ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

A Survey on Concealing Access Policies of Users from Clouds in Decentralized Access Control

Sagar Laxman Gurav[1], Sachin Patil[2]

M.E. Student, Dept of CSE, ADCET, Ashta, India

Assistant Professor. Dept of IT, ADCET, Ashta , India.

ABSTRACT: Cloud computing is flexible, on-demand and low-cost uses of computing resources so that rapid developments occurring in cloud computing and services, There has been a growing trend to use the cloud for large-scale data storage. This has raised the important security issue of how to control and prevent unauthorized access to data stored in the cloud. I have proposed new decentralized access control scheme for secure data storage in cloud which uses attribute based encryption. In the proposed scheme the authenticity of users has anonymously done by trustee before the data stores on clouds. Also proposed scheme has provides features that only authorized users are able to decrypt stored data. The cloud in proposed scheme used only for store information which reduce computation and storage overhead than previous scheme.

KEYWORDS: access control; attribute-based encryption; attribute-based signature; cloud computing; cloud storage;

I. Introduction

Cloud computing is flexible ,on-demand and a promising information technology by which computing resources are provided dynamically via Internet and the data storage and computation are outsourced on cloud.Research in cloud computing is receiving a lot of attention from both academic and industrial worlds. Cloud provides different service such as PASS ,SAAS and IAAS. Data that stored in clouds is highly sensitive, for example, medical records and social networks. Security and privacy are very important issues in cloud computing. First user should authenticate itself before starting any operation, and then it must be ensured that the cloud does not change any data that is outsourced. User privacy is also required so that the cloud or other users do not know the identity of the other user.The validity of the user who stores the data is also verified[1].

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.[2] Access control is also gaining importance in online social networking where users (members) store their personal information, pictures, videos and share them with selected groups of users or communities they belong to. Access control in online social networking has been studied in [3].

It is not enough to store the contents securely in the cloud but it might also be necessary to ensure anonymity of the userFor these reason scheme uses the attribute-based signature.ABS was proposed by Maji et al.[4]. 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 trustee 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 [5],[6] on access control in cloud are centralized in nature. In 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. To overcome this failure cloud should take a decentralized approach while distributing secret keys and

DOI: 10.15680/IJIRCCE.2016. 0405208

attributes to users.while distributing secret keys and attributes to users. It has many KDCs in different locations in the world. Although Yang et al. [7] proposed a decentralized approach, their technique does not authenticate users, who want to remain anonymous while accessing the cloud. In an earlier work, Ruj and Nayak.[5] proposed a distributed access control mechanism in clouds which support access control and authentication are collusion resistant. It has also support resilient to replay attack but it has one limitation that is the cloud knows the access policy and attributes of users.

II. RELATED WORK

In SushmitaRuj and AmiyaNayak[1] proposes decentralized access control scheme. It provides many KDC in different locations in the world and they distributes keys to all users. In this system user has been registered with trustee before data stored on the clouds. System supports the creation, modification and reading data stored on the clouds. The system uses the Attribute Based Encryption (ABE) and Attribute Based Signature (ABS) protocol which uses users attributes and access policy. The access policy decides who can access the data stored in the cloud. In this system cloud storage is used to verify the authenticity of user without knowing user credential and also for storing the encrypted data that is created by creator. The scheme has also supported the access control in which valid user only decrypt the stored information. The system has one limitation that is cloud knows access policy for cloud each record stored in the clouds.
Key-Policy Attribute-Based Encryption (KP-ABE) the cipher-texts are labeled with sets of attributes and private keys are associated with access structures that control which cipher-texts a user is able to decrypt. Author has developed sharing of audit-log information and broadcast encryption. Construction of system supports delegation of private keys which is part of Hierarchical Identity-Based Encryption (HIBE).A.B. Lewko and B. Waters has suggested that any party can become an authority and there is no requirement for any global coordination. System uses Multi-Authority Attribute Based Encryption (ABE) system. A party can simply act as an ABE authority by creating a public key and issuing private keys to different users that react their attributes. A user can encrypt data in terms of any boolean formula over attributes issued from any chosen set of authorities. The system has technical problem that the system is not collision resistance when ABE authority tied together different components of a users private key by randomizing the key. To overcome this problem system bind the users key with global identifier.
H. K. Maji, M. Prabhakaran and M. Rosulek [8], worked Attribute-Based Signatures (ABS). Signatures in an ABS scheme describe a message and a predicate over the universe of attributes. A valid ABS signature shows to the fact that a single user, whose attributes satisfy the predicate, validates the message. This method takesa decentralized approach and provides authentication without disclosing the identity of users. It also provides security against a malicious attribute authority.

