Effective Data Access Control in Mobile Cloud Computing Using LDSS CP-ABE

Anusha. R, S Meenakshi Sundaram*, Asha. P, Bhargavi. Y, Bindu Jayaram

Department of CSE, GSSSIETW, Mysuru, Karnataka, India

DOI: https://doi.org/10.21467/proceedings.1.2
* Corresponding author email: hodcse@gsss.edu.in

Abstract

Mobile Cloud Computing is an information technology paradigm that enables ubiquitous access to shared pools of configurable system resources provisioned with minimum management effort. Since the personal data is being stored and retrieved from the cloud. Data security problem become an obstacle for further developments. Although substantial studies have been conducted to improve the cloud security, most of them are not applicable for mobile cloud, as these mobile devices are generally less powerful and consume more energy. Solutions in with low computational overhead are in great need. In this paper, we have proposed a lightweight data secure scheme(LDSS) for mobile cloud computing. It adopts the CP-ABE, an access control technology used in normal cloud environment, but changes the structure of access control tree to mobile it suitable for mobile cloud environment. External proxy servers are being introduced for mobile devices to reduce computational overhead. The probing outcome of the scheme LDSS, effectively reduces the overhead on the mobile devices.

Index Terms- Mobile cloud computing data encryption, access control**.**

1 Introduction

Cloud computing is the buzz word now in the field of information technology. It is the concept where an organisation has its data and application hosted on third party infrastructure. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the internet as managed third-party services. These services typically provide access to advanced software application and high-end network of server computers.

Mobile devices are increasingly becoming an essential part of human life as the most effective and convenient communication tools not bounded by time and place. However, with mobility comes its inherent problem such as resource scarceness, finite energy. In such a circumstance to achieve the satisfactory performance. It is essential to use the resources provided by the cloud service provider to share the data.

The large benefit of using a CSP comes in efficiency and economies of scale. Rather than individuals and companies building their own infrastructure they support internal services and applications; the services can be purchased from the CSP whic provide the services to

 ISBN: 978-81-936820-0-5

many customers from a shared infrastructure. The access control mechanism provided by the CSP are either not sufficient or not very convenient. First, when people upload their data files onto the cloud there are possibilities of data breaching. Secondly, providing access permission to certain people/group by sending passwords to them becomes very inconvenient for users.

Evidently, to solve the above problems, personal sensitive data should be encrypted before uploaded onto the cloud so that the data is secure against the CSP. However, the data encryption brings new problems. How to provide efficient access control mechanism on ciphertext decryption so that only the authorized users can access the plaintext data is challenging. In addition, the system must offer data owners effective user privilege management capability, so they can grant/revoke data access privileges easily on the data users. There have been substantial researches on the issue of data access control over ciphertext. In these researches, they have the following common assumptions. First, the CSP is considered honest and curious. Second, all the sensitive data are encrypted before uploaded to the Cloud. Third, user authorization on certain data is achieved through encryption/decryption key distribution. In general, we can divide these approaches into four categories: simple ciphertext access control, hierarchical access control, access control based on fully homomorphic encryption and access control based on attribute-based encryption (ABE). All these proposals are designed for non-mobile cloud environment. They consume large amount of storage and computation resources, which are not available for mobile devices. The basic ABE operations take much longer time on mobile devices than laptop or desktop computers. Furthermore, current solutions don't solve the user privilege change problem very well. Such an operation could result in very high revocation cost. This is not applicable for mobile devices as well. Clearly, there is no proper solution which can effectively solve the secure data sharing problem in mobile cloud. As the mobile cloud becomes more and more popular, providing an efficient secure data sharing mechanism in mobile cloud is in urgent need.

To address this issue, in this paper, we propose a Lightweight Data Sharing Scheme (LDSS) for mobile cloud computing environment.

The main contributions of LDSS are as follows:

(1) We have designed an algorithm called LDSS-CP-ABE based on Attribute-Based Encryption (ABE) method to offer efficient access control over ciphertext.

(2) We have use proxy servers for encryption and decryption operations. In our approach, computational intensive operations in ABE are conducted on proxy servers, which greatly reduce the computational overhead on client- side mobile devices. Meanwhile, in LDSS-CP-ABE, to maintain data privacy, a version attribute is also added to the access structure. The decryption key format is modified so that it can be sent to the proxy servers in a secure way.

