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. 3, Issue 7, July 2015

Compliant Data Sharing With Aggregate Key Development in Cloud Environment

Kranti M. Chaudhari, Prof. Vilas S. Gaikwad

PG Scholar, Department of Computer Engineering, JSPM NTC, Rajarshi Shahu School of Engineering and Research, Narhe, Pune, India

Assistant Professor, Department of Computer Engineering, JSPM NTC, Rajarshi Shahu School of Engineering and Research, Narhe, Pune, India

ABSTRACT: Now a day's most of the sensitive data are stored in the cloud servers. The central schmaltz of the cloud computing is security, which is enforced by the encryption. Examination over the data and data organizations are the main requirements of large data stowage systems. Data allotment is significant functionality in cloud environment. The data sharing must be secure, efficient and flexible with others in cloud storage. A new cryptographic system with aggregated key expansion is introduced for secure data sharing in cloud storage with the help of ElGamal encryption technique. Implementation of this system, need to design an efficient public-key encryption scheme which supports flexible delegation by keeping constant size of ciphertext, public key, master secret key and aggregate key in the sense that it allows proliferate the ciphertext classes. Experimental evaluation shows that proposed schemes is more flexible than hierarchical key assignment that can save energy and time of all key-holders which share a similar set of privileges.

KEYWORDS: data sharing, cryptographic algorithms, security,cloud storage, Aggregate key.

I. INTRODUCTION

Cloud computing represents today's most exciting computing paradigm shift in recent technologies. However, security and confidentiality are perceived as primary obstacles to its wide adoption. Authors sketch numerous critical security experiments and motivate further investigation of security results for a trustworthy public cloud environment. Accessing the resource sharing services and products are provided by cloud service providers. Although cloud based amenities offer many rewards, privacy and security of the complex data is a huge concern.

To mitigate the concerns, it is needed to subcontract complex data in encrypted form. Encrypted stowage defends the data against proscribed access, but it obscures some basic, yet significant functionality such as the examination on the data. To achieve pursuit over encrypted data without negotiating the privacy, considerable amount of searchable encryption systems have been proposed in the literature. Some refined protected multi-party reckoning based cryptographic techniques are available for resemblance tests, they are computationally intensive and do not scale for large data sources[1].

Cloud Computing moves the application and data to the cloud storage, where the management of the data and services may not be fully trustworthy. This unique attribute, however, poses many new security challenges which have not been well understood. This paper focuses on cloud data storage security, which has always been an important aspect of quality of service. 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 Many researchers have found interesting to work on key aggregate cryptosystem.

The idea of performing simple computations on encrypted messages was first introduced by Rivest, Adleman, and Dertouzous [2]. The original motivation for these homomorphisms was to allow for an encrypted database to be stored by a third party and to allow the owners and other authorized people to perform calculations with the data without

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. 3, Issue 7, July 2015

decrypting it. The next cryptosystem is the RSA which is most widely used public-key cryptosystem. It was developed in the year 1978 by Rivest, Shamir, and Adleman. It is one of the first homomorphic encryption schemes.

The rest of the paper is organized as follows: section 2 will introduce about related work done so far. Section 3 describe the problem definition and its description. Section 4 will give proposed work, Section 5 give result analysis process, section 6 will give conclusion, Section 7 References that we used in our proposed model.

II. RELATED WORK

In this section map a brief review of data sharing and encryption techniques has been used in cloud storage area for different purpose which is depicted as comparison of existing systems.
The researchers [3] develop a brand new cryptosystem for fine-grained sharing of encrypted information that we tend to decision Key-Policy Attribute-Based secret writing (KP-ABE). In AN ABE system, a user s keys and ciphertexts area unit labelled with sets of descriptive attributes and a selected key will decode a selected ciphertext providing there s a match between the attributes of the ciphertext and therefore the user s key. The cryptosystem allowed for cryptography once a minimum of k attributes overlapped between a ciphertext and a non-public key. Whereas this primitive was shown to be helpful for error-tolerant encryption with biometrics. Goyal et. al. proposed Attribute-Based Encryption System, using public key Encryption Scheme. This schemes allows each cipher text to be associated with an attribute, and the master-secret key holder can extract a secret key for a policy of these attributes. However the decryption key size is not constant.

