

Trust and Reputation Based Access Control of Data in Cloud Computing

I. Ratna Jyothirmai¹, T.Swathi²P.G. Student, Department of Computer Science Engineering, G.Pulla Reddy Engineering College, Kurnool, A.P, India¹Assistant Professor, Department of Computer Science Engineering, G.Pulla Reddy Engineering College, Kurnool, A.P, India²

ABSTRACT: Cloud Computing provides many services among them storing data. Storing user's data at the cloud frees the burden of user's devices and it leads to better access of data. Due to distrust in cloud service providers data owners normally store their data in an encrypted form. But in various conditions, data need to be accessed by other entities to achieve an expected service, e.g., an eHealth service. By this control of personal data access at the cloud is a vital issue. Various applications scenarios request for flexible, fine-grained access control and better data confidentiality. Although the existing access control mechanism does not find a suitable solution. Furthermore, trust plays a main role in data sharing. Trust helps to control uncertainty and avoid the potential risks. But literature still needs a solution to access control of data based on trust and reputation. The proposed system provides an efficient access control mechanism which is applied by the data owner and reputation center in a flexible manner by implementing Attribute-Based Encryption and Proxy Re-Encryption.

KEYWORDS: Trust, Reputation Center, Access Control, Ciphertext Attribute Based Encryption, Proxy Re-Encryption

I. INTRODUCTION

Cloud Computing offers many services like storage, computing, and services to the users based on their demands. Cloud computing provides a big resource pool by linking various network resources together. It has a range of properties such as scalability, elasticity, fault-tolerance, and Pay-per-use. By this, it becomes a promising service platform. In many situations, Cloud Service Providers (CSPs) are collaborating together to give service to the user and When the data stored in the cloud the data owner has to provide access rights to the authorized users. It is necessary that data owner controls the users based on trust and reputation. Different access control mechanisms are present for protecting data access in the cloud. But these mechanisms cannot control based on trust and reputation. In the proposed scheme we are implementing Attribute-Based encryption and Proxy Re-encryption mechanisms. By applying Ciphertext Attribute-Based Encryption it allows the data owner to generate an access policy which helps data owner to evaluate the trust of the user and by using Proxy Re-encryption it allows the Reputation Center (trusted third-party) and the Data owner to generate the re-encryption keys to the users for decryption of data if the users satisfy access policy generated by the data owner. By applying two access control mechanism in a single system it achieves flexible, fine-grained access control and data confidentiality.

II. RELATED WORK

In Cloud computing, data owners upload the data into the cloud in encrypted form to ensure data privacy. The encrypted data can be decrypted by the authorized entities with permissions. To provide permissions to the authorized entities data owner has to use access control mechanisms. The existing access control mechanisms are as follows

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

ACCESS CONTROL MECHANISMS

There are many cryptographic access control mechanisms present. By adopting a traditional symmetric key cryptographic system, in this data owner combine the users into one Access Control Lists (ACLs) with similar data and then the data owner encrypts data with a symmetric key. The symmetric key will be distributed to the users in the ACL so that only the users in the ACL can access the corresponding data [1]. The main drawback of this approach is key management increases to the data owner as the number of users increases.

Another mechanism is Role-Based Access Control (RBAC). Role-Based Access Control (RBAC) was firstly proposed by Ferraiolo and Kuhn in [2]. The basic principle of RBAC is separating users and permissions by inserting different roles. Users are assigned to some roles based on their job functions and responsibilities. Each role has their corresponding operational permissions, and users can only obtain the permissions after activating their roles. By relating permissions only to roles instead of users, it is more scalable and easier to manage permissions in the cloud computing environment where a lot of users are presented, and it is difficult to track each of their identities. However, RBAC manages permissions statistically according to predefined roles, without considering some dynamic aspects such as time, and service context.

In this RBAC roles are assigned to limiting the access of information by this it controls the damage to information and there is no chance of misuse of information because each user assigned a role. The main drawback of this approach is if there is a change in the role of a user it leads to ambiguity in giving permission. RBAC mechanism cannot control the data access based on trust if two users have the same role. Fine-grained access control cannot support in RBAC.

Attribute-Based Encryption (ABE) has been recently well studied for data access control in cloud computing, because it does not require either the identities of the data requesters or their public keys, but only the attributes the requesters own [3]. ABE was further extended to two varieties, Key Policy-Attribute Based Encryption (KP-ABE) and Ciphertext Policy-Attribute Based Encryption (CP-ABE).