(3) We have introduced lazy re-encryption and description field of attributes to reduce the

revocation overhead when dealing with the user revocation problem.

(4) Finally, we have implemented a data sharing prototype framework based on LDSS. The experiments show that LDSS can greatly reduce the overhead on the client side, which only introduces a minimal additional cost on the server side. Such an approach is beneficial to implement a realistic data sharing security scheme on mobile devices.

2 Literature Survey

A literature survey is a text of a scholarly paper, which includes the current knowledge including substantive findings, as well as theoretical and methodological contributions of a particular topic.

- A. Sahai And B. Waters [1] have introduced a new type of Identity-Based Encryption (IBE) scheme that they called Fuzzy Identity-Based Encryption. In Fuzzy IBE they viewed an identity as set of descriptive attributes.
- V. Goyal, O. Pandey, A. Sahai and B. Waters [2] have introduced Attribute-based encryption for fine-grained access control of encrypted data. As more sensitive data is shared and stored by third-party sites on the Internet, there will be a need to encrypt data stored at these sites.
- R. Ostrovsky, A. Sahai and B. Waters [3] introduces Attribute-based encryption with non-monotonic access structures. They construct an Attribute-Based Encryption (ABE) scheme that allows a user's private key to be expressed in terms of any access formula over attributes.
- B. Waters [4] introduces Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. He presented a new methodology for realizing Ciphertext-Policy Attribute Encryption (CP- ABE) under concrete and noninteractive cryptographic assumptions in the standard model.
- A.B. Lewko, T. Okamoto, A. Sahai, K. Takashaima and B. Waters [5] introduces Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. They presented two fully secure functional encryption schemes: a fully secure attribute-based encryption (ABE) scheme and a fully secure (attribute-hiding) predicate encryption (PE) scheme for inner-product predicates.
- Qihua Wang, Hongxia Jin [6] introduces Data leakage mitigation for discertionary access control in collaboration clouds. With the growing popularity of cloud computing, more and more enterprises are migrating their collaboration platforms from in-enterprise systems to Software as a Service (SaaS) applications.
- Wang W, Li Z, Owens R, et al [7] introduces Secure and efficient access to outsourced data. Providing secure and efficient access to large scale outsourced data is an important component of cloud computing.
- Yu S., Wang C., Ren K., Lou W [8] introduces Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing. Cloud computing is an emerging

ISBN: 978-81-936820-0-5

Series: AIJR Proceedings
Proceedings DOI: 10.21467/proceedings.1

computing paradigm in which resources of the computing infrastructure are provided as services over the Internet.

- Kan Yang, Xiaohua Jia, Kui Ren [9] introduces Attribute-based fine-grained access control with efficient revocation in cloud storage systems. A cloud storage service allows data owner to outsource their data to the cloud and through which provide the data access to the users.
- Crampton J, Martin K, Wild P [10] introduces on key assignment for hierarchical access control. A key assignment scheme is a cryptographic technique for implementing an information flow policy, sometimes known as hierarchical access control.
- Maheshwari U, Vingralek R, Shapiro W [11] introduces How to build a trusted database system on untrusted storage. Some emerging applications require programs to maintain sensitive state on untrusted hosts.
- Kan Yang, Xiaohua Jia, Kui Ren, Bo Zhang, Ruitao Xie: DAC-MACS [12] introduces Effective Data Access Control for Multiauthority Cloud Storage Systems. Data access control is an effective way to ensure data security in the cloud. However, due to data outsourcing and untrusted cloud servers, the data access control becomes a challenging issue in cloud storage systems.
- Liang Xiaohui, Cao Zhenfu, Lin Huang, et al [13] introduces Attribute based proxy re-encryption with delegating capabilities. Attribute based proxy re-encryption scheme (ABPRE) is a new cryptographic primitive which extends the traditional proxy re-encryption (public key or identity-based cryptosystem) to the attribute- based counterpart, and thus empower users with delegating capability in the access control environment.
- D. Huang, X. Zhang, M. Kang, and J. Luo [14] introduces Mobicloud A secure mobile cloud framework for pervasive mobile computing and communication Cloud provides the environment for the mobile users called mobicloud to performs computationally intensive operation such as searching, data mining, and multimedia processing.
- Kan Yang, Xiaohua Jia, Kui Ren, Ruitao Xie, Liusheng Huang [15] introduces Enabling efficient access control with dynamic policy updating for big data in the cloud. Due to the high volume and velocity of big data, it is an effective option to store big data in the cloud, because the cloud has capabilities of storing big data and processing high volume of user access requests.