Various cryptographic key assignment schemes (e.g., [4-7]) object to minimize the expense in storing and managing secret keys for general cryptographic use. Utilizing a tree structure, a key for a given branch can be used to derive the keys of its descendant nodes (but not the other way round). Just granting the parent key implicitly grants all the keys of its descendant nodes. Sandhu [8] proposed a method to generate a tree hierarchy of symmetric keys by using repeated evaluations of pseudorandom function/block-cipher on a _xed secret. The concept can be generalized from a tree to a graph. More advanced cryptographic key assignment schemes support access policy that can be modeled by an acyclic graph or a cyclic graph [9], [10].

Atallah et al. [11] proposed Key Assignment Schemes for a predefined hierarchy. This schemes support access policy that can be modeled by an acyclic graph or a cyclic graph. The scheme addresses the matter of access management and, a lot of specifically, the key management drawback in AN access hierarchy. Informally, the overall model is that there s a group of access categories ordered mistreatment partial order. a user AN agency obtains access (i.e., a key) to an explicit category also can acquire access to any or all descendant categories of her category through key derivation. However the decryption key is depends on hierarchy. The Scheme is handled by following properties:

1. In this scheme to stem the descendant's key of nodule, uses a hash function area unit. This derivation is over and done with its own key.
2. The complexity requires for storing the hierarchy is same to the general public info.
3. A key is constantly associated with the class and this is considered as the isolated information of class and this is considered as the private details of a class.
4. The apprises are handled nearby in the hierarchy

The researchers [12], attempt to alleviate the issue of constructing a safe and protected system of cloud storage which supports active and even capricious users and data province The abovementioned advantageous and sought-after attributes & properties is not offered by the prior system as it is based on certain constructions.
Problems and Constraints:
The given aspects inspect the restrained problems involved in the relations and dealings of these cryptographic primitives as well as add to the research of safe cloud storage systems:
1. Survey of Cryptographic Toolkits and a Generic System Design
2. Revocation in Group Signatures
3. Dynamic Broadcast Encryption
4. Linkage between Group Signatures and Broadcast Encryption.
5. The updates are handled locally in the hierarchy.

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. 3, Issue 7, July 2015

Guo et al. [13] proposed IBE which is a Asymmetric- key encryption. Here every user is personalized by the an identity set as public key. Whereas trusted third party holds master secret key. Encryption uses user identity and public parameter to encrypt the message. Receiver can decrypt it by its secret key.Unfortunately scheme increase the cost of transmitting and storing the cipher texts which is impractical in shared cloud storage.

Cheng-Kang Chu et al.[1] proposes a special public key encryption system , key-aggregate cryptosystem (KAC), In this, user can encrypt the message with public key and classes (identifier of cipher text). The systems outputs the constant size ciphertext. Limitation – Predefined bound number of maximum cipher text classes.

III. PROBLEM DEFINITIONS

To intention an effective public-key encryption scheme which supports flexible delegation in the intelligence that any subclass of the cipher texts is decrypt able by a constant-size decryption key which is independent of the maximum number of cipher text classes. In this section, we will introduce the notations and define the problem

A. Notations:

- Let DO = { o1 , o2, o3 .....odi }; Where DO is the set of data owner.
- EU = { e1, e2, e3 ..... ei }; Where EU is the set of end users
- K = {k1, k2,k3 ..... ki }; Where K is the set of keys.
- G = {DO, EU, PC, VM}; where G is the main set of all entities evolved in the application. Where, D = Data owner, U = End user, PC = Public cloud, VM= Virtual machine
- Identify the set of key generation entities and key compression
- KG = {u1k1, u2k2, u3k3.....}; where KG is the main set of keys generated for each file.
- Set of Deployment cloud - DC = {u1PC1} Where DC is set of deployment cloud. The set is individual element of public cloud will be only one.
-