.

III. PROPOSED ACCESS SCHEME

A. *Scope:*

According to the proposed scheme a user can create and store it securely on the cloud. This scheme consists of use of the two protocols Attribute Based encryption (ABE) and Attribute Based Signature (ABS). For the attribute access policy the system is used the boolean format. The system has proposes the following objectives-• Hide the attribute and access policy from cloud : To hide the attribute and access policy from cloud the proposed system uses trustee (Authority). First user has to register the trustee. As result trustee generate the token and gives back to the user. Once the token has been received by user it will be authenticated by anonymously. Trustee has to perform verification of signature and encryption.
• Reduce the overhead from cloud : Perform the computation operation with the help of a trustee instead of cloud. The operation as token generation, verification of signature are performed by trustee. The cloud only used for storage purpose which will lead to reduce overhead on cloud.

B. *System Architecture*

The proposed system are four type of entities: Trustee (Authority), Cloud Server, KDCs and Users are shown in Fig.1. A user can be a creator, writer and reader. According to the system a user can create and store it securely in cloud. The trustee are assumed to have powerful computation abilities, and they are need to supervised by government offices because some attributes partially contain users personally identifiable information.

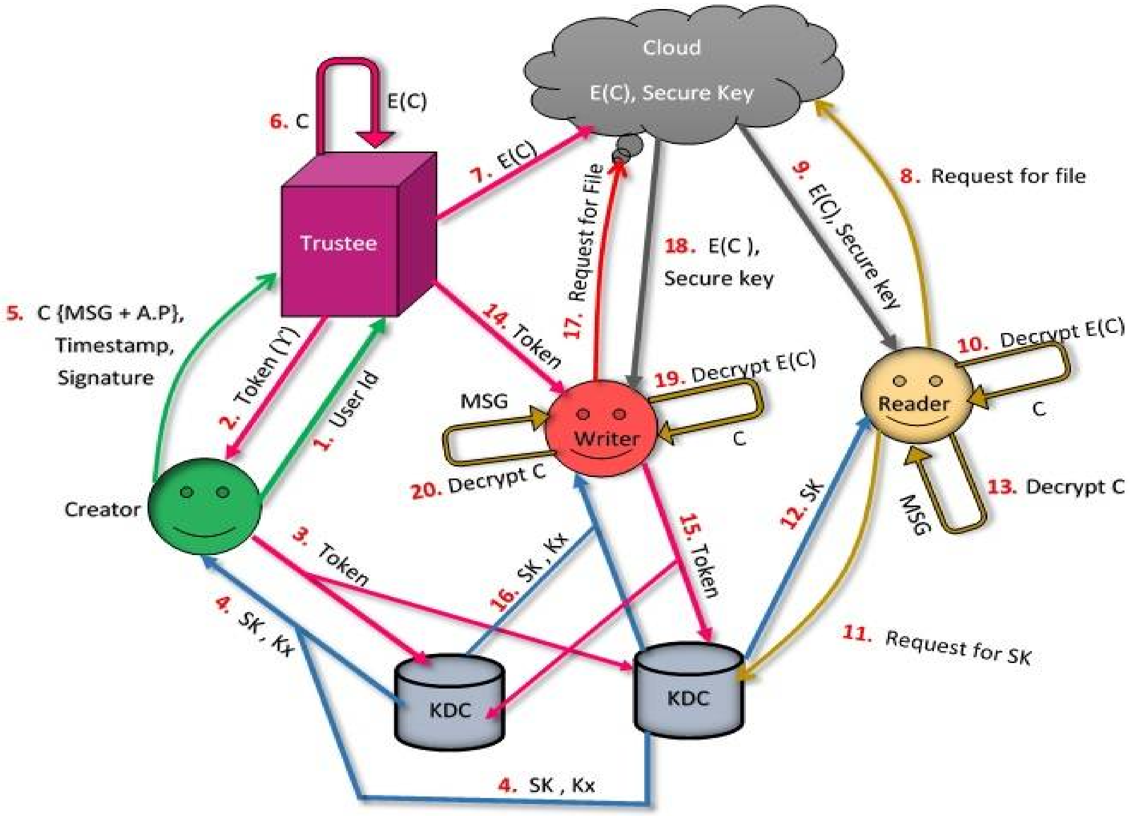


Fig. 1. An Architecture of the proposed System

The proposed system consists of ABE and ABS protocols. For the new user the system provides the authentication with the help of registration. The creator first register to trustee.As a result trustee generates token and send to user. This authentication is done with anonymously from cloud. In the system there are multiple KDC for simplicity we will consider only two KDC. After the token generation the user get keys from the one or more KDC. In Fig. 1 SKs are secret keys given for decryption, Kx are keys for signing. After getting keys the creator has encrypt themessage with access policy. Encrypted message i.e. E(MSG+ Access Policy) send to trustee along with timestamp and signature. Trustee will keep the signature and timestamp for verification and perform encryption using secure key that generates itself the trustee at the first time it logged onto system. For generation of secure key trustee use the symmetric key algorithm. The generated secure key will send to cloud and also keep on its own disk. Now E(C) is the encryption of E(MSG+ Access Policy) stored on the cloud. The creator decides on a claim policy to prove her authenticity and signs the message under this claim. The cloud stores cipher-text E(C) and secure key. The verification has done with trustee.
