International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015
Available at http://internationaljournalofresearch.org

A Novel Key Aggregating on multiple Files Sharing on Cloud Data Storage

Martha Madhuri[1]& N.Shiva Kumar[2]
[1]M-Tech Dept. of CSE Megha Institute of Engineering & Technology for Women
[2]Assistant ProfessorDept. of CSE Megha Institute of Engineering & Technology for Women

Abstract
The main purport is to apportion data securely among users. The cloud accommodation provider and users Authentication is compulsory to ascertain no loss or leak of users data in cloud.Cloud computing technology is widely used so that the data can be outsourced on cloud[2] can accessed facilely. Different users can apportion that data through different virtual machines which present on single physical machine. But the thing is utilizer don't have control over the outsourced data. The difference is one can amass a set of secret keys and make them as minuscule size as a single key with holding the same ability of all the keys that are composed in a group as aggregate[4][5] key.Privacy preserving in cloud is consequential[8]. Cryptography avails the data owner to apportion the data to the requested utilizer in safe way. For that the data owner encrypts the data and uploads on server. The encryption and decryption keys may be different or same for different set of data. For decrypting the required data only the set of decryption keys are shared. Here a public key cryptosystems which engenders a cipher text which is of constant size.

Keywords: Cloud Provider; Authentication; Privacy in Cloud; Cryptography; Encrypt; Cypher text; Aggregate Key

1. Introduction

Nowadays, many immensely colossal scale and diminutive scale organizations outsource their astronomically immense-scale data storage to the cloud for preserving the cost in maintaining their storage. With cloud storage accommodation, the members of an organization can apportion data with other members facilely by uploading their data to the cloud. Examples of organizations which may benefit from this cloud storage and sharing accommodation are numerous, such as international enterprises with many employees around the world, collaborative web application providers with a sizably voluminous utilizer base, or institutions dealing with sizably voluminous data, healthcare researchers, patients, etc. While the economic benefits brought by outsourcing data can be captivating, security is one of the most consequential factors that obstruct its wide development.

Cryptography is the method of storing and transmitting data in a form that only those intended for it can read and process the required data. It is technique of forfending information by encrypting the data it into an unreadable format utilizing some encryption algorithm[19]. Cryptography is an efficacious way of bulwarking sensitive information that is to be stored on media or transmitted through network communication paths. The main goal of cryptography is that to obnubilate information from unauthorized individuals like intruders or hackers[7]. Hackers now a day can hack most of the cryptography algorithms and the information can be revealed if the assailant has enough time and resources to hack the data. So a more authentic goal of cryptography is to decrypting the data to be arduous. Considering data privacy, rely on the server to enforce the access

control after authentication, if there is any unexpected privilege escalation will expose all data which is sensitive.

A cryptographic[11] solution, with proven security relied on number-theoretic assumptions is more acceptable, whenever the user is not perfectly happy with trusting the security of the VM or the honesty of the staff. These users are motivated to encrypt their data with their own keys before uploading them to the server.Cloud is a market-oriented distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements (SLAs) established through negotiation between the service provider and consumers. In cloud computing, users can outsource their computation and storage to servers (also called clouds) using Internet

User privacy is also required so that the cloud or other users do not know the identity of the user. Thus it is complex systems which possess highly securable processes. Consider that User1 sends all her private photos or any other important data's on Drop box or any other cloud application, and she does not want to make visible her photos to everyone. Due to various data exposing possibility User1 cannot feel relieved by just depends on the security protection mechanisms provided by Drop box, so she encrypts all the photos using her own keys before uploading for security. One day, User1's friend, User2, asks her to share the photos taken over all these years. User1 can then use the share function of Drop box, but the problem now is how to delegate the decryption rights for these photos owned to User2. A possible option User1 can choose is to securely send User2 the secret keys involved for authenticating the data's.Naturally, there are two extreme ways for her under the traditional encryption paradigm:

•User1 encrypts files with a single encryption key and gives User2 the corresponding secret key directly.

•User1 encrypts all files with distinct keys and sends User2 the corresponding secret keys.