3 Proposed Approach

We propose LDSS, a framework of lightweight data sharing scheme in mobile cloud. It has the following six components.

1. Data Owner (DO): DO upload data to the mobile cloud and share it with friends. DO determine the control policies.
2. Data User (DU): DU retrieves data from the mobile cloud.

3. Trust Authority (TA): TA is responsible for generating and distributing attribute keys.
4. Encryption Service Provider (ESP): ESP provides data encryption operations for DO.
5. Decryption Service Provider (DSP): DSP provides data decryption operations for DU.
6. Cloud Service Provider (CSP): CSP stores the data for DO. It faithfully executes the operations requested by

DO, while it may peek over data that DO has stored in the cloud.DO send data to the cloud. Since the cloud is not completely reliable. Hence the data has to be encrypted before being uploaded. The DU who desires to access certain data file must satisfy the policies assigned by the DO in the form of access control tree on those data files. In LDSS, data files are all encrypted with the symmetric encryption mechanism, and the symmetric key for data encryption is also encrypted using attribute-based encryption (ABE). The access control policy is embedded in the cipher text of the symmetric key. To reduce encumbrance of encryption and decryption process for the mobile users Encryption Service Provider[ESP]

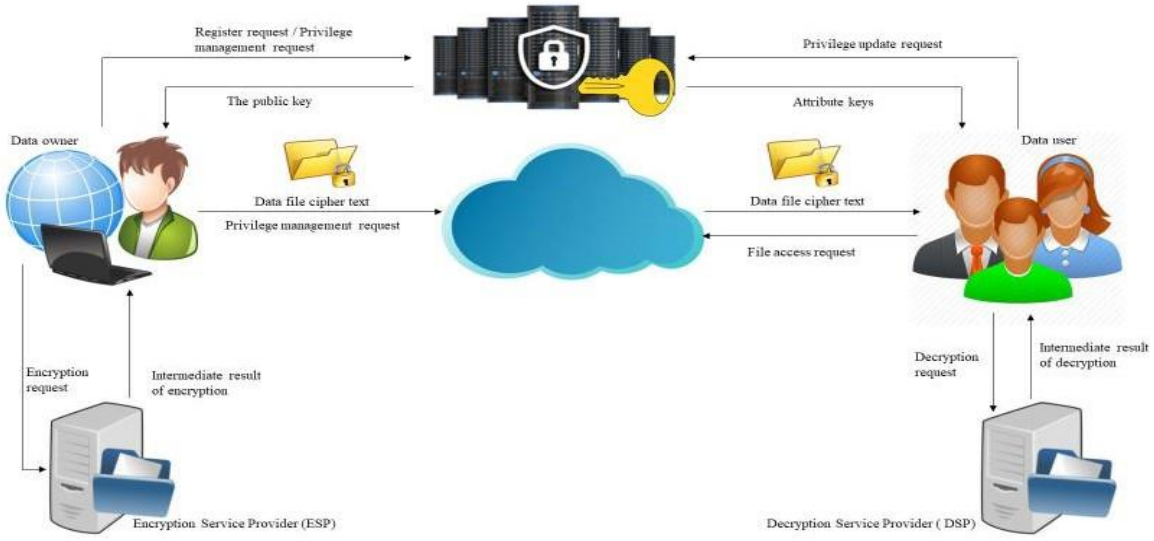

FIG 1. SYSTEM ARCHITECTURE

3.1 Data Owner

- At the first step a new data owner register himself and the get logged in.
- DO request the trusted authority for the public key, on receiving the key he uploads the encrypted file to the cloud.
- DO determines the access control policies on data files being uploaded.
- In later stages, DO responds to the decryption key request sent by the DU.
- DO can also view the files he has uploaded.