When reader wants read file it has to send request to the cloud. In return cloud sends encrypted cipher-text E(C) and secure key. By using this secure key the reader has to decrypt the E(C) and get cipher-text C. Reader get secrete key from KDC. Now reader has attributes matching with access policy, if it matches then it will decrypt message and get the original message. The reader cannot modify the data it will only read the data. To write data in existing file the writer has to verify the access policy from trustee if the policy matches then writer can write the data in existing file.

DOI: 10.15680/IJIRCCE.2016. 0405208

C. Approaches of proposed system

1) *Authority Setup*

The authority setup modules has perform following operations.

Secure Key Generation : At first time trustee logged onto system it generate secure key by using symmetric key algorithm. After that trustee will send secure key to cloud and also it keep secure on its own disk

Token Generation : New user has to register with trustee by using user id that is auto generated by system.Trustee will be create token using user id and send to user. Trustee use secure key for encryption that key generated at the first time trustee logged onto system. This operations show in Fig.2

2)Key Generation

This explains the generation of keys performed by the KDC. It will show in following steps

- User will send the token to each KDC.
- KDC will use token and generates the secret key (SK), public key (PK) which is used for encryption and decryption of data. It also generates KDC secret key (ASK) for signing and KDC public key (APK) for verification.
- The All keys are send to the creator , Reader get only SK key. Process showing in Fig. 3.

3)Data Stored on Cloud

This module explains the process of data storage in cloud storage. It will perform following steps-

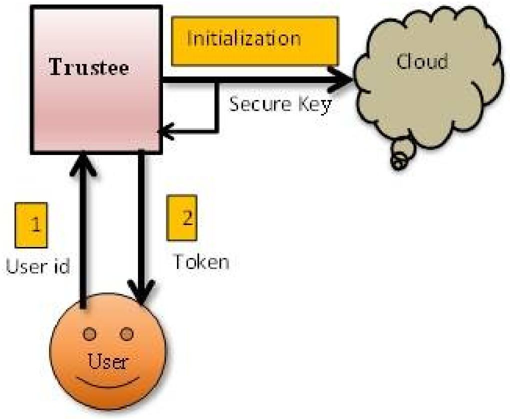
- Creator will upload data along with attributes. And this data will be encrypted by using PK that received from KDC. Creator send Encrypted message i.e. E(MSG + Access Policy) along with timestamp and signature to trustee.
- After receiving that data from creator trustee will keep the signature and timestamp on his own disk and it encrypt the cipher-text again using secure key generated by trustee itself.
- The encrypted cipher-text data has been stored on cloud.
  This operations show in Fig. 4

4)Data Read from cloud

This module explains the process of reading of data from the cloud storage. It will perform following step

- Reader send input data request to clouds.
- Cloud will send encrypted data E(C) to reader along with secure key.
- Using secure key the reader has to decrypt the cipher-text E(C) and get the C.
- User now send request to KDC for SK and check access policy.
- If the policy is matched then KDC's send the secret key to user.
- After receiving secret key the reader has decrypt cipher-text C get original message.
- This operations show in Fig. 5

