A Novel Approach of Providing Privacy for the Cloud Data by Using Group Key Mechanism

Midathana Rajeswari [#1], A.V.D.N.Murthy [*2], Ramesh Challagundla [*3]
*M.Tech Scholar [#1,] Assistant Professor [*2,] Professor & Principal[*3]*
Department of Computer Science & Engineering,
Pydah College of Engineering, Gambheeram,
Visakhapatnam, AP, India.

Abstract: **Now a day's cloud computing has achieved a lot of users attention for storing and retrieving their valuable data to and from server. This is mainly because of reason like all MNC and IT companies. Generally this is formed by interconnecting a large number of systems connected all together for remote servers hosted on internet to store, access, retrive data from remote machines not from local machines. As the cloud server has the capability to store a lot of valuable data on its memory block, a lot of users can connect with the centralized location to access, retrieve and modify the data which is stored on the cloud server. The major problem in the current cloud is there is no security for the cloud data in the current days because the cloud users will form as a group to access the data at the beginning and once if the user change from one group to other it is very tedicious to manage the accounts. As the users frequently change from one group to other, it is very difficult to maintain the updated keys for the remaining users in order to access the data which is stored on multi owner cloud. So in this paper, we have implemented a novel secure multiowner data sharing scheme, named Mona, for dynamic groups management in the cloud. By leveraging group signature and dynamic broadcast encryption techniques, any cloud user can anonymously share data with others. Here in the proposed system we made the scheme independent of number of users who revoked during the data sharing. So by conducting various experiments on our proposed system we finally came to an conclusion that this proposed system is best in providing security for the data sharing over multiple owners who reside in multi groups.**

Keywords: **Multi Keyword Ranked Search, Encryption, Authentication, Identity Verification.**

1. Introduction

Cloud Computing is one among the fascinating domain in recent analysis space wherever all the info is always processed remotely in unknown machines those users don't own or operate. As cloud computing has raised user attention in storing their valuable knowledge however limits in allocating resources dynamically. The Cloud computing presents a replacement thanks to supplement the present consumption and delivery model for IT services supported the web, by providing for dynamically ascendable and sometimes virtualized resources as a service over the net. Now a days there was a variety of notable business and individual cloud computing services, together with Yahoo, Silicon House, Amazon, Google, Microsoft, and sales division. Moreover, users might not recognize the machines that truly method and host their knowledge [1]. Whereas enjoying the convenience brought by this new technology, users additionally begin worrying regarding losing management of their own knowledge [2].

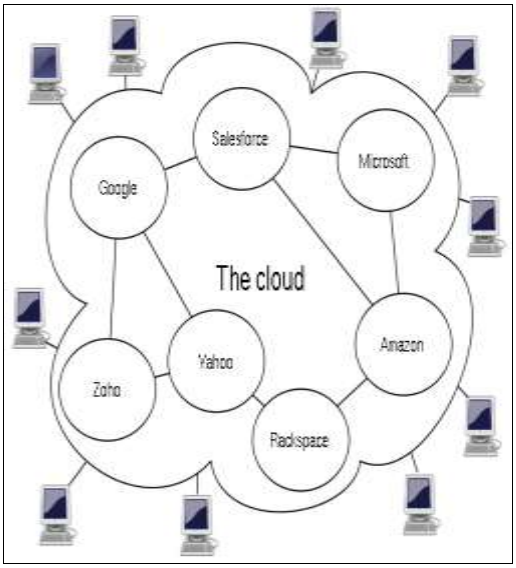


Figure. 1. Represents the Various Cloud Service Providers

From the figure 1, we can clearly represent various types of cloud service providers that are available in current days. All these service providers provide a rich facility to store the valuable data on their individual memory blocks that are provided for the individual cloud users. As we know that a lot of users shown their attention towards the usage of cloud servers for storing their valuable sensitive or private data, there was some limitations that were still not however resolved in cloud server.