Clearly, the first method is not much secure because all unchoosen data may be also leaked to User2. For the second method, there are some practical issues on symmetric encryption, when User1 wants the data to be came from a third party, she has to give the encryptor her secret key; Clearly, this is not always desirable. By contrast, the encryption key and decryption key differs in public-key encryption. The use of public-keyThe above two methods didn't provide security completely. The key handling process looks very simple but not promising. Thus proposed system will solve these two problems by providing a proper security structure. These two problems are very difficult and should be sorted out. encryption gives more flexibility for applications.

In latest cryptography area, a fundamental problem here often study is about leveraging the secrecy of a small piece of knowledge into the ability to perform cryptographic functions[13] (e.g. encryption, authentication) several times. In this paper how to create a decryption key more powerful in the sense that it allows decryption of multiple cipher texts, without increasing its size. Specifically, system problem statement is – " To design an efficient public-key encryption scheme which supports flexible delegation[12] in the sense that any subset of the cipher texts (produced by the encryption scheme) is decryptable by a constant-si4EWze decryption key (generated by the owner of the master-secret key)."

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015
Available at http://internationaljournalofresearch.org

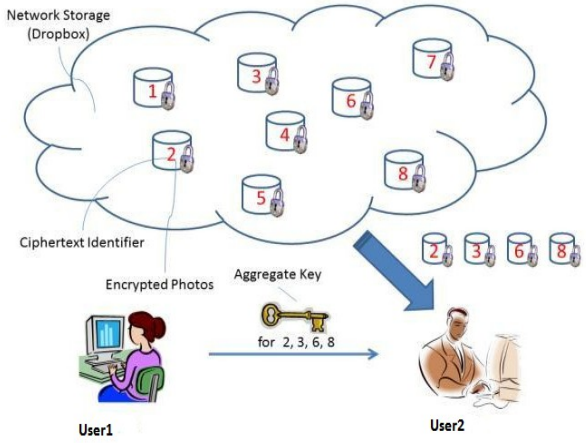

Fig 1: User1 share files to User2 by using single aggregate key.

Here now solve this problem by introducing a different type of public-key encryption which here call key-aggregate[1] cryptosystem (KAC). In KAC, users encrypt a message not only under a public-key, but also under an identifier of cipher text called class. That means the cipher texts are further categorized into various classes. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for various different classes.

Importantly, the extracted key have can be an aggregate key which is as compact as a secret key for a single class, but aggregates the power of many such keys, i.e., the decryption power for any subset of cipher text classes. With system solution, User1 can simply send User2 a single aggregate key via a secure e-mail. Now User2 can download the encrypted photos from User1's Drop box space and then use the same aggregate key to de-crypt these encrypted photos.The sizes of cipher text, public-key, master -secret key and aggregate key in system KAC schemes are all of constant size.

The public sys-tem parameter has size linear in the number of cipher text classes, but only a small part of it is needed each time and it can be fetched on demand from large cloud storage. Previous results may achieve a similar property featuring a constant-size decryption key, but the classes need to conform to some pre-defined hierarchical relationship. System work is flexible in the sense that this constraint is eliminated, that is, no special relation is required between the classes. The detail and other related works can be found in Section 3. Here propose several concrete KAC schemes with different security levels and extensions in this article.

2. Related Work

2.1 Existing System work:

Surmise that User1 puts all her private files on (cloud), and she does not optate to expose her files to everyone. Due to sundry data leakage possibility User1 cannot feel assuaged by just relying on the privacy aegis mechanisms provided by, so she encrypts all the files utilizing her own keys afore uploading. One day, User1's friend, User2, asks her to apportion the files surmounted all these years which User2 appeared in. User1 can then utilize the portion function of, but the quandary now is how to delegate the decryption rights for these files to User2. A possible option User1 can optate is to securely send User2 the secret keys involved. Naturally, there are two extreme ways for her under the traditional encryption paradigm:

- User1 encrypts all files with a single encryption key and gives User2 the corresponding secret key directly.
- User1 encrypts files with distinct keys and sends User2 the corresponding secret keys.