DOI: 10.15680/IJIRCCE.2016. 0405208



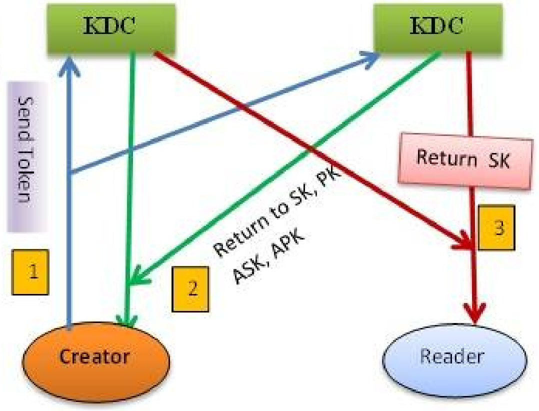
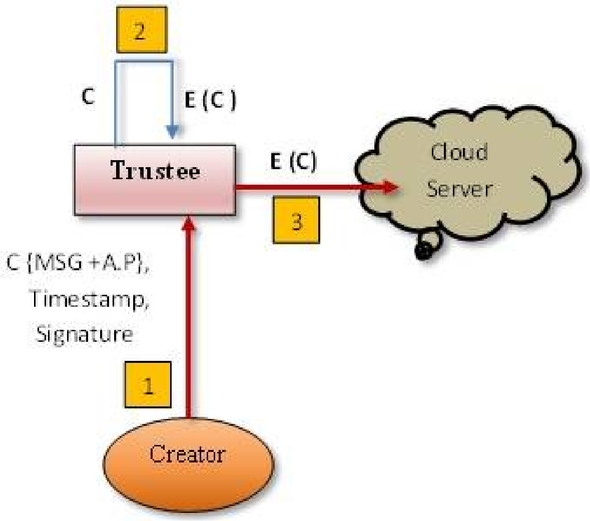


Fig-2 Process of authority setup Fig-3 Process of key generation



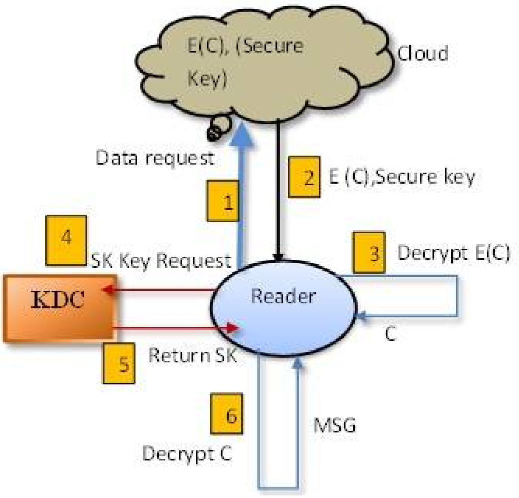


Fig-4 Process of data store on cloud fig-5 Process of data read from cloud

IV. PSEUDO CODE

A. *PSEUDO CODE FOR ACCESS POLICY GENERATION:*

```
Step 1:  Select one of  the  policy ( Pl ) either  and gate, or gate.
Step 2:  Takes  at  least 2 users attributes and at most 3 attributes (Au) from users.
Step 3:  If  (Au <2 && Au >3)
         go to  step 2
         else
                  Check the access policy (Ap).
                  If ( Ap already created )
                            Update access policy Ap.
                  else
                  Save  Access Policy Ap.
              end
          end
   Step 4:  End.
```

B. *PSEUDO CODE OF UPLOADING THE OWNER DATA ON TRUSTEE*

```
   Step 1:  Owner or creator load the data file (M) to upload.
   Step 2:  Takes public key of users (PUKu) generated by KDC and load the Ap.
   Step 3:   Use openssl-rsa encrypt to perform encryption.
          If  ( M size < Key Size)
          Encrypt ( Ap+M, PUKu );
          else
              encrypted_data=null;
              while ( M )
                  M is divides into chunks i.e (chunk size=key length/8-11).
                  Data=chunk size ( M );
                  Encrypt ( Data , PUKu , encrypted );
                  Encrypted_data +=encrypted;
              end
          end
   Step 4: Takes sign key of KDC( SignK) to generates signature(ρ) of data.
   Step 5: Generate signature by using openssl-sign.
           Sign( Encrypted_data,Signature, SignK , OPENSSL_ALGO_SHA1);
           If ( Signature not generates)
           go to step 1.
   Step 6:  Upload encrypted_data (C) and Signature(ρ) on trustee.
   Step 7:  End.
```