Cloud computing is a technology that exploits the internet and central remote servers to uphold data and applications. Cloud computing permits customers and businesses to use applications without installation and entrée their individual files at any computer with internet access. This technology permits for much more competent computing by centralizing storage, memory, processing and bandwidth. Cloud computing is a wide-ranging solution that conveys IT as a service. Before the cloud was developed almost all the current websites and server-based applications were accomplished on a specific system. Cloud computing is broken down into three divisions application, storage and connectivity. Cloud computing is entirely real and will affect almost all and sundry. In this day and era, we have all developed into stakeholders in the computing movement, and we are all affected when major hangs occur. Remember how things tainted when the Internet came along? Changes in computer technology appear to move at lightning speeds. It wasn't that long ago those desktop computers had 20MB hard drives and people relied on floppy disks for storage. For that matter, it wasn't that long ago that there were no desktop computers, and computing involved cardboard punch cards fed into a hopper. It should be no disclosure that fruition is upon us once again, as there have been quite a lot since the crack of dawn of the information age. In this book, we choose the term “era” because cloud compute is more than an evolution. Rather, we’re entering the type of decal shakeup that only comes around once every 20 to 30 years: a disruptive e shift in the primary computing platform-of-choice. Remember when we moved from host computers to PCs? Now, cloud computing is shifting that computing power back to hosts again. Only this time things are diverse, because those hosts have become conceptual, and are speckled all over the Internet. That is to say that computing power is being shifted to the “cloud”. Such a shift to cloud computing would not have been possible until now; because the enabling technology did not yet exist Broadband connectivity now makes cloud computing a realistic opportunity for not just larger companies, but for small businesses, SOHO operations, and individual consumers.

2. Related Work

In this section we will find the related work that was analyzed and studied in order to implement this current paper. This section will describe the work related to cloud data storage and search and also the preliminaries that are used in current paper.

A) Preliminary Knowledge

Consider a cloud data hosting service that involves three main entities:

1. Data Owner,
2. Cloud Server and
3. Data User.

The data owner may be an individual or an enterprise, who wishes to outsource a collection of documents D = (D1, D2, . . . , Dn) in encrypted form C = (C1, C2, . . . , Cn) to the cloud server and still preserve the search functionality on outsourced data.

Ci = ES[Di] is the encrypted version of the document Di computed using a semantically secure encryption scheme E with a secret key S. To enable multi-keyword ranked search capability, the data owner constructs searchable index I that is built on m distinct keywords K = (k1, k2, . . . , km) extracted from the original dataset D. Both I and C are outsourced to the cloud server. To securely search the document collection for one or more keywords K¯ ∈ K, the authorized data user uses search trapdoor (distributed by the data owner) that generates the search request to the cloud server. Once the cloud server receives such request, it performs a search based on the stored index I and returns a ranked list of encrypted documents L ⊆ C to the data user. The data user then uses the secret key S, securely obtained from the data owner, to decrypt received documents L to original view.

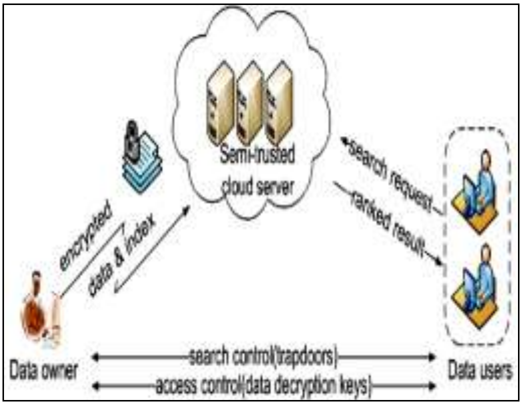


Figure. 2. Represents the Architecture of Search over Encrypted Cloud Data

We assume an honest-but-curious model for the cloud server. The cloud server is honest, that is, it is always available to the data user and it correctly follows the designated protocol specification, and it provides all services that are expected. The curious cloud server may try to perform some additional analysis to breach the confidentiality of the stored data. In the rest of the paper, the cloud server and the adversary are the same entity. That way, the adversary has access to the same set of information as the cloud server. For this work, we are not concerned about the cloud server being able to link a query to a specific user; nor are we concerned about any denial-of-service attacks.