B. Problem Description:

Data sharing is a main functionality in cloud storage. The data sharing is now considered as a crucial requirement. But this method is susceptible to many security attacks [14,15]. Thus there is need for secured data sharing methods. For example, bloggers can let their groups view a subset of their private pictures; an enterprise may fund her employee's access to a portion of sensitive data. The challenging problem is how to effectually share encrypted data [1]. Of course users can copy the encrypted data from the storage space, decrypt them, then send them to others for sharing, but it drops the value of cloud storage. Users should be able to give the access rights of the sharing data to others so that they can access these data from the server direct. Though, finding an efficient and secure way to share incomplete data in cloud storage is not trivial. Below we will take Dropbox as an example for design shown in Figure 1. Assume that Alice places all her private photos on Dropbox, and she does not want to representation her photo to everyone. Due to a variety of data leakage possibility Alice cannot feel reassured by just trusting on the privacy protection mechanisms provided by Drop box, thus she encrypts all the photos using her own keys before uploading. Single day, Alice's friend, Bob, requests her to share the photos taken over all these years which Bob looked in. Alice can then use the share function of Dropbox, but the problem now is how to give the decryption rights for these photos to Bob. A possible option Alice can choose is to securely send Bob the secret keys complicated. Naturally, there are two extreme ways for her under the traditional encryption example:

- Alice encrypts each and every one file with a single encryption key and gives Bob the corresponding secret key directly.
- Alice encrypts files with different keys and sends Bob the resultant secret keys.

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. 3, Issue 7, July 2015

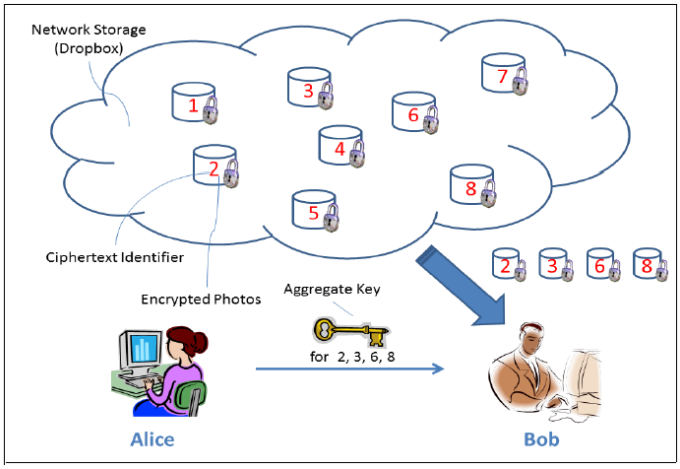

Figure 1. Alice shares files with identifiers 2, 3, 6 and 8 with Bob (by sending him a single aggregate key)

Clearly, the first method is inadequate since all un-chosen data may be also leak to Bob. For the second technique, there are practical concerns on efficiency. The amount of such keys is as many as the number of the shared photo, say, a thousand. Transfer these secret keys integrally requires a safe channel, and storing these keys require rather expensive secure storage [16]. The costs and complications involved generally increase with the number of the decryption keys to be shared. In short, it is very heavyweight and costly to do that. Therefore, the best solution for the above problem is that Alice encrypts files with different public-keys, but only sends Bob a single (constant-size) decryption key. Seeing as the decryption key should be sent through a secure channel and set aside secret, small key size is always desirable. For example, we cannot guess large storage for decryption keys in the resource-constraint devices like smart phones, smart cards or wireless sensor nodes.

IV. PROPOSED METHODOLOGY

In this section we first describe the proposed system design. The methodology used for developing the proposed system is also discussed in this section. Later we describe our main contribution of the cryptosystem.

A. System Architecture:

Cloud Systems can be used to enable data sharing capabilities and this is an abundant benefit to the user. As Cloud computing is a modern era computing technique that has a grater future and bringing a lot of benefit to the information technology. It can be defined as a set of resources or services that are offered to the users of internet. These services and products are provided by cloud service providers. These services and products are provided by cloud service providers. Through cloud computing the service providers deliver almost everything as a service over internet on user demand. The user demands may include operating system, network hardware, storage, resources, software, etc. are shown in figure 2.

The proposed methodology can be implemented in three phases

1. Setup Phase
2. Encrypt Phase
3. Key Gen Phase

1. Setup Phase: The setup algorithm takes no input other than the implicit security parameter. It outputs the public parameters PK and a master key MK.