C. *PSEUDO CODE OF UPLOADING THE OWNER DATA ON CLOUD BY TRUSTEE*

```
   Step 1:  Trustee load the encrypted data ( C ).
   Step 2:  Takes shared key (SKt) of trustee and verification key (VerK) of KDC.
   Step 3:  Perform verification by openssl-verify with signature.
           Ok=Verify (C , ρ , VerK , OPENSSL_ALGO_SHA1);
   If  (Ok == 1)
          Perform encryption by aes-rijndeal.
          Encrypt (C , SKt);
           else
   Go to step 5.
          end
   Step 4:  Upload encrypted E(C ) data on cloud.
   Step 5:  End.
```

D. *PSEUDO CODE FOR READING THE OWNER DATA FROMCLOUD*

```
   Step 1:  Gets encrypted data E(C) from cloud.
   Step 2:  Takes shared key (SKt) of trustee and private key of users (PRKu) of KDC.
   Step 3:  Perform decryption by aes-rijndeal.
          C = Decrypt (EC , SKt);
          Else
              Go to step 6.

          end
   Step 4:  To get M perform encryption by openssl-rsa
```

DOI: 10.15680/IJIRCCE.2016. 0405208

```
        M= Decrypt (C , PUKu);
        Else
    Go to step 6.
    Step 5:  Get the user data M.
    Step 6:  End.
```

V. Conclusion and Future Work

The presented a decentralized access control technique which provides anonymous authentication of users from cloud. The user validation done before any data transaction happen. Access policy and attributes of users are hiding from cloud.The cloud is used to only stores data. Key distribution is done in a decentralized way. In future, we would like to generate keys on trustee instead of the key generation center.

References

1. SushamitaRuj, Milos Stojmenovic and AmiyaNayak, “Decentralized Access Control with Anonymous Authentication of Data Stored in Cloud”, Published by the IEEE Computer Society 2014
2. M. Li, S. Yu, K. Ren, and W. Lou, “Securing Personal Health Records in Cloud Computing: Patient-Centric and Fine-Grained Data Access Control in Multi-Owner Settings,” Proc. Sixth Int’l ICST Conf. Security and Privacy in Comm. Networks (SecureComm), pp. 89-106, 2010.
3. S. Jahid, P. Mittal, and N. Borisov, “EASiER: Encryption-Based Access Control in Social Networks with Efficient Revocation,” Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS), 2011.
4. H.K. Maji, M. Prabhakaran, and M. Rosulek, “Attribute-Based Signatures: Achieving Attribute-Privacy and Collusion-Resistance,” IACR Cryptology ePrint Archive, 2008.
5. S. Ruj, A. Nayak, and I. Stojmenovic, “DACC: Distributed Access Control in Clouds,” Proc. IEEE 10th Int’l Conf. Trust, Security and Privacy in Computing and Communications (TrustCom), 2011.
6. G. Wang, Q. Liu, and J. Wu, “Hierarchical Attribute-Based Encryption for Fine-Grained Access Control in Cloud Storage Services,” Proc. 17th ACM Conf. Computer and Comm. Security (CCS), pp. 735-737, 2010.
7. K. Yang, X. Jia, and K. Ren, “DAC-MACS: Effective Data Access Control for Multi-Authority Cloud Storage Systems,” IACR Cryptology ePrint Archive, p. 419, 2012.
8. H.K. Maji, M. Prabhakaran, and M. Rosulek, “Attribute-Based Signatures,” Topics in Cryptology - CT-RSA, vol. 6558, pp. 376-392,2011.

BIOGRAPHY

I am Sagar Laxman Gurav**.** I am M.E. student of Computer Science and Engineering Department from College of ADCET, Ashta, India. I received Bachelor of Engineering (C.S.E) degree in 2012 from ADCET, Ashta, India. My research interests is Cloud Computing.

Mr. Sachin Patil is working as Assistant Professor at Department of Information Technology, ADCET, Ashta, India.