored by many researchers. Many homomorphic encryption techniques have been suggested to ensure that the cloud is not able to read the data whileperforming computations on them. Using homomorphicencryption, the cloud receives ciphertext of the data andperforms computations on the ciphertext and returns theencoded value of the result. The user is able to decode theresult, but the cloud does not know what data it hasoperated on. In such circumstances, it must be possible forthe user to verify that the cloud returns correct results.Accountability of clouds is a very challenging task andinvolves technical issues and law enforcement. Neitherclouds nor users should deny any operations performed orrequested. It is important to have log of the transactionsperformed; however, it is an important concern to decidehow much information to keep in the log.

Considering the following situation: A law student, Alice, wants to send a series of reports about some malpractices by authorities of University X to all theprofessors of University X, research chairs of universities in the country, and students belonging to Law department in all universities in the province. She wants to remain anonymous while publishing all evidence of malpractice.

She stores the information in the cloud. Access control is important in such case, so that only authorized users can access the data. It is also important to verify that the information comes from a reliable source. The problems ofaccess control, authentication, and privacy protection should be solved simultaneously. We address this problem in its entirety in this paper. Access control in clouds is gaining attention because it is important that only authorized users have access to valid service. A huge amount of information is being stored in the cloud, and much of this is sensitive information. Care should be taken to ensure access control of this sensitive information which can often be related to health, important documents (as in Google Docs or Dropbox) or even personal information (as in social networking). There are broadly three types of access control: user-based access control (UBAC), role-based access control (RBAC), and attribute-based access control (ABAC).

In UBAC, the access control listcontains the list of users who are authorized to access data. This is not feasible in clouds where there are many users. In RBAC users are classified based on their individual roles. Data can be accessed by users who have matching roles. The roles are defined by the system. For example, only faculty members and senior secretaries might have access to data but not the junior secretaries. ABAC is more extended in scope, in which users are given attributes, and the data has attached access policy. Only users with valid set of attributes, satisfying the access policy, can access the data. For instance, in the above example certain records might be accessible by faculty members with more than 10 years of research experience or by senior secretaries with more than 8 years experience. An area where access control is widely being used is health care. Clouds are being used to store sensitive information about patients to enable access to medical professionals, hospital staff, researchers, and policy makers. It is important to control the access of data so that only authorized users can access the data. Using ABE, the records are encrypted under some access policy and stored in the cloud. Users are given sets of attributes and corresponding keys. Only when the users have matching set of attributes, can they decrypt the information stored in the cloud.

Access control is also gaining importance in online social networking where users store their personal information, pictures, videos and share them with selected groups of users or communities they belong to. Such data are being stored in clouds. It is very important that only the authorized users are given access to those information. A similar situation arises when data is storedin clouds, for example, in Dropbox, and shared with certain groups of people. It is just not enough to store the contents securely in the cloud but it might also be necessary to ensure anonymity of the user. For example, a user would like to store some sensitive information but does not want to be recognized. The user might want to post a comment on an article, but does not want his/her identity to be disclosed. However, the user should be able to prove to the other users that he/ she is a valid user who stored the information without revealing the identity. There are cryptographic protocols like ring signatures , mesh signatures, group signatures , which can be used in these situations. Ring signature is not a feasible option for clouds where there are a large number of users. Group signatures assume the preexistence of a group which

OUR CONTRIBUTION:
The main contributions of this paper are the following:
1. Distributed access control of data stored in cloud sothat only authorized users with valid attributes canaccess them.
2. Authentication of users who store and modify theirdata on the cloud.
3. The identity of the user is protected from the cloudduring authentication.
4. The architecture is decentralized, meaning that therecan be several KDCs for key Management.
5. The access control and authentication are bothcollusion resistant, meaning that no two users cancollude and access data or authenticate themselves,if they are individually not authorized.
6. Revoked users cannot access data after they havebeen revoked.