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. 3, Issue 7, July 2015

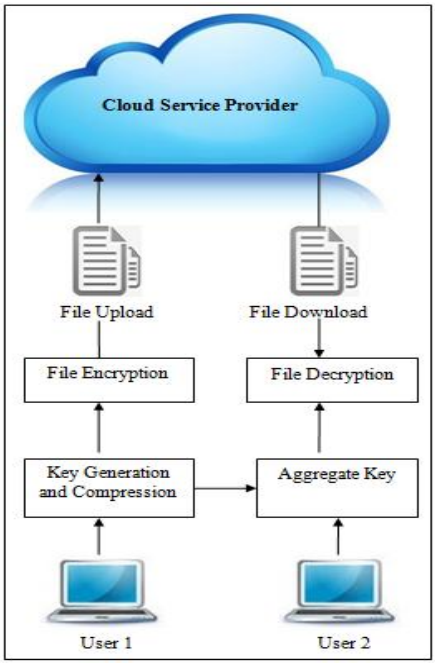

Figure 2: System Architecture

2. Encrypt Phase: Encrypt(PK,M, A). The encryption algorithm takes as input the public parameters PK, a message M, and an access structure A over the universe of attributes. The algorithm will encrypt M and produce a ciphertext CT such that only a user that possesses a set of attributes that satisfies the access structure will be able to decrypt the message. We will assume that the ciphertext implicitly contains A.
3. Key Gen Phase: Key Generation(MK,S). The key generation algorithm takes as input the master key MK and a set of attributes S that describe the key. It outputs a private key SK

B . ELGAMAL Encryption Algorithm:
The ElGamal encryption system is an asymmetric key encryption algorithm for public-key cryptography[17]. The ElGamal Algorithm provides an alternative to the RSA for public key encryption.
1. Security of the RSA depends on the (presumed) difficulty of factoring large integers.
2. Security of the ElGamal algorithm depends on the (presumed) difficulty of computing discrete logs in a large prime modulus.
ElGamal has the advantage, the same plaintext gives a different ciphertext (with near certainty) each time it is encrypted. ElGamal uses a smaller key length when compared to RSA. The following algorithm shows the general overview of ElGamal cryptosystem.
Step 1 : Alice chooses

- A large prime (say 200 to 300 digits),
- A primitive element α_A modulo ρ_A ,
- A (possibly random) integer d_A with $2 \leq d_A \leq \rho_A - 2$.

Step 2 : Alice computes

- $\beta_A \equiv \alpha_A^{d_A} (\mathrm{mod}\ \rho_A)$
- Alice's public key is (ρ_A, α_A , β_A)Her private key is d_A.

Step 3 : Bob encrypts a short message M ($M < \rho_A$) and sends it to Alice like this:

- Bob chooses a random integer k (which he keeps secret).
- Bob computes $r \equiv \alpha_A^k \pmod{\rho_A}$ and $t \equiv \beta_A^k M \pmod{\rho_A}$, and then discards K .
- Bob sends his encrypted message (r, t) to Alice

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. 3, Issue 7, July 2015

Step 4 : When Alice receives the encrypted message (r, t), she decrypts (using her private key d_A) by computing tr^{-d_A}.

- $tr^{-d_A} \equiv \beta_A^k M(\alpha_A^k)^{-d_A}$. (mod ρ_A)
- $\equiv (\alpha_A^{d_A})^k M (\alpha_A^k)^{-d_A}$ (mod ρ_A)
- $\equiv M$ (mod ρ_A)

Step 5 : Even if Eve intercepts the ciphertext (r, t), she cannot perform the calculation above because she doesn't know d_A

- $\beta_A \equiv \alpha_A^{d_A} (\text{mod } \rho_A)$ so $d_A \equiv L_{\alpha A}(\beta_A)$
- Even can find d_A if she can compute a discrete log in the large prime modulus ρ_A , presumably a computation that is too difficult to be practical.

Step 6 : If M is a longer message, so it is divided into blocks, he should choose a different k for each block. Say he encrypts two messages (or blocks) M1 and M2, using the same k, producing cipher texts

- $(r_1, t_1) = (\alpha_A^k, \beta_A^k M_1)$, $(r_2, t_2) = (\alpha_A^k, \beta_A^k M_2)$
- Then $t_2 t_1^{-1} = M_2 M_1^{-1} (mod\ \rho)$, $M_2 = t_2 t_1^{-1} M_1 (mod\ \rho)$
- If Eve intercepts both cipher text messages and discovers one plaintext message M1, she can compute the other plaintext message M2.