Conspicuously, the first method is inadequate since all un-culled data may be withal leaked to User2. For the second method; there are practical concerns on efficiency. The number of such keys is as many as the number of the shared files, verbally express, a thousand. Transferring these secret keys inherently requires a secure channel, and storing these keys requires rather expensive secure storage. The costs and involutions involved generally increase with the number of the decryption keys

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015
Available at http://internationaljournalofresearch.org

to be shared. In short, it is very heftily ponderous and costly to do that.

2.2 Proposed System work:

Introduce a special type of public-key encryption which call key-aggregate cryptosystem (KAC). In KAC, users encrypt a message not only under a public-key, but also under an identifier of cipher text called class. That means the cipher texts are further categorized into different classes. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for different classes. More importantly, the extracted key have can be an aggregate key which is as compact as a secret key for a single class, but aggregates the power of many such keys, i.e., the decryption power for any subset of cipher text classes. With a solution,USER1 can simply send USER 2 a single aggregate key via a secure e-mail. USER2 can download the encrypted files from USER 1 Drop box space and then use this aggregate key to decrypt these encrypted files. USER1 encrypts all files with a single encryption key and gives USER 2 the corresponding secret key directly. USER1 encrypts files with distinct keys and sends USER2 the corresponding secret keys.

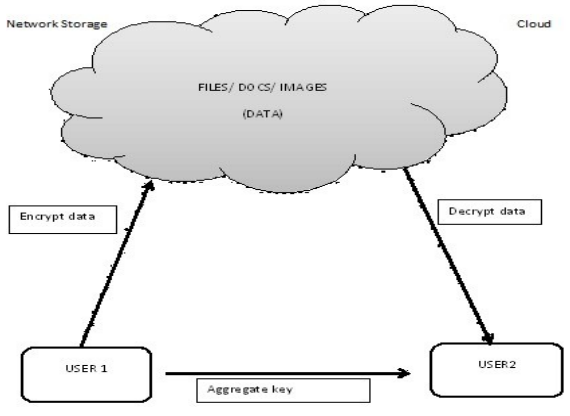

Fig 2: Proposed System Architecture Diagram.

To design an efficient public-key encryption scheme which supports flexible delegation[12] in the sense that any subset of the ciphertext (produced by the encryption scheme) is decryptable[10]by a constant-size decryption key (generated by the owner of the master-secret key).Solve this problem by introducing a special type of public-key encryption which call key-aggregate cryptosystem (KAC). In KAC, users encrypt a message not only under a public-key, but also under an identifier of ciphertext called class. That means the ciphertext are further categorized into different classes.

The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for different classes. More importantly, the extracted key have can be an aggregate key which is as compact as a secret key for a single class,but aggregate the power of many such keys, i.e., the decryption power for any subset of ciphertextclasses.With solution, USER1 can simply send USER2 a single aggregate key via a secure e-mail. USER2 can download the encrypted data from USER1 space and then use this aggregate key to decrypt these encrypted photos.

The sizes of ciphertext, public-key, master-secret key and aggregate key in our KAC schemes are all of constant size. The public system parameter has size linear in the number of ciphertext classes, but only a small part of it is needed each time and it can be fetched on demand from large (but non-confidential) cloud storage. Previous results may achieve a similar property featuring a constant-size decryption key, but the classes need to conform to some pre-defined hierarchical relationship. Work is flexible in the sense that this constraint is eliminated, that is, no special relation is required between the classes.

2.3 Cryptographic Keys for a Predefined Hierarchy:

Here start by discussing the most relevant study in the literature of cryptography/security. Cryptographic[13]key assignment schemes aim to minimize the expense in storing and

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015
Available at http://internationaljournalofresearch.org

managing secret keys for general cryptographic use[18]. Data management is the important aspects of cloud. So this area must be checked clearly. Securing huge amount of data is very much risky. So system proposed scheme is made to solve all these problems. The below tabular column depicts system scheme efficiency with other various schemes. Utilizing a tree structure, a key for a given branch can be used to derive the keys of its decreasing nodes. Just giving the parent key implicitly grants all the keys of its descendant nodes. Sandhu[14][15] proposed a method to generate a tree hierarchy of symmetric keys by using repeated evaluations of pseudorandom function/ block-cipher on a fixed secret.