Key Policy-Attribute Based Encryption (KP-ABE) was proposed by Goyal in 2006. In KP-ABE, a ciphertext is associated with a set of attributes during the encryption, while a secret key is generated based on an access policy, which is an access structure over a set of data attributes. To decrypt the ciphertext, the data attributes must satisfy the access structure. Although KP-ABE is believed secure for data access control in cloud computing, it comes with a high computational cost, especially if there is a large number of attributes. Moreover, the scheme also allows data owners to delegate most of the computation tasks to untrusted cloud servers without disclosing the underlying data contents.

The Ciphertext Policy-Attribute Based Encryption (CP-ABE) was proposed after KP-ABE, by Bethencourt[4]. Different from KP-ABE, a cipher text in CP-ABE is built upon an access policy, while a private key is associated with a set of attributes. A user can decrypt the data if only if attributes embedded in the ciphertext are matched the with access policy. CP-ABE scheme to reduce the computational burden of user revocation by removing the computations of revocation at cloud service providers. By this Ciphertext policy based ABE that has become more popular.

The CP-ABE is good for to control the data access but if a user is revoked the data owner has to provide re-distribute new keys to authorized users it leads to the heavy workload to the data owner. A better solution to this is Proxy Re-encryption [5]. The Proxy Re-encryption schemes are cryptosystems which allow third parties (proxies) to alter a cipher text which has been encrypted for one party so that it may be decrypted by another.

Reputation Center (RC) is a trust third party which is also used to control the data access [5]. The RC is always available for a Data owner for registration and specifying data access policies. When the Data owner is not available to give access permissions RC takes responsibility to provide access permissions to the authorized user's based on access policy.

All the existing access control mechanism individually does not control effectively in order to overcome this we are implementing two access control mechanism in a single system it will give the better result. In our proposed system policy based access control and re-encryption of keys are implemented at the Data owner and Reputation Center.

Algorithms used in Proposed System

A Ciphertext-policy attribute based encryption scheme consists of four fundamental algorithms: Setup, Encryption, Key Generation, and Decryption.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

- Setup (k): This algorithm takes input a security parameter and provides a set of public parameter(Pk) and the master key (Mk).
 - Encryption(M,T,PK): This algorithm takes input takes as input the message to be encrypted(M), the access structure T which needs to be satisfied and the public parameters (PK) to output the ciphertext CT.
 - Key-Generation(MK,PK,A): This algorithm takes input as the master key values(MK), the public parameters(PK) and the attribute set of the user(A) and output for the user a set of decryption keys SK.
 - Decryption(CT,SK,PK): This algorithm takes input as the ciphertext CT, the user secret key SK and the public parameters PK and outputs the message M.
- Proxy Re-Encryption consists of four fundamental algorithms KeyGen, RekeyGen, Encryption, and Decryption.
- KeyGen: This KeyGen generates algorithm out put a public and secret key (PK,SK)
 - RekeyGen(SK1,SK2):It takes input as t secret keys SK1and SK2 and generates a re-encryption key rk1 \rightarrow 2
 - Encryption(PK,M): It takes input as public key Pk and Message M and outputs a Ciphertext C1
 - Re-Encryption(rk1 \rightarrow 2,C1):It takes input as re-encryption key rk1 \rightarrow 2 and a Ciphertext C1and outputs a Second ciphertext C2
 - Decryption(SK,C):It takes input as secret key SK and a ciphertext C and outputs message M

III. PROPOSED SYSTEM

The proposed system is used to control the data access based on trust and reputation. The entities present in the system are Data Owners, Cloud Service Providers (CSP), Reputation Center (RC), Private Key Generator (PKG) and Users.

- **Data Owners (u1):** The data owners who store data in the Cloud and have a right of data access control and altering. To secure data over the cloud and to control the data access based on trust levels data owner applying ABE. By applying ABE the owner can issue to a number of eligible users by performing encryption computation only once. The scheme is implemented with Ciphertext attribute based encryption (CP-ABE).In CP-ABE access structure is embedded with Ciphertext and only the users who satisfy the access policy are allowed to access the data.
- **Cloud Service Provider (CSP):** The CSP is responsible for data storage and forwarding the requests. The data stored at the CSP is in the form of encrypted which is send by the data owner. The CSP forward the request of the user to Data owner and RC.
- **Reputation Center (RC):** The RC is fully trusted and employed for reputation management, as well as helping the data owners check if the users meet the access policies and generates reputation to the authorized users. At the Reputation Center in order to control the access of data it applies Proxy Re-Encryption.
- **Private Key Generator:** The PKG is responsible for the generation of keys Public keys, Master keys and Secret keys for Data owner, Reputation Center and users. The Public key is used for encrypting the data. Master key is used for generating secret keys to the each user. The Secret key is used to decrypt the data.
- **Users (u2):** The Users are the ones who request for data access.