B) Notations of Proposed Model

Let D = (D1, D2, . . . , Dn) be a set of documents and K = (k1, k2, . . . , km) be the dictionary consisting of unique keywords in all documents in D, where ∀ i ∈ [1,m] ki ∈ {0, 1} ∗ . C = {C1, C2, . . . , Cn} is an encrypted document collection stored in the cloud server. Ii is a searchable index associated with the corresponding encrypted document Ci. If A is an algorithm then a ← A (. . .) represents the result of applying the algorithm A to given arguments. Let R be an operational ring, we write vectors in bold, e.g. $v \in R$. The notation v[i] refers to the i-th coefficient of v. We denote the dot product of u, $v \in R$ as $u \otimes v = P$ i=1 $u[i] \cdot v[i] \in R$. We use |x| to indicate rounding x to the nearest integer, and |x|, |x| (for $x > 0$) to indicate rounding down or up.

3. Security Using Cryptography Techniques

Cryptography is that the study of techniques for secure communication inside the presence of third parties (called hackers/intruders) lots of sometimes, it's relating to constructing and analyzing protocols that block adversaries[5], [6] varied aspects in knowledge security like data confidentiality, data integrity, authentication, and non-repudiation are central to modern cryptography. Current cryptography exists at the intersection of the disciplines of arithmetic, technology, and technology. Applications of cryptography embrace ATM cards, laptop passwords, and electronic commerce. Cryptography before the fashionable age was effectively similar with cryptography, the conversion of information from a decipherable state to apparent nonsense. The mastermind of associate degree encrypted message shared the secret writing technique needed to recover the primary knowledge exclusively with meant recipients, thereby precluding unwanted persons from doing the same. Since warfare I and thus the appearance of the laptop, the methods accustomed perform cryptography became increasingly advanced and its application a lot of widespread.

The Modern cryptography is heavily supported mathematical theory and technology practice; crypto logical algorithms are designed around machine hardness assumptions, making such algorithms arduous to interrupt in apply by any somebody. It’s in theory potential to interrupt such a system, but it's impossible to undertake and do therefore by any superb smart means. These schemes are therefore termed computationally secure; theoretical advances, e.g., enhancements in range resolution algorithms, and faster computing technology would like these solutions to be often tailored. There exist information-theoretically secure schemes that incontrovertibly cannot be broken even with unlimited computing power—an example is that the one-time pad—but these schemes are harder to implement than the foremost effective in theory breakable but computationally secure mechanisms.

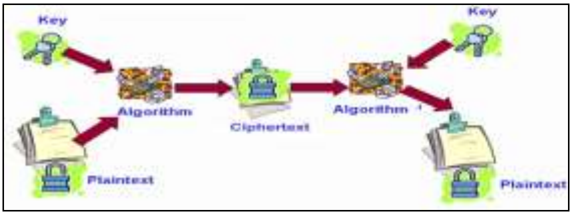


Figure. 3. Represents the Process of Encryption and Decryption of Valuable or Sensitive Data

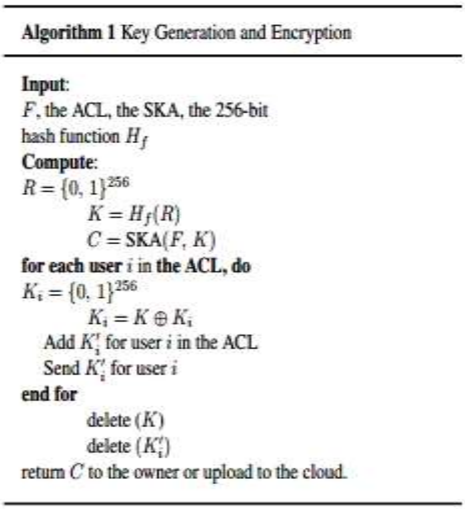
From the above figure 3, we can clearly find out that for encrypting a plain text we need a key based on the type of algorithm and once the data is encrypted with the help of a key this will be send to the receiver .The receiver will receive the encrypted data and he now try to decrypt the data with the key that was specified by the sender and then finally view the data in the form of plain text.

4. Proposed Algorithms

Here in this paper we have proposed two algorithms one for encryption and key aggregation and one for decryption. Now let us discuss about those algorithms in detail.

4.1 Key Generation and Encryption Algorithm

The below is the process for key generation and encryption.