Take the tree structure as an example. User1 can first classify the cipher text classes according to their subjects like Figure 3. Each node in the tree represents a secret key, while the leaf nodes represent the keys for individual cipher text classes. Filled circles represent the keys for the classes to be delegated and circles circumvented by dotted lines represent the keys to be granted. Note that every key of the non-leaf node can derive the keys of its descendant nodes.

	Decryption key size	Ciphertext size	Encryption type
Key assignment schemes a predefined hierarchy (e.g., [7])	Most Likely Non-Constant(Depends on the Hierarchy)	Constant	Symmetric or Public Key
Symmetric-Key encryption with Compact Key	Constant	Constant	Symmetric-Key
IBE with Compact Key	Constant	Non-Constant	Public Key
Attribute Based Encryption	Non-constant	Constant	Public key
KAC	Constant	Constant	Public key

Table 1:Comparison between Proposed basic KAC schema and other related schema.

2.4 Proposed Algorithm:

(a) Elliptic curve cryptography algorithm:

Elliptic curves have a rich and beautiful history, having been studied by mathematicians for over a hundred years. They have been used to solve a diverse range of problems. One example is the congruent number problem that asks for a classification of the positive integers occurring as the area of some right-angled triangle, the lengths of whose sides are rational numbers. Another example is proving Fermat's Last Theorem which states that the equation xn + yn = zn has no nonzero integer solutions for x, y and z when the integer n is greater than 2.

In 1985, Neal Koblitz and Victor Miller independently proposed using elliptic curves to design public-key cryptographic systems. Since then an abundance of research has been published on the security and efficient implementation of elliptic curve cryptography. In the late 1990's, elliptic curve systems started receiving commercial acceptance when accredited standards organizations specified elliptic curve protocols, and private companies included these protocols in their security products.

Elliptic curve key generation:

Let E be an elliptic curve defined over a finite field Fp. Let P be a point in E(Fp), and suppose that P has prime order n. Then the cyclic subgroup of E(Fp) generated by P is

(P) = {∞, P,2P,3P,...,(n −1)P}.

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015
Available at http://internationaljournalofresearch.org

The prime p, the equation of the elliptic curve E, and the point P and its order n, are the public domain parameters. A private key is an integer d that is selected uniformly at random from the interval [1,n − 1], and the corresponding public key is Q = d P.

The problem of determining d given the domain parameters and Q is the elliptic curve discrete logarithm problem (ECDLP).

Algorithm Elliptic curve key pair generation

INPUT: Elliptic curve domain parameters (p, E, P,n).

OUTPUT: Public key Q and private key d.

1. Select d ∈R [1,n −1].
2. Compute Q = d P.
3. Return (Q,d).

Elliptic curve encryption scheme:

Here present the encryption and decryption procedures for the elliptic curve analogue of the basic ElGamal encryption scheme as Algorithms 1.13 and 1.14, respectively. A plaintext m is first represented as a point M, and then encrypted by adding it to kQ where k is a randomly selected integer, and Q is the intended recipient's public key. The sender transmits the points C1 = k P and C2 = M +kQ to the recipient who uses her private key d to compute dC1 = d(k P) = k(d P) = kQ,

and thereafter recovers M = C2 − kQ. An eavesdropper who wishes to recover M needs to compute kQ. This task of computing kQ from the domain parameters, Q, and C1 = k P, is the elliptic curve analogue of the Diffie-Hellman problem[9].

Algorithm Basic ElGamal elliptic curve encryption

INPUT: Elliptic curve domain parameters (p, E, P,n), public key Q, plaintext m.

OUTPUT: Ciphertext (C1,C2).

1. Represent the message m as a point M in E(Fp).
2. Select k ∈R [1,n −1].
3. Compute C1 = k P.
4. Compute C2 = M +kQ.
5. Return (C1,C2)

Complexity of an ECC Algorithm:

Let A be an algorithm whose input has bit-length n A is a polynomial-time algorithm if its running time is O(n c) for some constant c>0, such as n10

A is a subexponential-time algorithm if its running time is O(e o(n)), such as e n1/3 .

