Secure Data Distribution in Mobile Cloud Computing

- **Shweta Bochare , Archana Bhagwat, Saroj Duche, Aditi Patankar, Prof. Mrs. Seema Hadke**

[1]*Bharati Vidyapeeth's College of Engineering for Women, Pune-43*

[2]*Bharati Vidyapeeth's College of Engineering for Women, Pune-43*

[3]*Bharati Vidyapeeth's College of Engineering for Women, Pune-43*

[4]*Bharati Vidyapeeth's College of Engineering for Women, Pune-43*

[5]*Professor at Department of Information Technology,*

6Bharati Vidyapeeth's College of Engineering for Women, Pune-43

Abstract:

Now-a-days people are using their smartphones for various purposes like uploading data, sharing data, use of online services, etc. along with their primary functions. But the problem with smartphones is that they are having limited computational and storage resources. Use of cloud computing in mobile computing environment solve this problem which also increases the capacity of mobile devices .But the major concern about the use of cloud is the security issue which also becomes the problem in mobile cloud computing environment. We proposed a light-weighed cryptographic mechanism a proxy re-encryption to solve the data integrity, data security issues in which users has to keep only short secret keys for all cryptographic operations in mobile cloud without involvement of any trusted third party.

Keywords:

Mobile Cloud Computing, Proxy Re-encryption, data integrity, data distribution.

Introduction:

The Cloud refers to a Network or Internet. Cloud is a place which is located at remote locations and gives services to public and private networks, i.e., WAN, LAN or VPN.

Cloud Computing refers to manipulating, configuring, and accessing the hardware and software resources remotely. It gives online data storage, infrastructure, and application. Cloud computing gives platform independency, as the software is not required to be installed locally on the PC. Therefore, the Cloud Computing is making our business applications mobile and collaborative.

To save data on cloud provides many benefits, i.e., low cost, good reliability and availability. But the data security issues result into privacy and integrity problems. Since mobile cloud computing combines the techniques of mobile computing and cloud computing. Many users and uploads there data on cloud through cell phones.

In this project we are going to develop an efficient data distribution system in MCC which allows mobile users to securely store their data in cloud storage services. And share their data with friends, family. We adopt several cryptographic primitive to achieve data privacy, data integrity, dynamical data modification and deletion, and flexible data distribution. Our system allows the data owner to dynamically modify and delete his data.

Problem Definition:

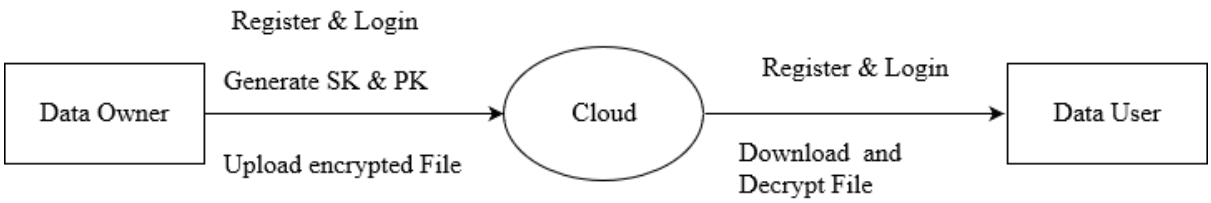
To design a framework for ensuring security of private data by using type based data distribution into different category without involvement of any trusted third party. It also achieves integrity and authentication of data owner as well as data consumer.