might not be possible in clouds. Mesh signatures do not ensure if the message is from asingle user or many users colluding together. For these reasons, a new protocol known as attribute-based signature (ABS) has been applied. In ABS, users have a claim predicate associated with a message. The claim predicate helps to identify the user as an authorized one, without revealing its identity. Other users or the cloud can verify the user and the validity of the message stored. ABS can be combined with ABE to achieve authenticated access control without disclosing the identity of the user to the cloud. Existing work on access control in cloud are centralized in nature. Most schemes use ABE. Some scheme uses a symmetric key approach and does not support authentication. However, the authors take a centralized approach where a single key distribution center (KDC) distributes secret keys and attributes to all users. Unfortunately, a single KDC is not only a single point of failure but difficult to maintain because of the large number of users that are supported in a cloud environment. We, therefore, emphasize that clouds should take a decentralized approach while distributing secret keys and attributes to users. It is also quite natural for clouds to have many KDCs in different locations in the world. Although a decentralized approach is proposed, their technique does not authenticate users, who want to remain anonymous while accessing the cloud. However, the scheme did not provide user authentication. The other drawback was that a user can create and store a file and other users can only read the file. Write access was not permitted to users other than the creator. In the preliminary version of this paper [1], we extend our previous work with added features that enables to authenticate the validity of the message without revealing the identity of the user who has stored information in the cloud. In this version we also address user revocation. We use ABS scheme to achieve authenticity and privacy. Unlike , our scheme is resistant to replay attacks, in which a user can replace fresh data with stale data from a previous write, even if it no longer has valid claim policy. This is an important property because a user, revoked of its attributes, might no longer be able to write to the cloud. We, therefore, add this extra feature in our scheme and modify appropriately. Our scheme also allows writing multiple times which was not permitted in our earlier work .

7. The proposed scheme is resilient to replay attacks. Awriter whose attributes and keys have been revokedcannot write back stale information.
8. The protocol supports multiple read and write onthe data stored in the cloud.
9. The costs are comparable to the existing centralizedapproaches, and the expensive operations are mostlydone by the cloud.

II RELATED WORK:

In ABE, a userhas a set of attributes in addition to its unique ID. There aretwo classes of ABEs. In key-policy ABE or KP-ABE , the sender has an access policy to encrypt data. A writer whose attributes and keys have been revoked cannotwrite back stale information. The receiver receives attributesand secret keys from the attribute authority and is

Special Issue - 2015

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
TITCON-2015 Conference Proceedings

able todecrypt information if it has matching attributes. InCiphertext-policy, CP-ABE , the receiver has the

access policy in the form of a tree, with attributes as leavesand monotonic access structure with AND, OR and otherthreshold gates.All the approaches take a centralized approach and

allow only one KDC, which is a single point of failure. Chase proposed a multiauthority ABE, in which thereare several KDC authorities (coordinated by a trustedauthority) which distribute attributes and secret keys tousers. Multiauthority ABE protocol, which required no trusted authority which requiresevery user to have attributes from at all the KDCs. Recently,

Lewko and Waters [35] proposed a fully decentralized ABEwhere users could have zero or more attributes from eachauthority and did not require a trusted server. In all thesecases, decryption at user's end is computation intensive. So,this technique might be inefficient when users access usingtheir mobile devices. To get over this problem, Green et al. proposed to outsource the decryption task to a proxyserver, so that the user can compute with minimumresources (for example, hand held devices). However, thepresence of one proxy and one KDC makes it less robustthan decentralized approaches. Both these approaches had

no way to authenticate users, anonymously. Yang et al. presented a modification of,authenticate users, whowant to remain anonymous while accessing the cloud.To ensure anonymous user authentication ABSs were introduced by Maji et al. This was also a centralizedapproach. A recent scheme by Majietal.takes adecentralized approach and provides authentication withoutdisclosing the identity of the users. However, as mentioned earlier in the previous section it is prone to replay attack.