Algorithm 1 Key Generation and Encryption

```
Input:
$F$, the ACL, the SKA, the 256-bit
hash function $H_f$
Compute:
$R = \{0, 1\}^{256}$
        $K = H_f(R)$
        $C = \text{SKA}(F, K)$
for each user $i$ in the ACL, do
$K_i = \{0, 1\}^{256}$
        $K_i = K \oplus K_i$
    Add $K'_i$ for user $i$ in the ACL
    Send $K'_i$ for user $i$
end for
        delete $(K)$
        delete $(K'_i)$
return $C$ to the owner or upload to the cloud.
```

The proposed methodology maintains a single cryptographic key for each of the data files. However, after encryption / decryption, the whole key is not stored and possessed by any of the involved parties. The key is partitioned into two constituent parts and are possessed by different entities. The following are the keys that are used within the proposed algorithm.

Symmetric Key K:
K is a random secret generated by the CS for each of the data files. The length of K in proposed is 256 bits, as is recommended by most of the standards regarding key length for symmetric key algorithms (SKAs). However, the length of the key can be altered according to the requirements of the underlying SKA. K is obtained in a two-step process. In the first step, a random number R of length 256 bits is generated such that R = $\{0, 1\}^{256}$. In the next step, R is passed through a hash function that could be any hash function with a 256-bit output. In our case, we used secure hash algorithm 256 (SHA-256). The second step completely randomizes the initial user-derived random number R. The output of the hash function is termed as K and is used in symmetric key encryption [e.g., the Advanced Encryption Standard (AES)] for securing the data.

CS Key Share Ki:
For each of the users in the group, the CS generates Ki, such that Ki = $\{0, 1\}^{256}$. Ki serves as the CS portion of the key and is used to compute K whenever an encryption/decryption request is received by the CS. Moreover, it is ensured by comparison that the distinct Ki is generated for every file user.

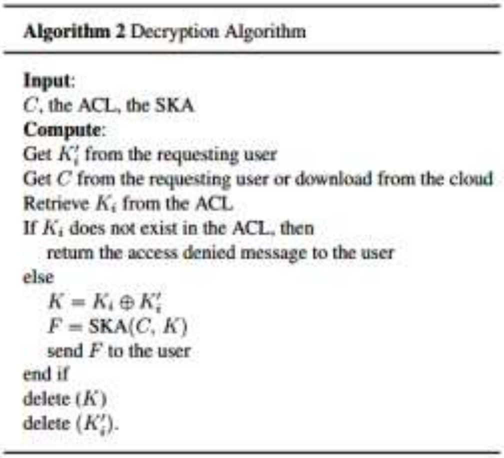
User Key Share Ki: Ki is computed for each of the users in the group as follows:

$$K\,i = K \oplus Ki.$$

The below section describes the decryption algorithm.

4.2 Decryption Algorithm

The below figure defines the decryption algorithm for the proposed paper. Now let us look at that algorithm in detail.



Algorithm 2 Decryption Algorithm

```
Input:
$C$, the ACL, the SKA
Compute:
Get $K'_i$ from the requesting user
Get $C$ from the requesting user or download from the cloud
Retrieve $K_i$ from the ACL
If $K_i$ does not exist in the ACL, then
    return the access denied message to the user
else
    $K = K_i \oplus K'_i$
    $F = \text{SKA}(C, K)$
    send $F$ to the user
end if
delete $(K)$
delete $(K'_i)$.
```

The authorized user sends a download request to the CS or downloads the encrypted file (C) from the cloud and sends the decryption request to the CS. The cloud verifies the authorization of the user through a locally maintained ACL. The decryption request is accompanied by the user portion of the key, i.e., K i, along with other authentication credentials. The CS computes K by applying XOR operation over K i and the corresponding Ki from the ACL. As each of the users correspond to a different pair of Ki and K i, none of the users can use other users' K i to masquerade identity. Subsequently, the CS proceeds with the decryption process after verifying the integrity of the file. If the correct K i is received by the CS, the result will be a successful decryption process; otherwise, the decryption will fail. After successful decryption, the file is sent to the requesting user through a secure communication channel that could be Secure Sockets Layer (SSL) or Internet Protocol Security (IPSec) channels. K is deleted via secure overwriting from the CS after decryption. The users are authenticated before the request processing according to standard procedures. The process is highlighted in above algorithm2.