Objectives

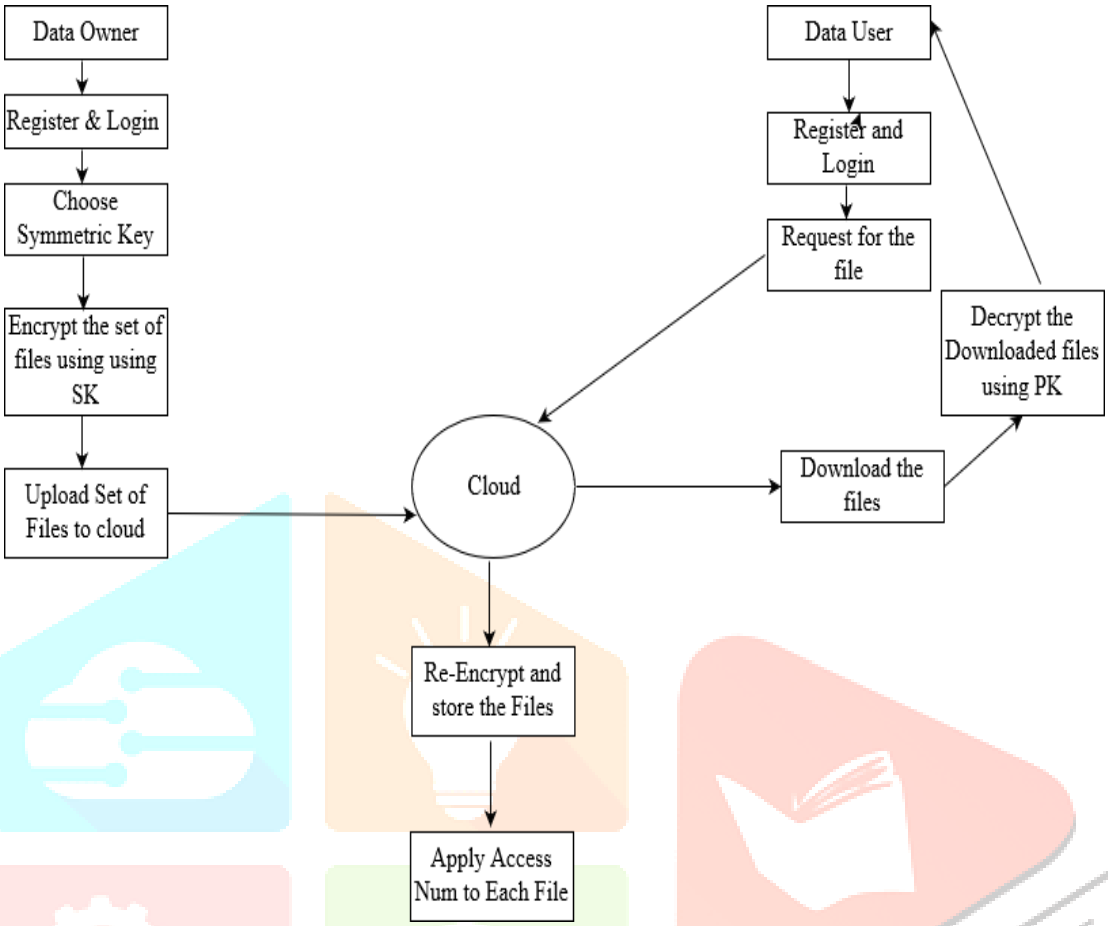
- Data confidentiality: Any user (including the cloud) without access permission cannot collude to read the private data of the data owner.
- Data integrity: Both the data owner and the data consumer can check the integrity of the data.
- Data authentication: The data consumer can authenticate the data owner's identity.
- Dynamic data operation: The data owner can perform data modification and deletion operations without affecting the confidentiality, integrity and authentication properties.
- Fine-grained data distribution: The data consumer can read the private data belonging to a category if and only if he obtains the access permission to that specific data category from the data owner.
- Lightweight: Both the data owner and the data consumer should perform all the operations with small storage, communication and computation overhead.

System design:

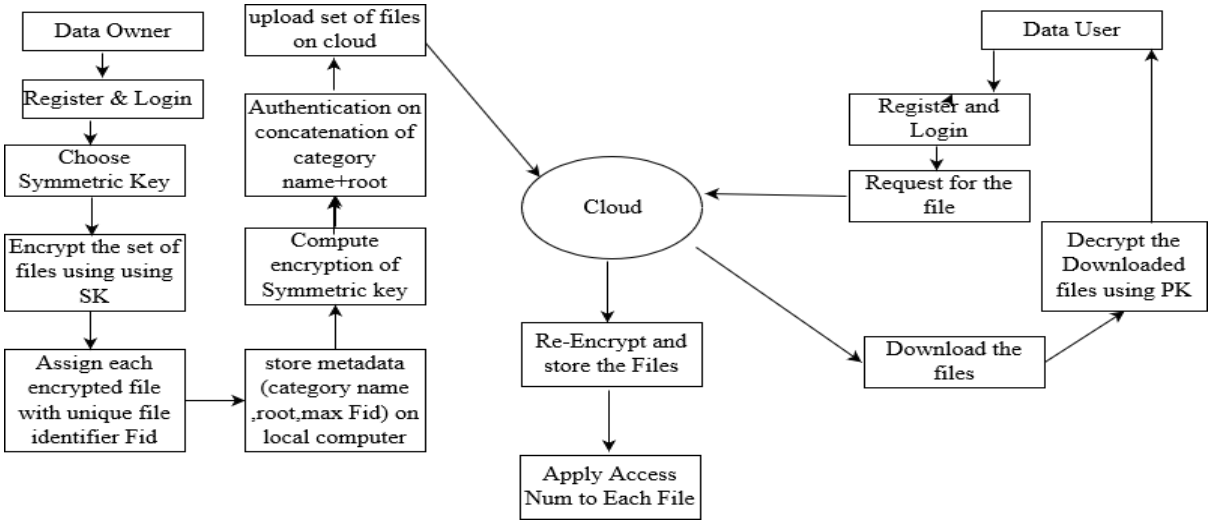
Data Flow Diagram 0:



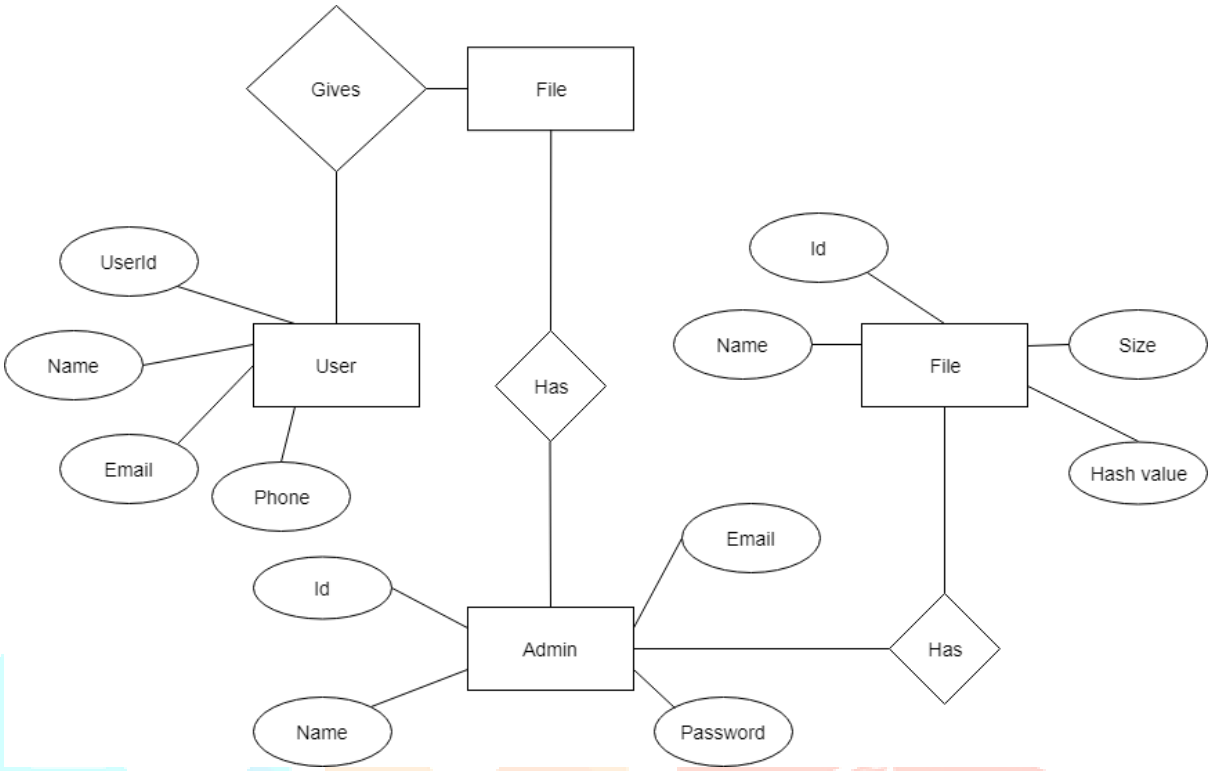
Data Flow Diagram 1:



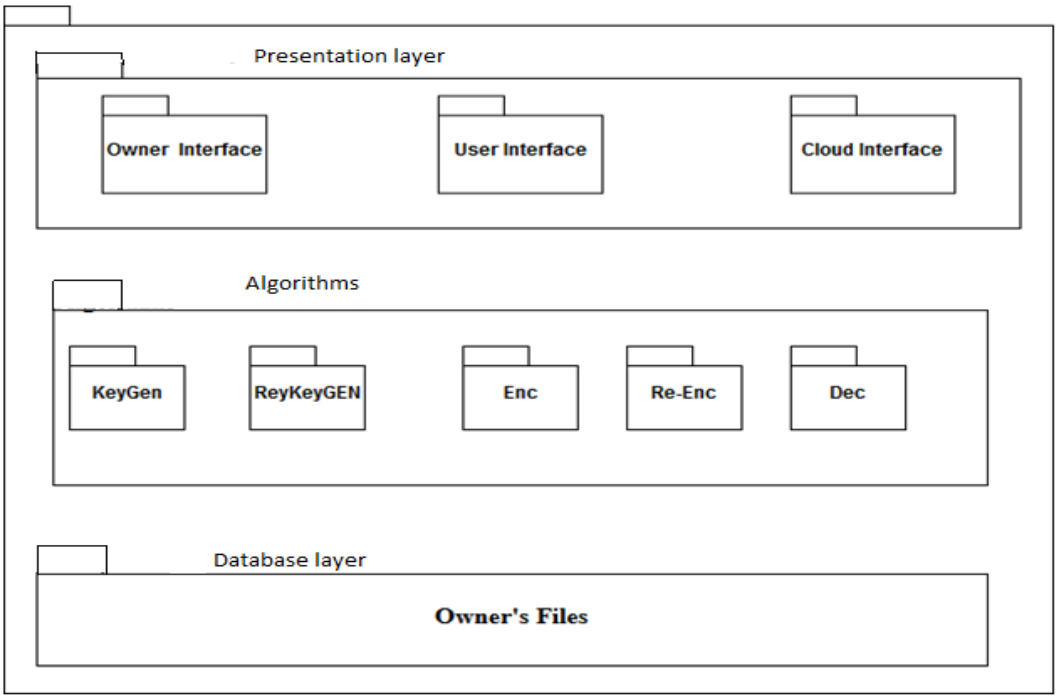
Data Flow diagram 2:



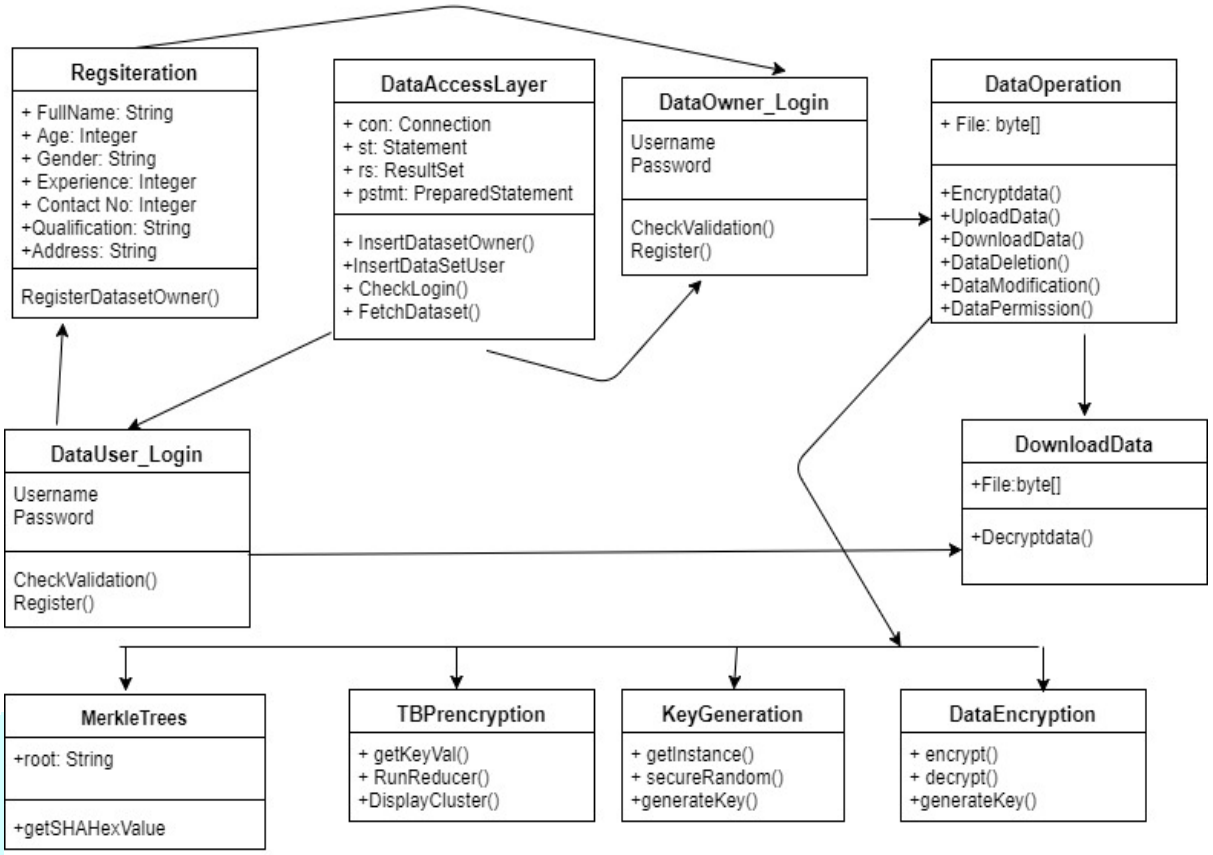
ER Diagram:



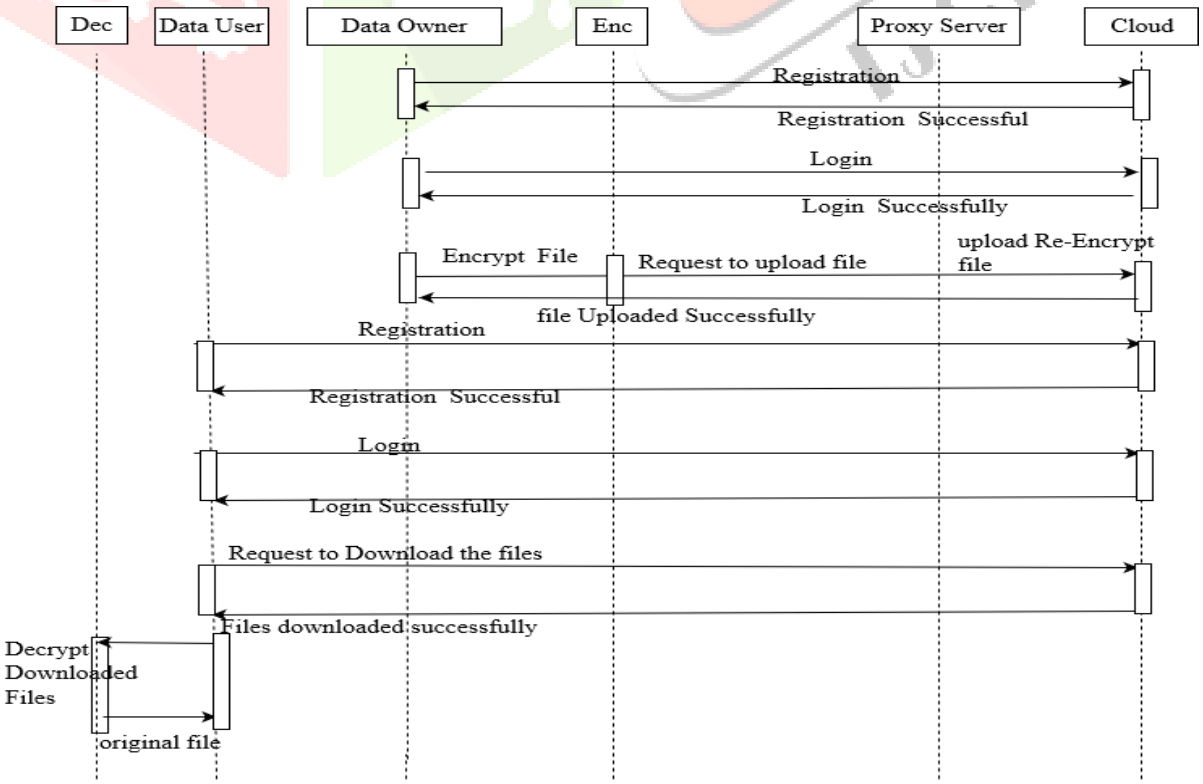
Package diagram:



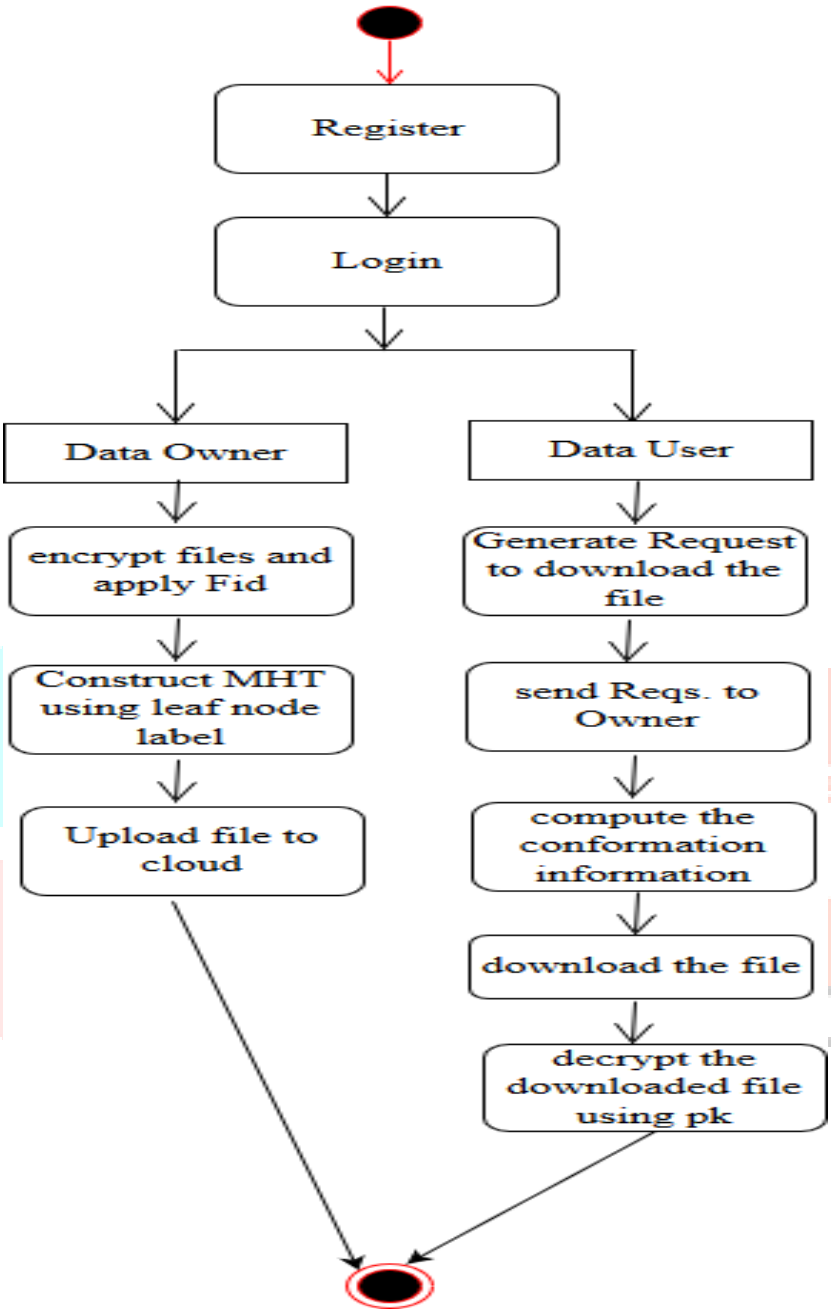
Class Diagram:



Sequence Diagram:



Activity Diagram:



System Architecture

In our system, the data owner will classify its own data into different categories. The data owner protects his data in each category with a unique type by using the TB-PRE scheme. A secure symmetric encryption, e.g. AES is employed to encrypt the data for each data category, and the TB-PRE scheme is

then used to encrypt the symmetric secret key. Since each data category may contain many data files, the data owner also constructs a Merkle Hash Tree (MHT) for each data category and only stores the metadata consisting of the root of the MHT corresponding to each category at his own local storage. Besides, the BLS signature scheme is used to authenticate the roots of the MHT such that the data consumer cloud also check the integrity of the data files as well as authenticate the identity of the data owner. System architecture of proposed system is shown in Fig.1. There are three main network entities in our data distribution system, namely, the cloud, the data owner and the data consumer. The data owner is a mobile user who stores his private data in the cloud (by different categories), and allows the data consumer to access his private data (of some category) from the cloud. The cloud is an entity that provides storage services and is responsible to help the data owner to distribute the private data (belonging to some particular category) to the data consumer. The data consumer is an entity who first gets data access permission (of some data category) from the data owner (and this only happens once per data.

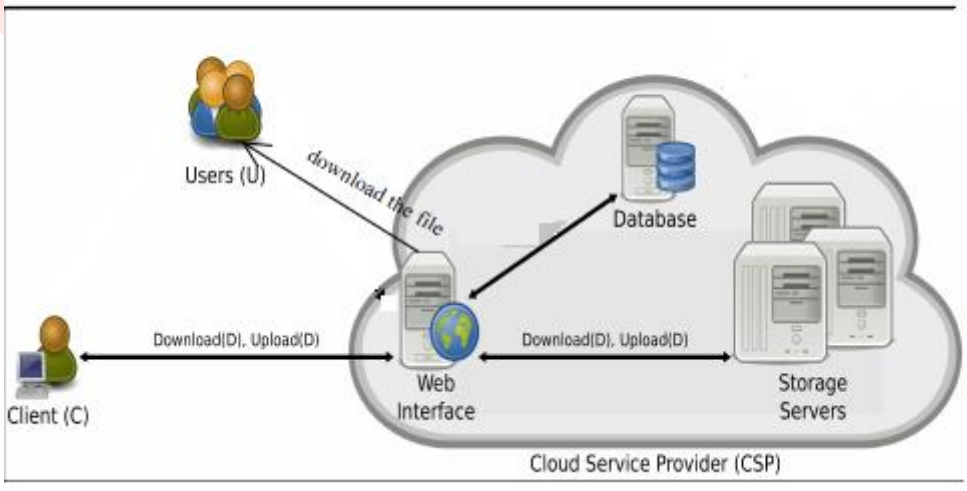


Fig 1: System Overview

Related Works

1. Provable data possession at untrusted stores.

Year: 2013

Author Name:

- Giuseppe Ateniese
- Randal burns
- Reza Curtmola
- Joseph herring,

Description: Provable Data Possession schemes provide data format independence, which is a relevant feature in practical deployments, and put no restriction on the number of times the client can challenge the server to prove data possession. A Provable Data Possession protocol checks that an outsourced storage site retains a file, which consists of a collection of n blocks. The client C (data owner) pre-processes the file, generating a piece of metadata that is stored locally, transmits the file to the server S, and may delete its local copy. The server stores the file and responds to challenges issued by the client. Storage at the server is in and storage at the client is in O(1). As part of pre-processing, the client may alter the file to be stored at the server. The client may expand the file or include additional metadata to be stored at the server. Before deleting its local copy of the file, the client may execute a data possession challenge to make sure the server has successfully stored the file.