<div align="center">

III PROPOSED PRIVACY PRESERVING AUTHENTICATED ACCESS CONTROL SCHEME

</div>

In this section, we propose our privacy preservingauthenticated access control scheme. According to ourscheme a user can create a file and store it securely in thecloud. This Scheme consists of use of the two protocols ABE and ABS. We will first discuss our scheme in details and then providea concrete example to demonstrate how it works. We referto the Fig. 1. There are three users, a creator, a reader, andwriter. Creator Alice receives a token _ from the trustee,who is assumed to be honest. A trustee can be someone likethe federal government who manages social insurancenumbers etc. On presenting her id (like health/socialinsurance number), the trustee gives her a token _. Thereare multiple KDCs (here 2), which can be scattered. Forexample, these can be servers in different parts of the world.A creator on presenting the token to one or more KDCsreceives keys for encryption/decryption and signing. In theFig. 1, SKs are secret keys given for decryption, Kx are keys for signing. The message MSG is encrypted under theaccess policy X. The access policy decides who can accessthe data stored in the cloud. The creator decides on a claim policy Y, to prove her authenticity and signs the messageunder this claim. The ciphertext C with signature is c, and issent to the cloud. The cloud verifies the signature and stores
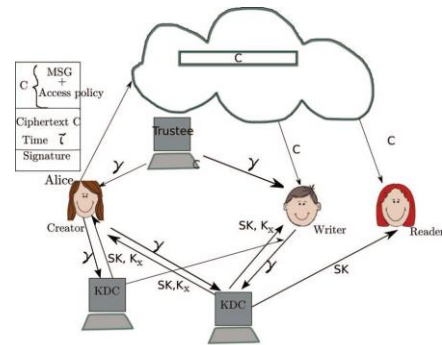
the ciphertext C. When a reader wants to read, the cloudsends C. If the user has attributes matching with accesspolicy, it can decrypt and get back original message.Write proceeds in the same way as file creation. Bydesignating the verification process to the cloud, it relieves

the individual users from time consuming verifications.When a reader wants to read some data stored in the cloud,it tries to decrypt it using the secret keys it receives from the

KDCs. If it has enough attributes matching with the accesspolicy, then it decrypts the information stored in the cloud.

DATA STORAGE IN CLOUDS

A user U_u first registers itself with one or more trustees. Forsimplicity we assume there is one trustee. The trustee gives it a token $\gamma=(u,k_{base},k_0,\rho)$where ρ *is* the signature onu$\|K_{base}$signed with the trustees private key T_{Sig} (by (6)).The KDCs are given keys $P_k[i];S_k[i]$for encryption decryption and ASK[i]APK[i] for signing/verifying. Theuser on presenting this token obtains attributes and secretkeys from one or more KDCs. A key for an attribute xbelonging to KDC A_i is calculated ask$_x=k_{base}^{1/(a=bx)}$, where(a,b) €ASK[i]. The user also receives secret keys sk$_{x;u}$ forencrypting messages. The user then creates an access policyX which is a monotone Boolean function. The message isthen encrypted under the access policy as

<div align="center">

C=ABE.Encrypt(MSG,X)

</div>

The user also constructs a claim policy *Y* to enable the cloudto authenticate the user. The creator does not send themessage MSG as is, but uses the time stamp and createsH(C)$\|$T.This is done to prevent replay attacks. If the timestamp is not sent, then the user can write previous stalemessage back to the cloud with a valid signature, evenwhen its claim policy and attributes have been revoked. Theoriginal work by Majietal.suffers from replay attacks.

In their scheme, a writer can send its message and correctsignature even when it no longer has access rights. In ourscheme a writer whose rights have been revoked cannotcreate a new signature with new time stamp and, thus,cannot write back stale information. It then signs themessage and calculates the message signature as

ϭ=ABS.Sign(Public key of trustee; Public key of KDC:token;
<div align="center">signing key; message; access claim);</div>

The following information is then sent in the cloud

<div align="center">

c=(C,T,ϭ,Y).

</div>

Special Issue - 2015

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
TITCON-2015 Conference Proceedings

The cloud on receiving the information verifies theaccess claim using the algorithm ABS.*verify*. The creatorchecks the value of V=ABS.VERIFY(TPK,σ,c,Y).if V=0,then authentication has failed and the message is discarded.Else, the message (C,T) stored in the cloud.