3.2 Data User

- First the new data user register himself and then login

ISBN: 978-81-936820-0-5
Proceedings DOI: 10.21467/proceedings.1

- The attribute key request is sent to trusted authority, on receiving one will be able to view the uploaded file in the cloud.
- The decryption key request is sent to cloud.
- On policy satisfaction as determined by the DO the desired data file can be downloaded.

3.3 Trusted Authority

- The logged in trusted authority, has the provision to view both owner and user.
- The trusted authority is responsible for generating the keys as per the request from owner and user

3.4 Cloud

The logged in cloud has the privileges to view the uploaded files

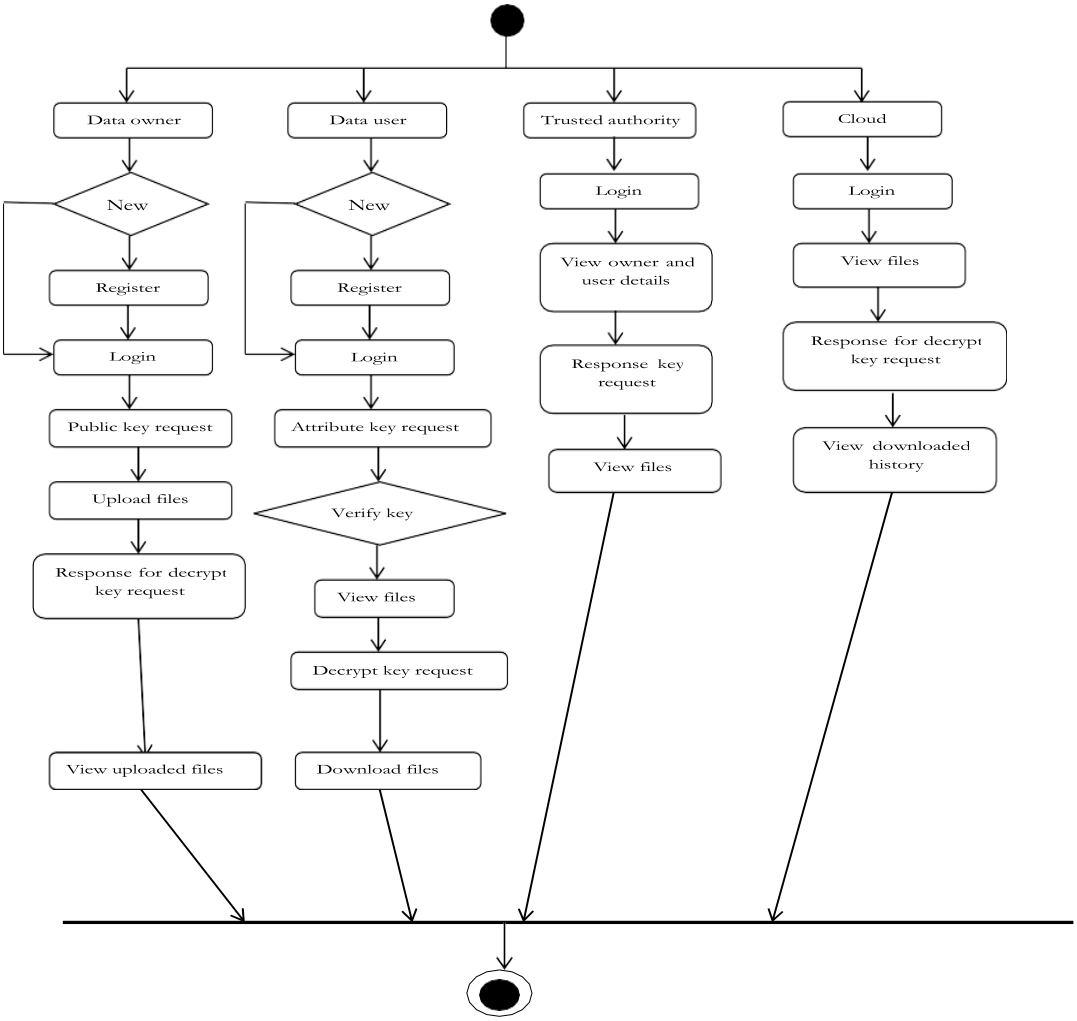

FIG 2. ACTIVITY DIAGRAM

4 Implementation

LDSS-CP-ABE algorithm is designed using the following functions.