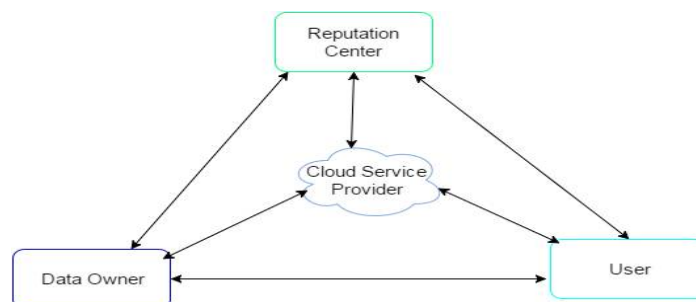


Fig.1 A System Model

Proposed Scheme:

In the Proposed System, it provides flexible control data access based by applying Attribute-based Encryption and Proxy Re-Encryption.

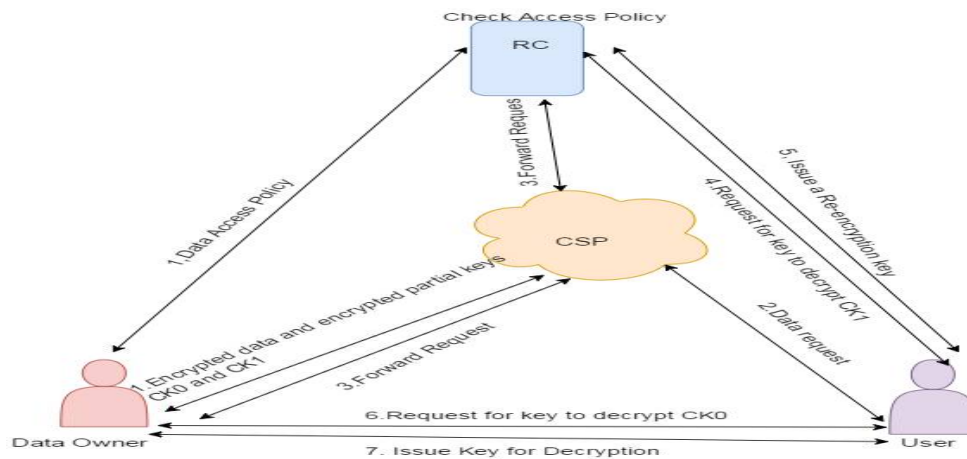


Fig.2. Procedure of data access control

In this proposed scheme it is possible multi-dimensional controls on cloud data access based on the policies and strategies set by the data owner. The steps involved in the proposed scheme are as follows

1. The Data owner(u1) encrypts its data with a symmetric secret key K. The access policy (AA) is defined by the data owner is send to RC. The data (M) is encrypted with Symmetric key K and the access policy (AA) is set to the ciphertext to control data access by using CP-ABE. The Symmetric key K is divided into two parts K0 and K1 in order to support various control strategies. The partial Keys K0 and K1 are again encrypted with the Public key of Data owner (PK_u1) and Reputation center(PK_RC) and generates CK0 and CK1.The encrypted data (CT) and partial encrypted Keys (CK0 and CK1) are sent to the Cloud Service Provider for storing.
2. When a User(u2) send a request to CSP for accessing of data.
3. The CSP forward the request to Data owner and RC.
4. The user sends a request to RC for the decryption of CK1 because it is encrypted with Reputation Center public key(PK_RC) to decrypt CK1.
5. RC check the access policy(AA) which is sent by the Data owner(u1) if User(u2) satisfies the access policy then RC first decrypt CK1 with secret key (SK_RC) and then encrypted with User’s public key(PK_u2) here proxy re-encryption is performed by the RC and generates . Then User(u2) decrypt CK1 with his/her own secret key(SK_u2) by this K1 is achieved.
6. In order to decrypt data complete key is required for that User(u2) sends a request for decryption of another partial key CK0 to Data owner(u1).
7. Then the Data owner checks the policy and counts the number of attributes satisfied. If User satisfied all attributes the Data owner consider him/her as trusted one and decrypt the CK0 first with secret key(SK_u1) and encrypt the User’s public key. Then User decrypt the partial key CK0 with secret key (SK_u2) K0 is achieved. By combing the both keys K0 and K1 the User(u2) get Symmetric key K. The Symmetric key is used to decrypt the ciphertext CT and then User(u2) get data(M).