READING FROM THE CLOUD

When a user requests data from the cloud, the cloud sendsthe ciphertext C using SSH Protocol. Decryption proceedsusing algorithm ABE.*DECRYPT*(C,{SK$_{i,U}$ and the messageMSG is calculated

WRITING TO THE CLOUD

To write to an already existing file, the user must send itsmessage with the claim policy as done during file creation.The cloud verifies the claim policy, and only if the user is authentic, is allowed to write on the file.

USER REVOCATION

We have just discussed how to prevent replay attacks. Wewill now discuss how to handle user revocation. It shouldbe ensured that users must not have the ability to accessdata, even if they possess matching set of attributes. For thisreason, the owners should change the stored data and sendupdated information to other users. The set of attributes Iupossessed by the revoked user Uu is noted and all userschange their stored data that have attributes i 2 Iu. In,revocation involved changing the public and secret keys ofthe minimal set of attributes which are required to decryptthe data. We do not consider this approach because heredifferent data are encrypted by the same set of attributes, sosuch a minimal set of attributes is different for differentusers. Therefore, this does not apply to our model. Once the attributes Iu are identified, all data that possess theattributes are collected. For each such data record, thefollowing steps are then carried out:

1. A new value of s, s,s$_{new}$∈Zqis selected.
2. The first entry of vector v$_{new}$ is changed to new s$_{new}$.
3λ$_x$=R$_x$V$_{new}$ is calculated, for each row x correspondingto leaf attributes in Iu.
4. C1,x is recalculated for x.
5. New value of C1,x is securely transmitted to the cloud.
6. New C$_0$=Me(g,g)^{snew}is calculated and stored in thecloud.
7. New value of C1,x is not stored with the data, but istransmitted to users, who wish to decrypt the data.

We note here that the new value of C1,x is not stored inthe cloud but transmitted to the nonrevoked users who haveattribute corresponding to x. This prevents a revoked user todecrypt the new value of C0 and get back the message.

IVATTRIBUTE BASED ENCRYPTION

ABE with multiple authorities as proposed by lewko and waters proceed as follows

SYSTEM INITIALIZATION

select a prime q,generator g of G$_0$,groups G$_0$ and G$_t$ of order q, a map e:G$_0$*G$_0$->G$_T$,and a hash function H:{0,1}* -> G$_0$that maps the identities of users to G0. The hash function used herre is SHA-1.Each KDC A$_j$∈ A has a set of attributes L$_J$.The

attributes disjoint (Li∩ $Lj = \emptyset$ for i≠ j).Each KDC chooses two random exponents α$_i$,y$_i$∈ Zq.The secret key of KDC A $_j$ is
$$SK[j]=\{\alpha i, yi, i \in Lj\}.$$
The public key of KDC Aj is published
$$PK[j]=\{e(g,g)^{\alpha i}, g^{yi}, i \in Lj\}$$

KEY GENERATION AND DISTRIBUTION BY KDCS

User Uu receives a set of attributes I[j, u]from KDC Aj, andcorresponding secret key sk$_{i,u}$ for each i∈ $I[j,u]$
$$sk_{i,u}=g^{\alpha i}h(u)^{yi}$$
where α,y$_i$∈ $SK[j]$. Note that all keys are delivered to theuser securely using the user's public key, such that only thatuser can decrypt it using its secret key.

ENCRYPTION BY SENDER

The encryption function is ABE.*Encrypt*(MSG,X). Senderdecides about the access tree X. LSSS matrix R can bederived. Sender encryptsmessage MSG as follows:
1. Choose a random seed s∈Zq and a random vector v∈Z^h$_q$, with s as its first entry; h is the number of leaves in the access tree (equal to the number of rowsin the corresponding matrix R).
2. Calculate λx= Rx .v, where Rx is a row of R.
3. Choose a random vector w ∈Z^h$_q$with 0 as the firstentry.
4. Calculate ω$_x$¼=Rx .w.
5. For each row Rx of R, choose a random ρ∈Zq.
6. The following parameters are calculated:
$$c0=MSGe(g,g)^s,$$
$$c1.x=e(g,g)^{\lambda x}e(g,g)^{\alpha}\pi(x)^{\rho x}, \forall x$$
$$c2.x=g^{\rho x} \forall x,$$
$$c3.x=g^y \pi(x) \rho x g \omega x \forall x,$$
whereπ(x) is mapping from Rx to the attribute i thatis located at the corresponding leaf of the access tree.
7. The ciphertext C is sent by the sender (it also includes the access tree via R matrix):
c=<Rπ,c0,{c1.x,c2.x,c3.x,∀x}>