Setup (*A, V*): Generate the master key *MK*, the public key *PK* based on attribute set *A* of the Data Owner and the version attribute *V*.

KeyGen (*A*u, *MK*): Generate attribute keys *SK*u for a data user *U* based on his attribute set *A*u and the master key *MK*.

Encryption (*K, PK, T*): Generate the ciphertext *CT* based on the symmetric key *K*, public key *PK* and access control tree *T* .

Decryption (*CT, T, SK*u): Decrypt the ciphertext *CT* using the access control tree *T* and the attribute keys *SK*u .

Function Setup() is called by the trusted third party (TA) to generate the master key and the public key. The master key is used to generate attribute keys and the public key is used to encrypt data files.

Function 1: Setup()
INPUT: The attribute set A , the version attribute V. OUTPUT: The master key MK, the public key PK.

1. Construct a p-order bilinear group G0 of generator g and a bilinear mapping e :$G_0 * G_0 = G_1$.
2. Randomly choose a,bε Z_p and calculate g^b ,$e(g, g)^a$.
3. For each attribute ai in A, randomly choose $t_i \varepsilon Z_p$, and calculate $X_i = g^{t_i}$
4. For V, randomly choose $t_v \varepsilon Z_p$, and calculate $X_v = g^{t_v}$.
5. Return the master key MK and the public key PK, Wherein MK={a,b}, PK={ G0 , g , g^b, $e(g,g)^a$, $\{X_i\}_{i=1}^k$,X_v}.

Function 2: KeyGen()
INPUT: The attribute set *A*u, the master key *MK*={*a,b*}.
OUTPUT: Attribute keys associated with *A*u.

1. Randomly choose a parameter, rε ZP , and calculate $SK_r = g^{(a+r)/b}$

ISBN: 978-81-936820-0-5
Proceedings DOI: 10.21467/proceedings.1

Series: AIJR Proceedings

2. For each attribute a_i in A_u, randomly choose , and $r_i \in Z_p$, and calculate $SK_r = \{g^{ri}, gr . X_i^{ri}\}_{i=1}^{j}$
3. For V, randomly choose $r_V \in Z_p$, and calculate $SK_u = \{ g^{rv}, g^r . X_v^{rv}\}$
4. Return $SK_u = \{SK_r, Sk_a, SK_u\}$

Function 3: Encryption()
INPUT: The symmetric key *K*, public key *PK*, access control tree *T* (including the left subtree *T*a, right subtree *T*v, and left subtree has *num* leaf nodes).
OUTPUT: The ciphertext *CT*.

1. Randomly choose $S \in Z_p$ as the secret of *T*, and calculate $CT_k = \{g^{bS}, K\, e(g,g)^{aS}\}$.
2. Get the value of the two children (namely S_a, S_v) of the root node according to the access control tree.
3. Calculate $CT_v = \{ g^{s_v} , g^r . X_v^{s_v} \}$
4. Return $CT = \{ CT_k, CT_a, CT_v \}$.

Function 4: Decryption()
INPUT: Ciphertext *CT*, the access control tree *T* (including the left subtree *T*a, right subtree *T*v, and left subtree has *num* leaf nodes), *SK*u (attribute keys of user *U*).
OUTPUT: The plaintext of *K*.

1. Randomly choose t, and get $SK_u' = \{ SK_r' = SK_r^{t}, SK_a, SK_v \}$.
2. For every leaf node z of T_a, calculate DecryptLeaf(CT_a, SK_u', z) = $e(g , g)^{qz(0)}$.
3. For the leaf node in right subtree, calculate DecryptLeaf(CT_v, SK_u', V) = $e(g , g)^{qv(0)}$.
4. Let $CT_k\text{-}1 = g^{bS}$, $CT_k\text{-}2 = K\, e(g, g)^{aS}$, *calculate K*

5 Conclusion and Future Work

In recent years, many studies on access control in cloud have been done that are based on attribute-based encryption algorithms (ABE). However, traditional ABE is not suitable for mobile cloud because it is computationally intensive and mobile devices only have limited resources. In this paper, we have proposed LDSS to address this issue. It introduces a novel LDSS CP-ABE algorithm to migrate major computation overhead from mobile devices onto proxy servers, thus it can solve the secure data sharing problem in mobile cloud. The experimental results show that LDSS can ensure data privacy in mobile cloud and reduce the overhead on users' side in mobile cloud. In future we have a proposal to design new approaches to ensure data integrity. To further tap the potential of mobile cloud, we will also study how to do ciphertext retrieval over existing data sharing schemes.