The proposed scheme is flexible if Data owner would like to control only by itself, set K1=null and K0=K. If it would like to control by Reputation Center set K1=K and K0=null. If the Data owner would like to control by Reputation Center neither K1 nor K0 is null.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

Table 1: Summary of Keys Applied

Keys	Description
K	The Data encryption key
MK	Master key to generate secrete keys
PK_u1	Public key of Data owner
SK_u1	Secrete key of Data owner
PK_u2	Public key of User
SK_u2	Secrete key of User
PK_RC	Public of Reputation Center
SK_RC	Secrete key of Reputation Center
rk_RC→u2	Re-encryption key for User

Scheme algorithms:

The proposed scheme consists of following algorithms are as follows

- **Setup:** The setup algorithm is used for generating the public keys (PK) and the master key (MK) for the users and also generates the public key (PK_RC) and private key(SK_RC) to Reputation center.
- **ABEUserkeyGeneration(PK,MK, u):** This algorithm takes inputs Public key, Master Key and user identity u and generates secret key SK_u. By using the secret key user can decrypt the data.
- **PREUserKeyGeneration(u):** This algorithm also used to generate public key PK_u and private key SK_u for Proxy Re-encryption.
- **CreateEncryptionKey():** This algorithm is used to generate Symmetric key K for data encryption it is performed by the data owner.
- **Divide key (K, n):** The Dividekey algorithm divides input K into n+ 1 part.
- **CombineKey(K0,K1,...Kn):** The Combinekey algorithm aggregates partial keys (K0,K1...Kn) to get complete key K
- **Encrypt1(PK_RC,K1):**This algorithm encrypts K1 with Reputation Center public key PK_RC and outputs CK1.
- **ReencryptionKeyGeneration(PK_RC,SK_RC,PK_u2):** This algorithm takes inputs PK_RC, SK_RC and PK_u2 generates a re-encryption key rk_RC u2 for u2 if it satisfies access policy generated Data owner(u1)
- **Decrypt1(SK_u2,rk_RC u2):** Inputs as secret of user SK_u2 and re-encryption key rk_RC u2 and to outputs plain key K1
- **Encrypt0(PK_u1,K0):** This algorithm takes inputs public key of Data owner Pk_u1,partial key K0 and then K0 with these PK_u1 and outputs cipher-key CK0.
- **Decrypt0(CK0,AA,SK_u1,PK_u2,SK_u2):** This algorithm is used to decrypt the CK0 with data owner SK_u1 if user satisfies the access policy and then CK0 re-encrypt with user public key PK_u2 then user decrypt with SK_u2 and outputs a plain key K0.
- **Encrypt(K,M,AA):** This algorithm takes input K, data M, and access policy AA to get encrypted data CT.
- **Decrypt(CT,K):** This algorithm takes input a ciphertext CT and encryption K to output the plaintext M.

Advantages:

Dynamic Authorization: Our scheme supports dynamic authorization if a user is revoked instead of re-encrypts the data simply change the access policy.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

Fine-grained access control: The Proposed scheme achieves fine-grained access control. Various access policies are enforced in the scheme. We remove the burden evaluating the trust of the user by using CP-ABE. The reputation center reduces the computation burden of data owners.

Efficient and Flexibility: This scheme releases the re-encryption load to Cloud Service Provider. With our scheme data access is controlled by the data owner and/ or by reputation center.

Data Confidentiality: Data Confidentiality is achieved by encrypting the Symmetric key and plain data is only get when two partial keys are decrypted. The proposed scheme supports data storage at CSP in a secure way and enhances user data privacy by hiding the plain data from the CSP and RCs. It is hard for the CSP to know the plaintext of the user data. In this case, even though CSP and RC really collude, it is still impossible for the RC to know the complete key and get the plain data.

IV. RESULTS

The below figure 3 shows the number of entities present in our proposed system. In our system there are five entities data owner, user, private key generator, and reputation center and cloud service provider.