C. Flow-chart Of The System:

The proposed system has been divided into three modules: Module one will focus on creation of UI for entire application. This module will create the platform for data owner to browse the data and store it on cloud. Module two performs constant size key generation and encryption using Elgamal will be performed. This will be done at data owner side and files will be stored on cloud. Module three performs decryption. The user will get aggregated keys from other user who will send the files over cloud.

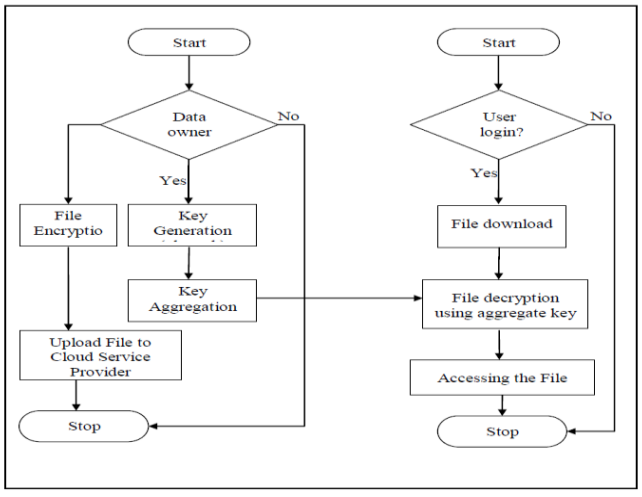

Figure 3. Flowchart of proposed system

V. RESULT AND ANALYSIS

To make the best out of our extended scheme (i.e., to make the key size as small as possible), suggest that the cipher text classes for different purposes should be corresponded to different public-keys. This is reasonable in practice and does not contradict our criticism on hierarchical methods that an efficient assignment of hierarchy requires a priori knowledge on what to be shared.The proposed system is compared with KAC, results are shown in Table 2. Figure 5(a) shows the comparison of execution time between KAC [1] and proposed system. KAC system requires execution time

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. 3, Issue 7, July 2015

33.3 millisecond and 57.35 millisecond to upload and encrypt file of 150 KB respectively. In our experiment execution time to upload a 150 KB size file is 16 milliseconds and Encrypt takes 39 milliseconds.
Throughput of proposed system is as shown in figure 5(b) estimated with Throughput Calculator Tool[18].

TABLE 2: COMPARISON OF EXECUTION TIME

Approach	Execution time in ms
KAC	90.65
Proposed System	55

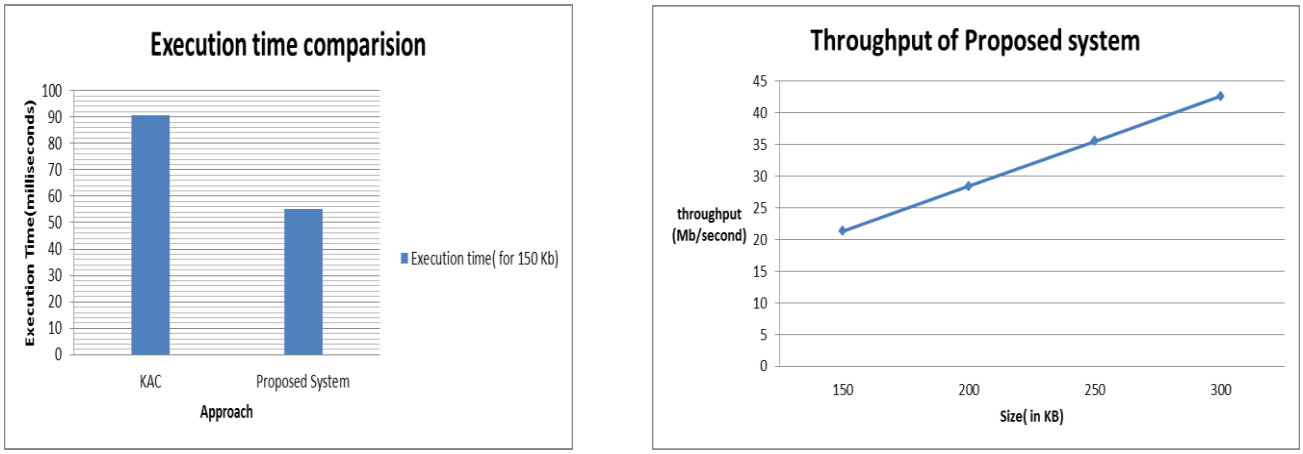

Figure 5 (a)Execution Time comparisons between KAC and proposed system. (b) Throughput with elapsed time as 55 milliseconds.