6 Acknowledgement

We gratefully acknowledge the help and cooperation offered by Dr. S Meenakshi Sundaram, Project Guide & Professor and Head, Department of Computer Science and Engineering, GSSSIETW, Mysuru. Also we thank our project coordinator Rummana Firdaus Asst. Professor Department of Computer Science and Engineering GSSSIETW, Mysuru for providing consistent help and support to carry out this project work. This project has been successfully implemented in the Research Centre of GSSSIETW, Mysuru.

References

[1]. Sahai, Amit, and Brent Waters. "Fuzzy identity-based encryption." In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457-473. Springer, Berlin, Heidelberg, 2005.

[2]. Goyal, Vipul, Omkant Pandey, Amit Sahai, and Brent Waters. "Attribute-based encryption for fine-grained access control of encrypted data." In *Proceedings of the 13th ACM conference on Computer and communications security*, pp. 89-98. Acm, 2006.

[3]. Ostrovsky, Rafail, Amit Sahai, and Brent Waters. "Attribute-based encryption with non-monotonic access structures." In *Proceedings of the 14th ACM conference on Computer and communications security*, pp.195-203. ACM, 2007.

[4]. Waters, Brent. "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization." In *International Workshop on Public Key Cryptography*, pp. 53-70. Springer, Berlin, Heidelberg, 2011.

[5]. Lewko, Allison, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, and Brent Waters. "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption." In Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 62-91. Springer, Berlin, Heidelberg, 2010.

[6]. Qihua Wang, Hongxia Jin. "Data leakage mitigation for discertionary access control in collaboration clouds". the 16th ACM Symposium on Access Control Models and Technologies (SACMAT), pp.103-122, Jun. 2011.

[7]. Wang W, Li Z, Owens R, et al. Secure and efficient access to outsourced data. in: Proceedings of the 2009 ACM workshop on Cloud computing security. Chicago, USA: ACM pp. 55-66, 2009.

[8]. Yu S., Wang C., Ren K., Lou W. Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing. INFOCOM 2010, pp. 534-542, 2010.

[9]. Kan Yang, Xiaohua Jia, Kui Ren: Attribute-based fine-grained access control with efficient revocation in cloud storage systems. ASIACCS 2013, pp. 523-528, 2013.

[10]. Crampton J, Martin K, Wild P. On key assignment for hierarchical access control. in: Computer Security Foundations Workshop. IEEE press, pp. 14-111, 2006.

[11]. Maheshwari U, Vingralek R, Shapiro W. How to build a trusted database system on untrusted storage. in: Proceedings of the 4th conference on Symposium on Operating System Design & Implementation-Volume 4. USENIX Association, pp. 10-12, 2000.

[12]. Kan Yang, Xiaohua Jia, Kui Ren, Ruitao Xie, Liusheng Huang: Enabling efficient access control with dynamic policy updating for big data in the cloud. INFOCOM 2014, pp.2013-2021, 2014.

[13]. Liang Xiaohui, Cao Zhenfu, Lin Huang, et al. Attribute based proxy re-encryption with delegating capabilities. in: Proceedings of the 4th International Symposium on Information, Computer and Communications Security. New York, NY, USA: ACM press, pp. 276-286, 2009.

[14]. D. Huang, X. Zhang, M. Kang, and J. Luo. Mobicloud: A secure mobile cloud framework for pervasive mobile computing and communication. in: Proceedings of 5th IEEE International Symposium on Service- Oriented System Engineering. Nanjing, China: IEEE, pp. 90-98, 2010.

[15]. Kan Yang, Xiaohua Jia, Kui Ren, Bo Zhang, Ruitao Xie: DAC-MACS: Effective Data Access Control for Multiauthority Cloud Storage Systems. IEEE Transactions on Information Forensics and Security, Vol. 8, No. 11, pp.1790-1801, 2013.