Simplest form of Proposed MONA Representation

The below is the simplest form of defining the proposed algorithm. This is represented in 3 steps they are as follows:

Algorithm: Userid Key Generation Algorithm:

Input: GroupId(x) and UserId(y),

System Parameters (H, H1, H2, U, V, W, Y, Z).

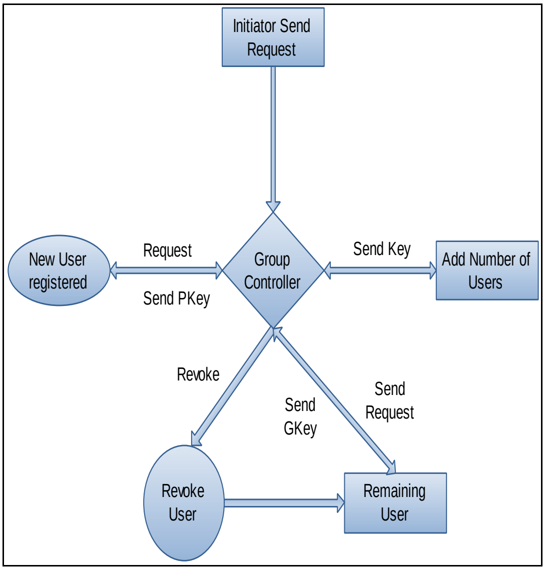
Output: Generate Valid Key Generation.

4.3 Data Flow for the proposed Algorithm

The below section is mainly used to describe the data flow diagram for the proposed algorithm, now let us discuss about the proposed paper in detail. This is represented in following two data flow diagrams which is shown clearly in below section.

Data flow diagram for the Group Controller handling a list of users

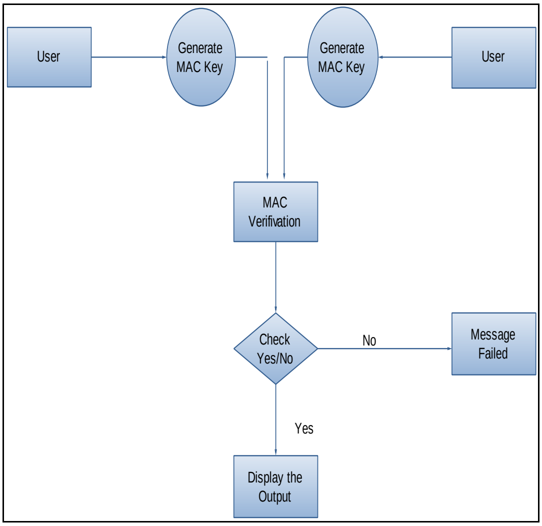
In this section we will mainly look at the data flow diagram which represents the group controller which controls the new user who gets registered as well as the new key generation at the time of user revocation.



From the above figure we can clearly tell that group controller is a person who will be operating in the center and who will take new users into the cloud server and it will then initiate the nodes to participate for the sending the data to and from the cloud. Now the same group controller can also operate the facility like revoking the user's key whenever needed. If any user leave or come out from the group the group controller will immediately revoke the group key of that existing users.

Data flow diagram for the Group Controller generating MAC for data while encryption and decryption

In this section we will mainly define the MAC generation for the data which is encrypted and stored into the cloud, the MAC should be of valid type at the time of decryption, if MAC fails the receiver can get an idea that data is not from original source it have been modified by any intermediate source during transmission.



From the above figure we can clearly get an idea that if MAC verification fails the data cant be viewed at the receiver end.The data will be opened in a plain text for the valid user with MAC address as Success.

5. Pseudocode for the Proposed Encryption Algorithm

As the proposed paper is implemented with a new cryptography technique like encryption and decryption, it is done with the help of AES algorithm. Here in the below section we can clearly get an idea about that AES algorithm in details with the following detailed explanation.

AES Encryption Algorithm

Step 1: As per the application in the step 1 ,we will upload or browse a file in order to encrypt that before it is stored into the server.

Step 2: (Encryption of the actual data begins here)

Let the message to be transmitted be "CRYPTOGRAPHY".

First find the ASCII equivalent of the above characters.