DECRYPTION BY RECEIVER

The decryption function isABE.*Decrypt*(C,{ski;u}), whereC is given by (5). Receiver Uu takes as input ciphertext C,secret keys {ski;u}, group G0, and outputs message *msg*. Itobtains the access matrix R and mapping πfrom C. It thenexecutes the following steps:
1. Uu calculates the set of attributes {π(x):x ∈ X} ∩ Iuthat are common to itself and the access matrix. X isthe set of rows of R.
2. For each of these attributes, it checks if there is a subset X0 of rows of R, such that the vector (1.0,......0) is their linear combination. If not, decryption is impossible. If yes, it calculates constantsCx∈ Zq, such that∑ $x \in X'CxRx$=(i,0.....0)
3. Decryption proceeds as follows:
a.for each x∈ X',dec(x)=$\frac{c1.x(h(u),c3.x)}{e(sk\pi(x)c2.x)}$
b.Uu computes MSG =c0/π$x \in X'dec(x)$

Special Issue - 2015

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
TITCON-2015 Conference Proceedings

V ATTRIBUTE-BASED SIGNATURE SCHEME

ABS scheme [24] has the following steps.

SYSTEM INITIALIZATION
Select a prime q, and groups G1 and G2, which are of order q. We define the mapping
$\hat{}e : G1 \_ G1 \text{->} G2$.
Let g1; g2 begenerators of G1 and hj be generators of G2, for j∈[tmax],for arbitrary tmax. Let H be a hash function. Let $A0 = h_0^{a0}$, where $a_0 \in Zq$ is chosen at random. (TSig.TVer) meanTSig is the private key with which a message is signed andTVer is the public key used for verification. The secret key for the trustee is TSK=(a0; TSig) and public key is
TPK =(G1,G2,H, g1,A0,h0,h1,. . . .,htmax, g2,TV er).

USER REGISTRATION
For a user with identity Uuthe KDC draws at randomKbase∈G. Let $K0= k^{1/a0}_{base}$.The following token γ is output
γ=(u,k$_{base}$,k$_0$,ρ),
whereρ is signature on u‖K$_{base}$using the signing key TSig.

KDC SETUP
Choose a,b∈ $Zq *$ randomly and compute: $A_{ij}=h^a_j$
, $B_{ij}=h^b_j$,for Ai∈A, j∈[tmax]. The private key of ith KDC isASK[i] =(a, b)and public key APK[i]=(A$_{ij}$;,B$_{ij}$|j∈ $[tmax]$).

ATTRIBUTE GENERATION
The token verification algorithm verifies the signaturecontained in γ using the signature verification key TVer inTPK. This algorithm extracts Kbase from γ using (a, b) from ASK[i] and computes$k_x=k_{base}^{1/(a+bx)}$,x∈ $j[i,u]$. Thekey Kx can be checked for consistency using algorithm ABS.*KeyCheck*(TPK;APK[i],γ,kx), which checks
$\hat{}e(k_x,A_{ij}B_{ij}^x)=\hat{}e(k_{base},h_j)$,
for all x∈ $j[i,u] and j \in [tmax]$