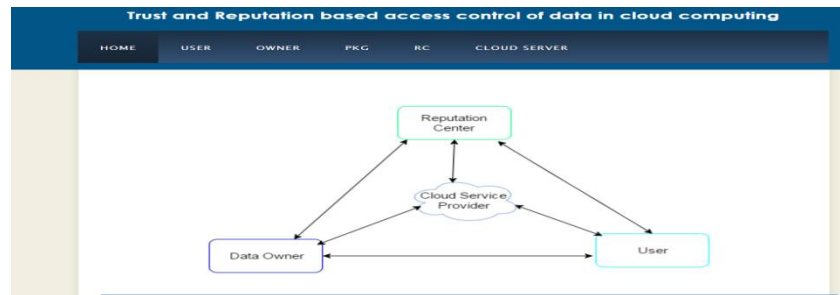


Fig.3 Entities in the system

The below figures 4 and 5 shows the registration of data owner and user each have same fields but attributes are different. When a user or data owner are registered their attributes are stored in the cloud

Name :	Rani
Username :	Rani
Password:	****
Gender :	FeMale
Date of Join :	15-07-1993
Experience :	4
Branch :	Pune
Department :	Engineering
SubDepartment :	Programming
E-mail Id:	rani@gmail.com

Fig.4 Data owner registration

Name :	Sandhya
Username :	Sandhya
Password:	*****
Gender :	FeMale
Date of Join :	14-04-1995
Experience :	5
Branch :	Delhi
Department :	Engineering
SubDepartment :	Programming
E-mail Id:	sandhya@gmail.com

Fig.5 User registration

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

The below figures 6 and 7 shows the generation of Public key Master key and Secret key to the data owner and same process for user and reputation center. The public key is used to encrypt the data and Master key is used to generate the secret key to the users with set of attributes. The secret key is used for decryption of data.



Fig.6 Public and Master key generation

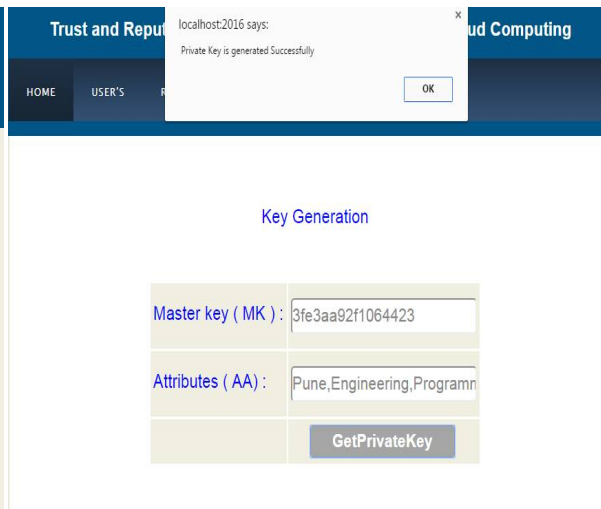


Fig.7 Secret key generation

The below figures 8 and 9 shows data owner upload a file in the cloud with a unique id and file details and generation of symmetric key. The symmetric key randomly generated key by the data owner which is used for encrypting the data