Limitations: Considered the data privacy with respect to the third auditor, but they usually do not keep the confidentiality of the data against the cloud.

2. A practical revocasstion scheme for broadcast encryption using smartcards

Year: 2013

Author Name:

- Noam Kogan
- Yuval Shavitt
- Avishai Wool

Description: An anti-pirate revocation scheme for broadcast encryption systems proposed, in which the data is encrypted to ensure payment by users. In the systems, decryption of keys is done on smartcards and key management is done in-band. Our starting point is a scheme of Naor and Pinkas. Their basic scheme uses secret sharing to remove up to t parties, is information theoretic secure against coalitions of size t, and is capable of creating a new group key. However, with current smartcard technology, this scheme is only feasible for small system parameters, allowing up to about 100 pirates to be revoked before all the smartcards need to be replaced. We first present a novel implementation method of their basic scheme that distributes the work among the smartcard, set-top terminal, and center. Based on this, we construct several improved schemes for many revocation rounds that scale to realistic system sizes. We allow up to about 10,000 pirates to be revoked using current smartcard technology before recording is needed. The transmission lengths of our constructions are on par with those of the best tree-based schemes. However, our constructions have much lower smartcard CPU complexity: only O(1) smartcard operations per revocation round (a single 10-byte field multiplication and addition), as opposed to the complexity of the best tree-based schemes, which is poly logarithmic in the number of users.

Limitations: The solution of using the anti-pirate revocation scheme to ensure secure data-sharing usually cannot support dynamic data distribution.

3. Attribute-based encryption for fine-grained access control of encrypted data.

Year: 2015

Author Name :

- Ee-Chien Chang
- Jia Xu

Description: In proposed cryptosystem, cipher texts are labeled with sets of attributes and private keys are associated with access structures that control which cipher texts a user is able to decrypt. Each private key is associated with an access structure that specifies which type of cipher texts the key can decrypt. We call such a scheme a Key-Policy Attribute-Based Encryption (KPABE), since the access structure is specified in the private key, while the cipher texts are simply labeled with a set of descriptive attributes.

Limitations: An Attribute Based Encryption system relying on a third party authority to issue secret keys for all system users usually suffers from the key escrow problem.

4. Improving privacy and security in multi-authority attribute-based encryption

Year**:** 2012

Author Name:

- Melissa chase
- sherman s.m. chow

Description: In proposed cryptosystem, cipher texts are labeled with sets of attributes and private keys are associated with access structures that control which cipher texts a user is able to decrypt.

Each private key is associated with an access structure that specifies which type of cipher texts the key can decrypt. We call such a scheme a Key-Policy Attribute-Based Encryption (KPABE), since the access structure is specified in the private key, while the cipher texts are simply labeled with a set of descriptive attributes.

Limitations: This general method cannot completely solve the key escrow issue.

5. Achieving secure, scalable, and fine-grained data access control in cloud computing.

Year: 2011

Author Name**:**

- Shucheng yu,
- Cong wang, Kuiren

Description: System is composed of the following parties: the Data Owner, many Data Consumers, many Cloud Servers, and a Third Party Auditor if necessary. To access data files shared by the data owner, Data Consumers, or users for brevity, download data files of their interest from Cloud Servers and then decrypt. Neither the data owner nor users will be always online. They come online just on the necessity basis. For simplicity, we assume that the only access privilege for users is data file reading. Extending our proposed scheme to support data file writing is trivial by asking the data writer to sign the new data file on each update. Cloud Servers are always online and operated by the Cloud Service Provider (CSP). They are assumed to have abundant storage capacity and computation power. The Third Party Auditor is also an online party which is used for auditing every file access event. In addition, we also assume that the data owner can not only store data files but also run his own code on Cloud Servers to manage his data files.

Limitations**:** The number of secret keys of each data consumer is usually linear to the number of the data owners.

Tools Used

- **Software Requirement:**

- Operating System : windows 8 and above.
- Application Server : Tomcat5.0/6.X
- Language : Java
- Front End : HTML, JSP
- Database : MySQL

- **Hardware Requirement:**

  - Processor : Intel i3/i4/i5
  - RAM : 4 GB (min)
  - Hard Disk : 20 G/B(min)

Statistical Technique Used

We have developed Login and Registration which manages the user profiles (User and Admin), so that the users can upload the file on cloud in the encrypted format. Here data will be saved on cloud in the form of merkle hash tree. Here the hash value of the data is calculated.

Algorithm

- **BLS Signature**: This API us to generate a signature for the file.
- **Advance Encryption Standard**: In our system, we have used AES to provide encryption to the data i.e., files.