SIGN
The algorithm
ABS.Sign(TPK,{APK[i] : i∈ $AT[u]$},
γ,{kx:x∈ j_u},MSG,Y),
has input the public key of the trustee, the secret key of thesigner, the message to be signed and the policy claim Y. Thepolicy claim is first converted into the span program M∈ Zq. q , with rows labeled with attributes. Mx denotesrow x of M. Let π' denote the mapping from rows to the attributes. So,$\pi'(x)$ is the mapping from Mx to attribute x. A vector v is computed that satisfies the assignment {x : x ∈ J[i; u]}. Compute μh(MSG‖y).Choose r0∈ Zq and ri∈ Zq,i∈ ju and compute:
y=k$_{base}^{ro}$,si=(k$_i^{xi}$)^{r0}.(g2g1^μ)^{ri}(∀i∈ ju),
w=k0r0,pj=$\pi i \epsilon AT[u]$(A$_{ij}$B$_{ij}\pi'(i)$)^{mijri}(∀j ∈ $[t]$).
the signature is calculated as
$\sigma = (y, w, s1, s2, ... st, p1, p2 ..., pt)$

VERIFY
Algorithm
ABS.*verify*(TPK,σ=(y,w,s1,s2,...s$t, p1, p2 ..., pt$),MSG,Y),

converts Y to the corresponding monotone program M ∈ zq^{l*t}, with rows labeled with attributes.Computeμ=h(MSG‖y). If Y=1, ABS.V erify= 0 meaning false.Otherwise, the following constraints are checked

$$e\hat{}(W,A0)=?e\hat{}(y.H0),$$
$$\pi i \in e\hat{}(si, AijBij))=?\{\begin{cases} e^{(y,h1)}e^{g2g1\mu,p1}, j = 1 \\ e^{g2g1\mu,pj}, j > 1, \end{cases}$$

where i'=AT[i].

VI SECURITY OF THE PROTOCOL

In this section, we will prove the security of the protocol.We will show that our scheme authenticates a user whoequwants to write to the cloud. A user can only write providedthe cloud is able to validate its access claim. An invalid usercannot receive attributes from a KDC, if it does not have thecredentials from the trustee. If a user's credentials arerevoked, then it cannot replace data with previous staledata, thus preventing replay attacks.

Theorem 1.*Our access control scheme is secure (no outsider or cloud can decrypt ciphertext), collusion resistant and allows access only to authorized users.*

Proof. We first show that no unauthorized user can access data from the cloud. We will first prove the validity of our scheme. A user can decrypt data if and only if it has a matching set of attributes. This follows from the fact that access structure S (and hence matrix R) is constructed if and only if there exists a set of rows X0 in R, and linear constants$c_{x \in z_q}$ such that $\sum x \in X$cxRx-(1,0,.....0).we note that

$$dec(x)=\frac{c1.xe(h(u),c3.x)}{e(sk(x),u,c2.x)} =e(g,g)^\lambda e(h(u).g)^{wx}$$
Thus,
$$\pi x \in x'dec(x)$$
$$=\pi x \in x'(e(g,g)\lambda e(h(u),g\omega x$$
$$=e(g,g)^s$$

equation above holds because$\lambda_x=R_x.v$ and$\omega x = Rx.\omega$,where v.(1,0,...0)=r and $\omega.(1,0,....0) = 0c0/$ πx∈X'dec(x)=c0/e(g,g)s=M.
For an invalid user, there does not exists attributes corresponding to rows x, such that $\sum x \in 'xCx Rx$ $(1, 0, . . . , 0)$. Thus, e(g,g)^s cannot be calculated.
We next show that two or more users cannotcolludeand gain access to data that they are not individuallysupposed to access. Suppose that there exist attributes$\pi(x)$ from the colluders, such that $\sum x \in 'xCx Rx$=(1, 0, . . . , 0). However, e(h(u),g)ω needs to be calculatedaccording to (15). Since different users have differentvalues of e(h(u),g) even if they combine their attributes,they cannot decrypt the message.
We next observe that the cloud cannot decode storeddata. This is because it does not posses the secret keysski;u (by (3)). Even if it colludes with other users, itcannot decrypt data which the users cannot themselvesdecrypt, because of the above reason (same as collusionof users). The KDCs are located in different servers andare not owned by the cloud. For this reason, even if some(but not all) KDCs are compromised, the cloud cannotdecode data.