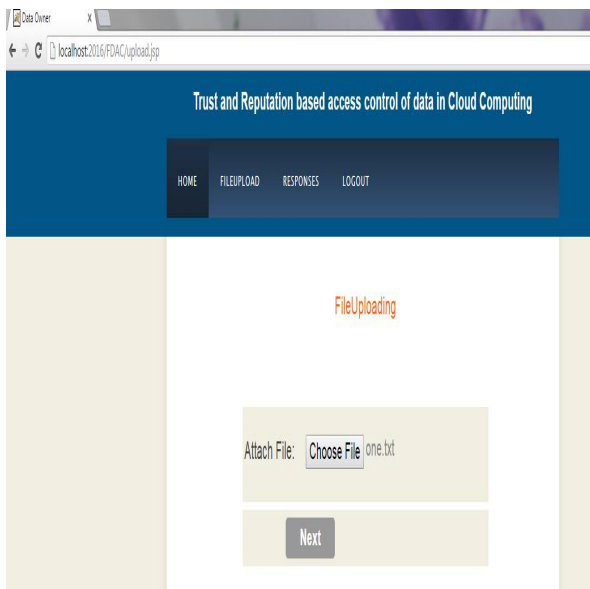


Fig.8 Uploading a file

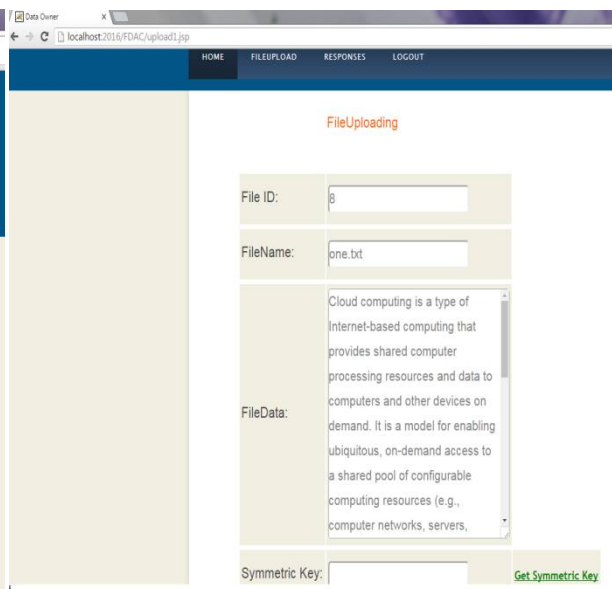


Fig.9 File details and Symmetric key generation

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

Figure 10 shows the Symmetric key generation and data owner defines access policy for authorized users and then data is encrypted. Here we are using CP-ABE so access policy is embedded with the ciphertext.



Fig.10 Access policy creation

Below Figure11 shows encryption of data and division/ splitting of Symmetric key K in to K0 and K1



Fig.11 Division/Splitting of symmetric key K

Figure 12 and 13 depicts the encryption of K0 with the public key of data owner and K1 with public key of reputation center.

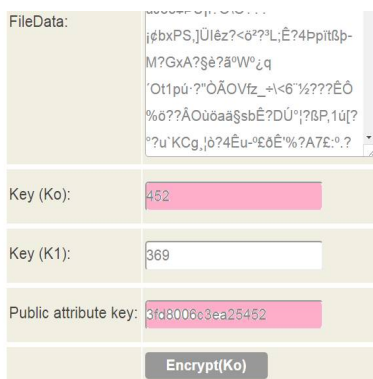


Fig.12 Partial key K0 encryption

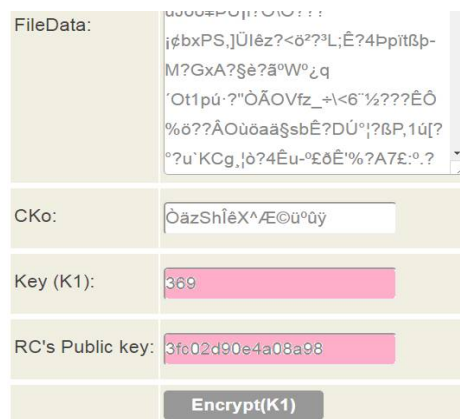


Fig.13 Partial key K1 encryption

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

Below figure 14 shows data owner upload the file and encrypted partial keys.

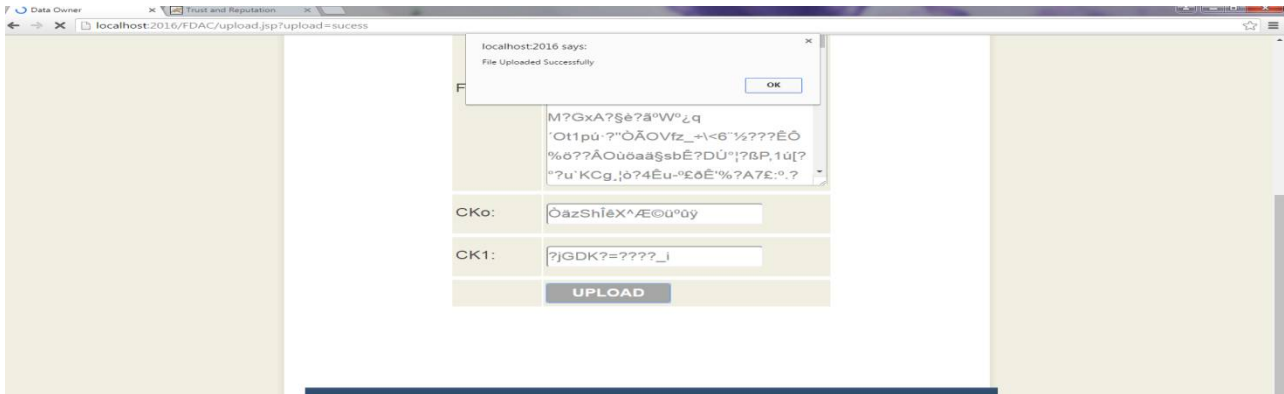


Fig.14 Encrypted data and Partial encrypted keys uploading

The figure15 below shows the user login and wants to download the file which is encrypted form. In order to download the file requires partial key decryption.