C R Y P T O G R A P H Y
67 82 89 80 84 79 71 82 65 80 72 89

Step 3: Now add these numbers with the digits of the Armstrong number as follows

```
    67 82 89 80 84 79  71  82   65  80 72 89
(+)1    5    3  1  25  9   1    125 27  1    5   3
-----------------------------------------------
   68 87 92 81 109 88 72 207 92 81 77 92
```

Step 4: Convert the above data into a matrix as follows

A=

$$\begin{bmatrix} 68 & 81 & 72 & 81 \\ 87 & 109 & 207 & 77 \\ 92 & 88 & 92 & 92 \end{bmatrix}$$

Step 5: Consider an encoding matrix...

B = $\begin{bmatrix} 1 & 5 & 3 \\ 1 & 25 & 9 \\ 1 & 125 & 27 \end{bmatrix}$

Step 6: After multiplying the two matrices (B X A) we get

C =

$$\begin{bmatrix} 779 & 890 & 1383 & 742 \\ 3071 & 3598 & 6075 & 2834 \\ 13427 & 16082 & 28431 & 12190 \end{bmatrix}$$

The encrypted data is...

779, 3071, 13427, 890, 3598, 16082, 1383, 6075, 28431, 742, 2834, 12190

The above values represent the encrypted form of the given message.

1) Decryption

Decryption involves the process of getting back the original data using decryption key. The data given by the receiver (the color) is matched [6] with the data stored at the sender's end. For this process the receiver must be aware of his own color being assigned and the key values.

Step 1: As per the application the receiver first chooses the valid data which he want to download it, once he choose that file, he then do the below steps internally in order to view the data in a decrypted manner.

Step 2 :(Decryption of the original data begins here)

The inverse of the encoding matrix is

$$D = (-1/240)*\begin{bmatrix} -450 & 240 & -30 \\ -18 & 24 & -6 \\ 100 & -120 & 20 \end{bmatrix}$$

Step 3: Multiply the decoding matrix with the encrypted data

$$\begin{bmatrix} 68 & 81 & 72 & 81 \\ 87 & 109 & 207 & 77 \\ 92 & 88 & 92 & 92 \end{bmatrix}$$

(D X C) we get

Step 4: Now transform the above result as given below

68 87 92 81 109 88 72 207 92 81 77 92

Step 5: Subtract with the digits of the Armstrong numbers as follows

```
   68 87 92 81 109 88 72 207 92 81 77 92
(-)1   5  3   1   25   9   1  125 27  1   5   3
-----------------------------------------------------
   67 82 89 80 84 79 71 82 65 80 72 89
```

Step 6: Obtain the characters from the above ASCII
Equivalent

67 82 89 80 84 79 71 82 65 80 72 89
C R Y P T O G R A P H Y

6. Implementation Modules

Implementation is the stage where theoretical design is converted into programmatically way. Generally in the implementation stage we will divide the application into number of modules in order to make the application develop very easily. search over encrypted cloud data. The application is divided mainly into following 5 modules. They are as follows:

1. Group Manager Module.
2. Group Member Module.
3. Cloud Module.
4. File Security Module.
5. User Revocation List.

1) Group Manager Module

Group manager takes charge of followings things. They are as follows:

1. System parameters generation,

2. User registration,

3. User revocation, and

4. Revealing the real identity of a dispute data owner.

Therefore, we assume that the group manager is fully trusted by the other parties. The Group manager is the admin. The group manager has the logs of each and every process in the cloud. The group manager is responsible for user registration and also user revocation too.

2) Group Member Module

Group members are a set of registered users that will store their private data into the cloud server and Share them with others in the group. Note that, the group membership is dynamically changed, due to the staff resignation and new employee participation in the company. The group member has the ownership of changing the files in the group. Whoever in the group can view the files which are uploaded in their group and also modify it.

3) Cloud Module

In this module, we create a local Cloud and provide priced abundant storage services. The users can upload their data in the cloud. We develop this module, where the cloud storage can be made secure. However, the cloud is not fully trusted by users since the CSPs are very likely to be outside of the cloud users' trusted domain. Similar to we assume that the cloud server is honest but curious. That is, the cloud server will not maliciously delete or modify 8 user data due to the protection of data auditing schemes, but will try to learn the content of the stored data and the identities of cloud users.