Special Issue - 2015

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
TITCON-2015 Conference Proceedings

Theorem 2.*Our authentication scheme is correct, collusionsecure, resistant to replay attacks, and protects privacy ofthe user.*

Proof. We first note that only valid users registered with thetrustee(s) receive attributes and keys from the KDCs. Auser's token is $K\gamma=(u,k_{base},k0,\rho)$where ρ is signatureon ukK_{base} with TSig belonging to the trustee. An invaliduser with a different user-id cannot create the samesignature because it does not know TSig.We next show that only a valid user with validaccess claim is only able to store the message in thecloud. This follows from the functions ABS.*Sign* andABS.*Verify*. A user who wants tocreate a file and tries to make a false access claim,cannot do so, because it will not have attribute keys $K_x$from the related KDCs. At the same time since themessage is encrypted, a user without valid accesspolicy cannot decrypt and change the information.

Table 1
NOTATIONS

Symbols	Computation
E_x	Exponentiation in group G_x
τ_H	Time to hash using function H
$\tau_{\mathcal{H}}$	Time to hash using function $\mathcal{H}$
$\tau_{P}/\tau_{\hat{P}}$	Time taken to perform 1 pairing operation in $e/\hat{e}$
$\|G\|$	Size of group G
a	Number of KDCs which contribute keys to user

Two users cannot collude and create an access policyconsisting of attributes shared between them. Suppose,there are two users A and B who have attributes x_A andx_B, respectively. They have the following informationK_{baseA},$K_{xA}$and K_{baseB},K_{xB} , respectively. A new value of

$$K_{xB}=K_{baseA}^{1/(a+bx')}$$

cannot be calculated by B, because itdoes not know the values of $\eth a$; $b\Þ$. Thus, the authentication

is collusion secure.Our scheme is resistant to replay attacks. If a writer'saccess claims are revoked, it cannot replace a data withstale information from previous writes. This is because ithas to attach a new time stamp and sign the messageH©||t again. Since it does not have attributes, it cannothave a valid signature.

VII CONCLUSION

We have presented a decentralized access control techniquewith anonymous authentication, which provides userrevocation and prevents replay attacks. The cloud doesnot know the identity of the user who stores information,but only verifies the user's credentials. Key distribution isdone in a decentralized way. One limitation is that thecloud knows the access policy for each record stored in thecloud. In future, we would like to hide the attributes andaccess policy of a user.

VIII REFERENCES

[1] S. Ruj, M. Stojmenovic, and A. Nayak, "Privacy Preserving AccessControl with Authentication for Securing Data in Clouds," Proc.IEEE/ACM Int'l Symp.Cluster, Cloud and Grid Computing, pp. 556-563, 2012.

[2] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "TowardSecure and Dependable Storage Services in Cloud Computing,"IEEE Trans. Services Computing, vol. 5, no. 2, pp. 220-232, Apr.-June 2012.

[3] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "FuzzyKeyword Search Over Encrypted Data in Cloud Computing,"Proc. IEEE INFOCOM, pp. 441-445, 2010.

[4] S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc.14th Int'l Conf. Financial Cryptography and Data Security, pp. 136-149,2010.

[5] H. Li, Y. Dai, L. Tian, and H. Yang, "Identity-Based Authenticationfor Cloud Computing," Proc. First Int'l Conf. Cloud Computing(CloudCom), pp. 157-166, 2009.

[6] C. Gentry, "A Fully Homomorphic Encryption Scheme," PhDdissertation, Stanford Univ., http://www.crypto.stanford.edu/craig, 2009.

[7] A.-R. Sadeghi, T. Schneider, and M. Winandy, "Token-BasedCloud Computing," Proc. Third Int'l Conf. Trust and TrustworthyComputing (TRUST), pp. 417-429, 2010.

[8] R.K.L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M.Kirchberg, Q. Liang, and B.S. Lee, "Trustcloud: A Frameworkfor Accountability and Trust in Cloud Computing," HP TechnicalReport HPL-2011-38, http://www.hpl.hp.com/techreports/
2011/HPL-2011-38.html, 2013.