Fig.15 User checks file details

The figure 16 depicts user send a request to the reputation center for decryption of CK1 as it is encrypted with the public key of reputation center.



Fig.16 User send request to RC for decryption of CK1

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

Figure 17 shows that reputation center check access policy of the user sent by the data owner if user satisfied then reputation center first decrypts CK1 with secret key and then encrypted with public key of user(issue re-encryption key). Here we are using Proxy Re-encryption

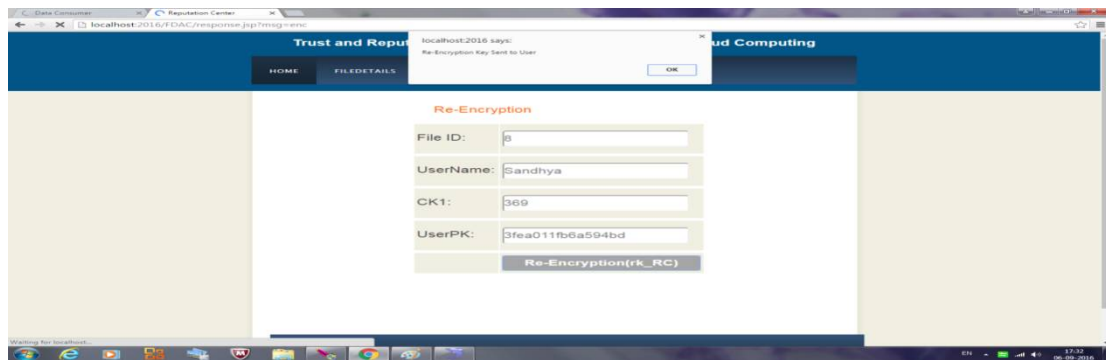


Fig. 17 RC generates re-encryption key

The below figure 18 shows the decryption of partial key CK1 with secret key of the user



Fig.18 Decryption of CK1

Figures 19 and 20 shows user request the data owner for decryption of CK0 and then data owner check if user satisfies all attributes consider the user as trusted one and generate key for decryption of CK0