4) File Security Module

In this module we use the following things like

1. Encrypting the data file.
2. File stored in the cloud can be deleted by either the group manager or the data owner (i.e., the member who uploaded the file into the server).

5) User Revocation Module

User revocation is performed by the group manager via a public available revocation list (RL), based on which group members can encrypt their data files and ensure the confidentiality against the revoked users.Now in the below section we will define the pseudocode for the proposed algorithm.

7. Conclusion

In this paper, we have design a secure data sharing scheme, Mona, for dynamic groups in an entrusted cloud. In Mona, a user is able to share data with others in the group without revealing identity privacy to the cloud. Additionally, in this we support efficient user revocation and new user joining. More specially, efficient user revocation can be achieved through a public revocation list without updating the private keys of the remaining users. In this secret key is used for authorized user to access the data in cloud could not allow authorized user. Also in this paper as an extension we have also implemented a new concept like encrypting the cloud without affecting the original data by an un-trusted user. As the cloud servers which are available in the current cloud don't have any facility to encrypt the data before storing into the cloud in this paper we have implemented a new concept like encrypting the cloud before it store into the cloud server.

References

[1] M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. *J. Cryptology*, 22(1):1–61, 2009.

[2] K. Zhang, X. Zhou, Y. Chen, X.Wang, and Y. Ruan. Sedic: privacy aware data intensive computing on hybrid clouds. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS'11, pages 515–526, New York, NY, USA, 2011. ACM

[3] S. Quinlan and S. Dorward. Venti: a new approach to archival storage. In *Proc. USENIX FAST*, Jan 2002.

[4] E.-J. Goh, "Secure Indexes," Cryptology ePrint Archive, http://eprint.iacr.org/2003/216. 2003.

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

[6] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Plutus: Scalable Secure File Sharing on Untrusted Storage," Proc.USENIX Conf. File and Storage Technologies, pp. 29-42, 2003.

[7] E. Goh, H. Shacham, N. Modadugu, and D. Boneh, "Sirius: Securing Remote Untrusted Storage," Proc. Network and Distributed Systems Security Symp. (NDSS), pp. 131-145, 2003.

[8]. Professional Java Network Programming

[9].Computer Networking: A Top-Down Approach, by James F.Kurose.

9. About the Authors

Midathana Rajeswari is currently pursuing her 2 years M.Tech in Department of Computer Science and Engineering at Pydah College of Engineering and Technology, affiliated to JNTUK University, AP, India. Her area of interest includes Networks and Cloud Computing.

A.V.D.N.MURTHY completed his MCA and M.Tech. in Computer Science and Engineering. He is currently working as Assistant Professor of Department of Computer Science and Engineering at Pydah College of Engineering and Technology, affiliated to JNTUK University. He is having industrial experience of 1.2 years and teaching experience of 9 years. His areas of interest include Data mining, Image Processing, Cryptography & Network security, Computer Networks and Operating Systems.

Dr.Ramesh Challagundla received his M.Sc. Degree in Phy.Electronics from Meerut University which is recognized as equivalent to B.E (ECE) in the year 1988, M.E. With specialization in Applied/Power Electronics, from Gulbarga University in 1991, was granted A.M.I.E. In 1997 and Ph.D. from Andhra University College of Engineering, Andhra university, Visakhapatnam. He joined as service engineer in Hast Alloy Castings Ltd in the year 1990. After serving a year and half, he switched over to teaching and served as Lecturer in R.E.C. Affiliated to Gulbarga University during 1992-1993. He joined EEE Department, GITAM, Visakhapatnam and served as Lecturer during 1993-96. During 1996-97 he served as Lecturer in Bhilai Institute of Technology, during 1997-98 served as Assistant Professor in Birla Institute of Technology, Mesra, Ranchi, during 1998-2001 served as Assistant Professor in GITAM College of Engineering, Visakhapatnam and from 2001 onwards with ANITS.Currently working as Professor and Principal at Pydah College of Engineering and Technology, Gambheeram, Visakhapatnam. Ratified as Professor by the expert committee of Andhra University in the field of ECE constituted by Vice-Chancellor, who himself was the chairman for the selection committee.