- **Secure Hash Algorithm** 1: In cryptography, **SHA-1** is a cryptographic hash function which takes an input and produces a 160-bit (20-byte) signature value known as hash value.

.

Advance Encryption Standard:

```
Cipher(byte in[16], byte out[16], key_array round_key[Nr+1])
begin
byte state[16];
state = in;
AddRoundKey(state, round_key[0]);
```

```
for i = 1 to Nr-1 stepsize 1 do
SubBytes(state);
ShiftRows(state);
MixColumns(state);
AddRoundKey(state, round_key[i]);
end for
SubBytes(state);
ShiftRows(state);
AddRoundKey(state, round_key[Nr]);
end
```

Secure Hash Algorithm:

1. add some extra data to the end of the input

```
        set the initial sha-1 values
            for each 64-byte chunk do
              extend the chunk to 320 bytes of data
              perform first set of operations on chunk[i] (x20)
              perform second set of operations on chunk[i] (x20)
              perform third set of operations on chunk[i] (x20)
              perform fourth set of operations on chunk[i] (x20)
            end
    return sha-1 values as a hash
```

Our Approach:

The system will work in three operating modes:

1. User :

The user is the holder and owner of the data file. The data will be uploaded by the user; this file will be saved on cloud in encrypted format.

2. Admin :

The admin is an intermediate between user and cloud. Here you can see which uploaded which file.

Experiment Result:

The system will provide decrypted files when user request to download particular files of different category. User will get an authentication key on their mail i.e., called as BLS Signature. This key will be different for different users so that main key will not get hacked.

Future scope:

In future, we will develop a flexible system which can work with real time data. In the future we will consider mp3, mp4 data and pdf files also and we can design the BLS Signature more accurately.

Acknowledgment:

It gives us great pleasure in presenting the preliminary project report on 'Data Distribution in Mobile Cloud Computing'.

I would like to take this opportunity to thank my internal guide Prof. Mrs. S. A. Hadke for giving me all the help and guidance I needed I am really grateful to them for their kind support. Their valuable suggestions were very helpful.

I am also grateful to Prof. Dr. D. A. Godse Head of Information Technology Department, *Bharati Vidyapeeth's College of Engineering for women, Pune* , for his indispensable support and suggestions.

In the end our special thanks to Prof. Dr.H.V.Vankudre for providing various resources such as laboratory with all needed software platforms,

Shweta Bochare

Archana Bhagwat

Saroj Duche

Aditi Patankar

(B.E. Information Technology).

Conclusion:

We propose a system in mobile cloud computing which does not involve any trusted third party and provides many properties like data integrity, data authentication, data modification, data deletion. With access control policies as well as fine-grained access control. Our system leverages a new efficient and provably secure type-based proxy re-encryption scheme, Merkle Hash tree, as well as the BLS signature to ensure the security. An extensive performance analysis and a proof-of-concept implementation show that our data distribution is practical.

References:

[1] Jiang Zhang, Zhenfeng Zhang, and Hui Guo, "Towards Secure Data Distribution Systems in Mobile Cloud Computing", DOI 10.1109/TMC.2017.2687931, IEEE Transactions on Mobile Computing, 2017.

[2] Giuseppe Ateniese, Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner, Zachary Peterson, and Dawn Song, "Provable data possession at untrusted stores", In Proceedings of the 14th ACM Conference on Computer and Communications Security, CCS '07, pages 598–609, New York, NY, USA, 2007.

[3] Noam Kogan, Yuval Shavitt, and Avishai Wool, "A practical revocation scheme for broadcast encryption using smartcards", ACM Trans. Inf. Syst. Security., 9(3):325–351, August 2006.

[4] Ming Li, Shucheng Yu, Yao Zheng, Kui Ren, and Wenjing Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption. Parallel and Distributed Systems," IEEE Transactions on, 24(1):131–143, Jan 2013.

[5] Melissa Chase and Sherman S.M. Chow, "Improving privacy and security in multi-authority attribute-based encryption", In Proceedings of the 16th ACM

Conference on Computer and Communications Security, CCS '09, pages 121–130, New York, NY, USA, 2009.

[6] Shucheng Yu, Cong Wang, Kui Ren, and Wenjing Lou, "Achieving secure, scalable and fine-grained data access control in cloud computing", In INFOCOM, 2010 Proceedings IEEE, Pages 1–9, March 2010.

[7] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage", ACM Trans. Inf. Syst. Secure, 9:1–30, February 2006.

[8] Abdul Nasir Khan, M.L. Mat Kiah, Samee U. Khan, and Sajjad A. Madani. "Towards secure mobile cloud computing: A survey", Future Generation Computer Systems, 29(5):1278 – 1299, 2013.