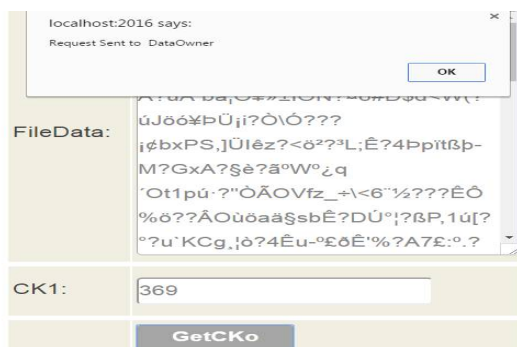


Fig.19 User send request to data owner

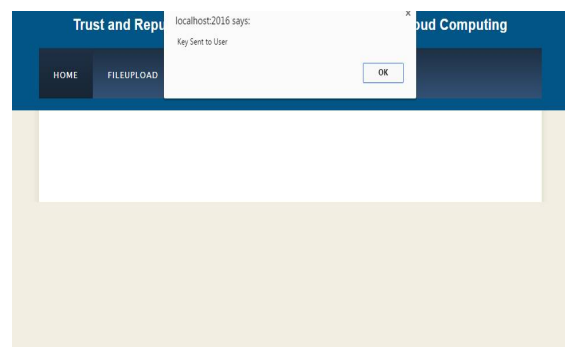


Fig.20 Data owner send response to user

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 10, October 2016

The below figure depict the user achieves full key K and download the file

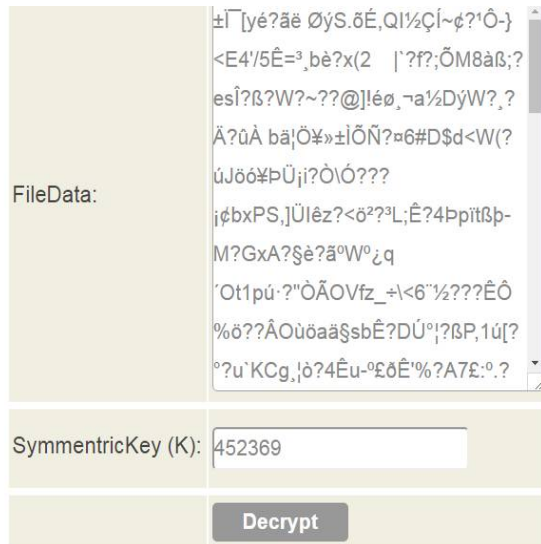


Fig.21 Achieves complete symmetric key K

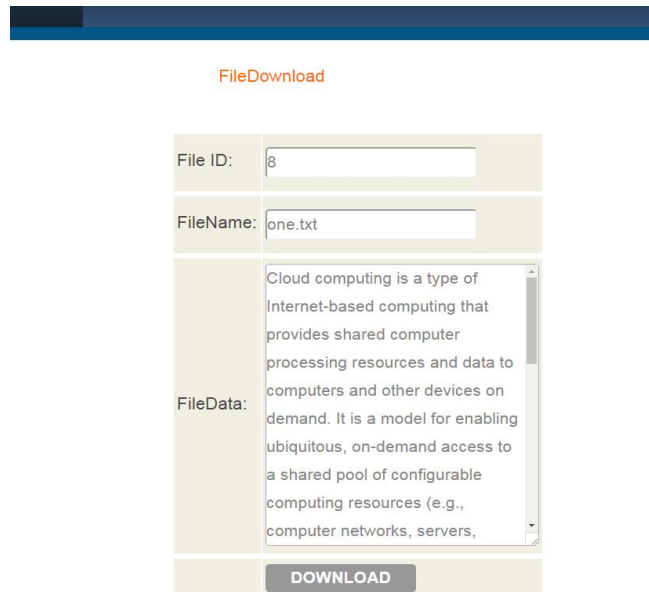


Fig.22 user download the file

V. CONCLUSION

To protect the data and privacy a number of schemes have proposed for data access control in cloud computing, however, all schemes are not effective in the case of trust and reputation. We proposed a scheme to control cloud data access based on trust and reputation its supports various access control strategies by applying Attribute-Based encryption and Proxy Re-encryption. The proposed scheme proves highly efficient, flexible and achieves better confidentiality.

REFERENCES

- [1] M. Kallahalla, "Plutus: Scalable secure file sharing on untrusted storage", Proc. of the USENIX Conference on File and Storage Technologies (FAST), pp. 29-42, 2003
- [2] D.F.Ferraiolo and D.R.Kuhn, "Role-Based Access Control", 15th National Computer Security Conference, P.554, Oct.1992.
- [3] V.Goyal, O.Pandey, A.Sahai, and B.Waters, "Attribute-based encryption for fine-grained access control of encrypted data", in Proceedings of the 13th ACM conference on Computer and communications security, ACM P.89, ISBN 1-59593-518-5, 2006.
- [4] J.Bethencourt, A.Sahai, and B.Waters, "Ciphertext-policy attribute based encryption", IEEE symposium on Security and Privacy, IEEE P.321, ISBN 0-7695-2848-1, May 2007.
- [5] Z. Yan, X. Li, and R. Kantola, "Controlling cloud data access based on reputation", Mobile Networks and Applications, Springer, 2015
- [6] T. Zhu, W. Liu, and J. Song, "An efficient role based access control system for cloud computing", Proc. of IEEE CIT2011, pp. 97-102, 2011.
- [7] Z. Yan, X. Li, and R. Kantola, "Personal data access based on trust assessment in mobile social networking", in Proc. of IEEE TrustCom2014, pp. 989 - 994, 2014.
- [8] W. Wang, J. Han, M. Song, and X. Wang, "The design of a trust and role based access control model in cloud computing", in Proc. of 6th International Conference on Pervasive Computing and Applications, pp. 300-334, 2011.
- [9] S. Yu, C. Wang, K. Ren, W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing", Proc. of the IEEE INFOCOM, pp. 534-542, 2010.
- [10] Z. Yan, Y. Chen, and Y. Shen, "A practical reputation system for pervasive social chatting", Computer and System Sciences, vol. 79, no. 5, pp. 556-572, 2013.