[9] R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance: TheEssential of Bread and Butter of Data Forensics in CloudComputing," Proc. Fifth ACM Symp. Information, Computer andComm. Security (ASIACCS), pp. 282-292, 2010.

[10] D.F. Ferraiolo and D.R. Kuhn, "Role-Based Access Controls," Proc.15th Nat'l Computer Security Conf., 1992.

[11] D.R. Kuhn, E.J. Coyne, and T.R. Weil, "Adding Attributes to Role-Based Access Control," IEEE Computer, vol. 43, no. 6, pp. 79-81,June 2010.

[12] M. Li, S. Yu, K. Ren, and W. Lou, "Securing Personal HealthRecords in Cloud Computing: Patient-Centric and Fine-GrainedData Access Control in Multi-Owner Settings," Proc. Sixth Int'l
ICST Conf. Security and Privacy in Comm.Networks (SecureComm),pp. 89-106, 2010.

[13] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute Based DataSharing with Attribute Revocation," Proc. ACM Symp. Information,
Computer and Comm. Security (ASIACCS), pp. 261-270, 2010.

[14] G. Wang, Q. Liu, and J. Wu, "Hierarchical Attribute-BasedEncryption for Fine-Grained Access Control in Cloud StorageServices," Proc. 17th ACM Conf. Computer and Comm. Security
(CCS), pp. 735-737, 2010.

[15] F. Zhao, T. Nishide, and K. Sakurai, "Realizing Fine-Grained andFlexible Access Control to Outsourced Data with Attribute-BasedCryptosystems," Proc. Seventh Int'l Conf. Information SecurityPractice and Experience (ISPEC), pp. 83-97, 2011.

[16] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed AccessControl in Clouds," Proc. IEEE 10th Int'l Conf. Trust, Security andPrivacy in Computing and Communications (TrustCom), 2011.

[17] http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-cs-01-en.pdf, 2013.

[18] http://securesoftwaredev.com/2012/08//xacml-in-the-cloud,2013.

[19] S. Jahid, P. Mittal, and N. Borisov, "EASiER: Encryption-BasedAccess Control in Social networks with Efficient Revocation,"Proc. ACM Symp. Information, Computer and Comm. Security
(ASIACCS), 2011.

[20] R.L. Rivest, A. Shamir, and Y. Tauman, "How to Leak a Secret,"Proc. Seventh Int'l Conf. Theory and Application of Cryptology andInformation Security (ASIACRYPT), pp. 552-565, 2001.

[21] X. Boyen, "Mesh Signatures," Proc. 26th Ann. Int'l Conf. Advancesin Cryptology EUROCRYPT), pp. 210-227, 2007.

[22] D. Chaum and E.V. Heyst, "Group signatures," Proc. Ann. Int'lConf. Advances in Cryptology (EUROCRYPT), pp. 257-265, 1991.

[23] H.K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-BasedSignatures: Achieving Attribute-Privacy and Collusion-Resistance," JIACR Cryptology ePrint Archive, 2008.

[24] H.K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-Based Signatures," Topics in Cryptology - CT-RSA, vol. 6558, pp. 376-392,2011.

[25] A. Beimel, "Secure Schemes for Secret Sharing and Key Distribution,"PhD thesis, Technion, Haifa, 1996.

[26] A. Sahai and B. Waters, "Fuzzy Identity-Based Encryption," Proc.Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 457-473,2005.

[27] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-BasedEncryption for Fine-Grained Access Control of Encrypted Data,"Proc. ACM Conf. Computer and Comm. Security, pp. 89-98, 2006.

[28] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-PolicyAttribute-Based Encryption," Proc. IEEE Symp.Security andPrivacy, pp. 321-334, 2007.

[29] X. Liang, Z. Cao, H. Lin, and D. Xing, "Provably Secure andEfficient Bounded Ciphertext Policy Attribute Based Encryption,"Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS), pp 343-352, 2009.

[30] M. Chase, "Multi-Authority Attribute Based Encryption," Proc.Fourth Conf. Theory of Cryptography (TCC), pp. 515-534, 2007.