A is an exponential-time algorithm if its running time is O(c n) or O(nf(n)) for c>1, such as 1.1 n and n n 2 .

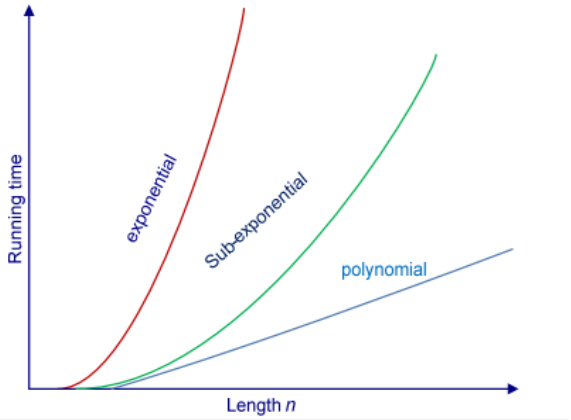

Fig 4: Complexity of ECC Algorithm.

The mathematic background of ECC is more complex than other cryptographic systems Geometry, abstract algebra, number theory ECC provides greater security and more efficient performance than the first generation public key techniques (RSA and Diffie-Hellman) Mobile systems Systems required high security level (such as 256 bit AES) The next step is to apply the ECDH principle to the group key management protocol[17]. Unless the explicit statement of sources, the materials used in this tutorial are from Hankerson'sbook[9] and www.certicom.com.

(b) Aggregate Secret Key Generation:

The data owner produces public/master-secret key pair in this phase. Aggregate key[1] generation phase is divided in three steps. Messages can be encrypted using Encrypt( ) function by anyone who also decides which ciphertext class is associated with the plain text message. The data owner can use master-secret

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015
Available at http://internationaljournalofresearch.org

to generate an aggregate decryption key for ciphertext classes using function Extract( ).

- **Setup and KeyGen:** Same as the basic construction.
- **Extend(pkl; mskl):** Execute KeyGen() to get (vl+1; l+1) 2 G _ Zp, output the extended public and master-secret keys as pkl+1 = (pkl; vl+1); mskl+1 = (mskl; l+1)
- **Encrypt(pkl; (a; b);m):** Let pkl = fv1; _ _ _; vlg. For an index (a; b); 1 _ a _ l; 1 _ b _ n, pick t 2R Zp, output the ciphertext as C = hgt; (vagb)t;m _ ^e(g1; gn)ti
- **Extract(mskl; Sl):** Let mskl = f1; 2; _ _ _ ; lg. For a set Sl of indices (i; j); 1 _ i _ l; 1 _ j _ n, getgn+1_j = g_n+1_j from param, output: KSl = (Y(1;j)2Sl g1 n+1_j ; Y (2;j)2Sl g2 n+1_j ; _ _ _ ; Y (l;j)2Sl gl n+1_j)
- **Decrypt(KSl ; Sl; (a; b); C):** If (a; b) =2 Sl, output ?. Oth-erwise, let KSl = (d1; _ _ _ ; dl) and C = hc1; c2; c3i. Output the message: m = c3 _ ^e(da _ Q (a;j)2Sl;j6=b gn+1_j+b; c1) ^e(Q (a;j)2Sl gn+1_j ; c2) Just like the basic construction, the decryption can be done more efficiently with the knowledge of i's. Cor-rectness is not much more difficult to see: c3 _ ^e(da _ Q (a;j)2Sl;j6=b gn+1_j+b; c1) = ^e(Q (a;j)2Sl gn+1_j ; c2) = c3 _ ^e(Q (a;j)2Sl ga n+1_j _ Q (a;j)2Sl;j6=b gn+1_j+b; gt) =^e(Q (a;j)2Sl gn+1_j ; (vagb)t) =c3 _ ^e(Q (a;j)2Sl;j6=b gn+1_j+b; gt)=^e(Q (a;j)2Sl gn+1_j ; gt b) = m _ ^e(g1; gn)t=^e(gn+1; gt) = m: Herecan also prove the semantic security of this extend-ed scheme. The proof is very similar to that for the ba-sic scheme and therefore is omitted. The public-key of system CCA construction to be presented below can also be extended using the same Extend algorithm.