VI. CONCLUSION

The rapid growth in cloud computing has also increased the security concerns related to cloud computing environment. Usually users' needs to face security challenges during the Data sharing and data privacy of cloud aided computation The public key cryptosystem enhances the user confidentiality and privacy of data in cloud storage. But still it needs support of secret keys for variety of text classes and generation of keys. The proposed system implemented with the Elgamal algorithm by extending the public key, create aggregate key which is independent of maximum number of cipher text classes. Comparing to most of the existing schemes, the proposed scheme is more secured and flexible, better suiting users efficient and an interesting direction towards flexible key delegation.

REFERENCES

1. Cheng-Kang Chu, Sherman S. M. Chow, Wen-Guey Tzeng, Jianying Zhou, and Robert H. Deng "Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage" IEEE Transactions on Parallel and Distributed Systems. Volume: 25, Issue: 2. Year :2014.
2. Rivest, Adleman, and Dertouzous Foudations of Secure Computations,Academia press 1978.
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 (CCS '06). ACM, 2006, pp. 89–98.
4. S. G. Akl and P. D. Taylor, Cryptographic Solution to a Problem of Access Control in a Hierarchy, ACM Transactions on Computer Systems (TOCS), vol. 1, no. 3, pp.239248, 1983.
5. G. C. Chick and S. E. Tavares, Flexible Access Control with Master Keys, in Proceedings of Advances in Cryptology CRYPTO 89, ser. LNCS, vol. 435. Springer,1989, pp. 316322.
6. W.-G. Tzeng, A Time-Bound Cryptographic Key Assignment Scheme for Access Control in a Hierarchy, IEEE Transactions on Knowledge and Data Engineering (TKDE), vol. 14, no. 1, pp. 182188, 2002.
7. G. Ateniese, A. D. Santis, A. L. Ferrara, and B. Masucci, Provably-Secure Time-Bound Hierarchical Key Assignment Schemes, J. Cryptology, vol. 25, no. 2, pp.243270, 2012.
8. R. S. Sandhu, Cryptographic Implementation of a Tree Hierarchy for Access Control, Information Processing Letters, vol. 27, no. 2, pp. 9598, 1988.
9. Y. Sun and K. J. R. Liu, Scalable Hierarchical Access Control in Secure Group Communications, in Proceedings of the 23th IEEE International Conference on Computer Communications (INFOCOM04). IEEE, 2004.

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. 3, Issue 7, July 2015

10. Q. Zhang and Y. Wang, A Centralized Key Management Scheme for Hierarchical Access Control, in Proceedings of IEEE Global Telecommunications Conference(GLOBECOM 04). IEEE, 2004, pp. 20672071.
11. M. J. Atallah, M. Blanton, N. Fazio, and K. B. Frikken, "Dynamicand Efficient Key Management for Access Hierarchies," ACM Transactions on Information and System Security (TISSEC), vol. 12, no. 3, 2009.
12. D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and Verifiably Encrypted Signatures from Bilinear Maps," in Proceedings of Advances in Cryptology - EUROCRYPT '03, ser. LNCS, vol. 2656. Springer, 2003, pp. 416–432.
13. F. Guo, Y. Mu, Z. Chen, and L. Xu, "Multi-Identity Single-Key Decryption without Random Oracles," in Proceedings of Information Security and Cryptology (Inscrypt '07), ser. LNCS, vol. 4990. Springer, 2007, pp. 384–398.
14. S. S. M. Chow, Y. J. He, L. C. K. Hui, and S.-M. Yiu, "SPICE -Simple Privacy-Preserving Identity-Management for Cloud Environment,"in Applied Cryptography and Network Security – ACNS 2012, ser. LNCS, vol. 7341. Springer, 2012, pp. 526–543.
15. Bender D (2012) Privacy and security issues in cloud computing. Comput Internet Lawyer 1–15.
16. Judith H, Robin B, Marcia K, Fern H (2009) Cloud computing for dummies. For Dummies.
17. Scale ME (2009) Cloud computing and collaboration. Library Hi Tech News, pp 10–13.
18. http://www.itadmintools.com/2012.