3. System Implementation

a) **Key Gen:** executed by the data owner to randomly generate a public/master-secret key pair(pk; msk).

b) **Encrypt (pk; i;m):** executed by anyone who wants to encrypt data. On input a public-key pk, an index i denoting the ciphertext class, and a message m, it outputs a ciphertext C.

c) **Extract (msk; S):** executed by the data owner for delegating the decrypting power for a certain set of ciphertext classes to a delegatee. On input the mastersecret key msk and a set S of indices corresponding to different classes, it outputs the aggregate key for set S denoted by KS.

d) **Decrypt (KS; S; i; C):** executed by a delegatee who received an aggregate key KS generated by Extract. On input KS, the set S, an index i denoting the[20]

4. System Design:

The System Design describes the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces.

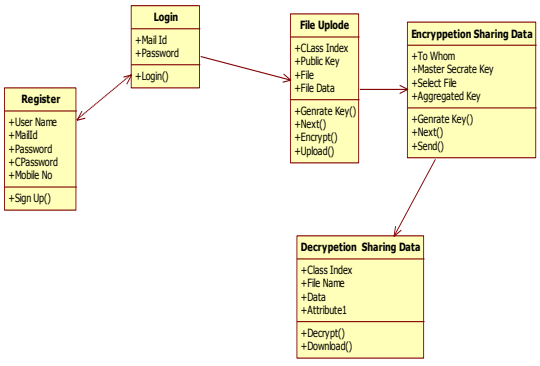

Fig 5: System Class model.

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 09, September 2015
Available at http://internationaljournalofresearch.org

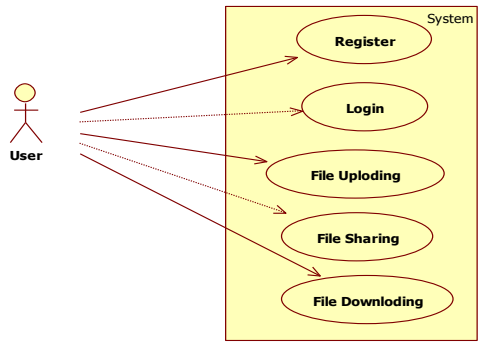

Fig 6: System Use case Diagram.

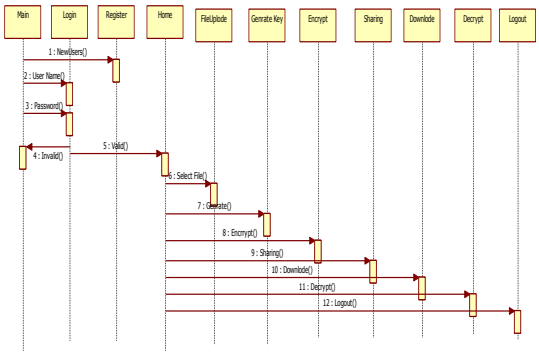

Fig 7: System Sequence Diagram.

5. Conclusion

In this paper, literature survey on key aggregate cryptosystem was auxiliary to grasp the technique and the way the techniques area unit developed to apportion cognizance among users in cloud. To apportion erudition flexibly is paramount factor in cloud computing. Users relish to transfer their cognizance on cloud and among consummately different users. Outsourcing of erudition to server could lead to leak the personal erudition of utilizer to everybody. Coding could be a one solution that provides to apportion culled erudition with desired candidate. Sharing of cryptography keys in secure method plays vital role. Public -key cryptosystems provides delegation of secret keys for plenarily different ciphertext categories in cloud storage. The delegate gets firmly associate amalgamation key of constant size. It'sneeded to keep enough range of cipher texts categories as they increment expeditious and the ciphertext categories area unitfinite that's the circumscription.

6. References

[1]Cheng-Kang Chu, Sherman S.M. Chow, Wen-GueyTzeng, Jianying Zhou, and Robert H," Key-Aggregate Cryptosystem for Scalable DataSharing in Cloud Storage" ,IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 25, NO. 2, FEBRUARY 2014.

[2]Cloud Storage "IEEE TRANSACTIONS ON COMPUTERS, VOL. 62, NO. 2, FEBRUARY 2013.

[3]Reza Curtmola and Osama Khan Randal Burns, "Robust Remote Data Checking" ,Proceedings of the 4th ACM international workshop on Storagesecurity and survivability PAGES63-68 ACM 978-1-60558-299-3.

[4]D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and Verifiably Encrypted Signatures from Bilinear Maps",Proc. 22nd Int'l Conf.Theory and Applications of Cryptographic Techniques (EUROCRYPT '03), pp. 416-432, 2003.

[5]Melissa Chase and Sherman S.M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption",Pages 121-130 ACMNew York, NY, USA ©2009 978-1-60558-894-0.G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage",ACM Trans. Information and System Security, vol. 9, no. 1, pp. 1-30, 2006.

[6]QianWang ,KuiRen, Wenjing Lou and Jin Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing",Parallel and Distributed Systems,

IEEE Transactions on Volume:22 , Issue: 5 Page(s):847 – 859.

[7]Giuseppe Ateniese , Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner ,Zachary Peterson and Dawn Song, "Provable Data Possession atUntrusted Stores",Proceeding CCS '07 Proceedings of the 14th ACM conference on Computer and communications security Pages 598-609.

[8]Mehul A. Shah Ram Swaminathan and Mary Baker, " Privacy-Preserving Audit and Extraction of Digital Contents",HP Labs Technical ReportNo. HPL-2008-32.

[9]Ari Juels1 and Burton S. Kaliski, "PORs: Proofs of Retrievability for Large Files" , Proceeding CCS '07 Proceedings of the 14th ACM conferenceon Computer and communications security Pages 584-597.

[10]V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attri-bute- Based Encryption for Fine-Grained Access Control of Encrypted data," in Proceedings of the 13th ACM Conference on Computer and Communications Secu-rity (CCS '06). ACM, 2006, pp. 89–98.

[11]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. 239–248, 1983.

[12]G. C. Chick and S. E. Tavares, "Flexible Access Con-trol with Master Keys," in Proceedings of Advances in Cryptology – CRYPTO '89, ser. LNCS, vol. 435. Springer, 1989, pp. 316–322.

[13]W.-G. Tzeng, "A Time-Bound Cryptographic Key As-signment Scheme for Access Control in a Hierarchy," IEEE Transactions on Knowledge and Data Engineering (TKDE), vol. 14, no. 1, pp. 182–188, 2002.

[14]G. Ateniese, A. D. Santis, A. L. Ferrara, and B. Masuc-ci, "Provably-Secure Time-Bound Hierarchical Key As-signment Schemes," J. Cryptology, vol. 25, no. 2, pp. 243–270, 2012.

[15]R. S. Sandhu, "Cryptographic Implementation of a Tree Hierarchy for Access Control," Information Pro-cessing Letters, vol. 27, no. 2, pp. 95–98, 1988. [16]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 (INFOCOM '04). IEEE, 2004.

[17]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. 2067– 2071.

[18] J. Benaloh, "Key Compression and Its Application to DigitalFingerprinting," Microsoft Research, Tech. Rep., 2009.

[19] B. Alomair and R. Poovendran, "Information Theoretically SecureEncryption with Almost Free Authentication," J. UCS, vol. 15,no. 15, pp. 2937–2956, 2009.

[20] D. Boneh and M. K. Franklin, "Identity-Based Encryption from theWeil Pairing," in Proceedings of Advances in Cryptology – CRYPTO'01, ser. LNCS, vol. 2139. Springer, 2001, pp. 13–229.

Author Profile

Dr.Shaik Abdul Muzeer
Professor & Principal
Megha institute of Engineering & Technology for Women

Dr.S.A.Muzeer, at present working as a principal of Megha institute of engineering & Technology has completed his PG and P.HD in Electronics & Communication Engineering and published around 25 Papers in National & International Journals. His area of research is Digital signal processing and Bio-medical engineering