A FRAMEWORK BASED ON PUBLIC KEY FOR DATA SHARING IN CLOUDS

Dr P Vijaya Bharati[1] (Associate Professor),
Yang Shyr Mei[2] (19NM1A05I3)
Pragati Kumari[3] (19NM1A05D6)
V Surya Meghana[4] (19NM1A05I0)
S Geetha Niharika[5] (19NM1A05E7)
V Deepika[6](19NM1A05H4)
Vignan's Institute of Engineering for Women,Visakhapatnam
[1]pvijayabharati@gmail.com
[2]yangshyrmei24@gmail.com
[3]pragatikumari112001@gmail.com
[4]suryameghana2002@gmail.com
[5]geetha15051992@gmail.com

1. ABSTRACT

Parallel, grid, and distributed computing have all influenced the evolution of cloud computing. Cloud facilities provide extensive data storing, computing, and sharing that has become popular in recent years. For public cloud environments, information security and confidentiality are major analysis and discussion. Accordingly, An Architecture is proposed to tackle these security issues while sharing external data in the public cloud. The framework uses effective user revocation methods and flexible access control rules for subscribers to guarantee the secrecy of content in public cloud environments. Several cryptographic tools are also suggested for data owners, all of which are built on a brand-new content hash keying scheme. Subsequently, the membership-based key operation and confidentiality-oriented asymmetric encryption rules are separated which makes it possible for the deployment of the system to be flexible and scalable.
Keywords: Hash keying system, data protection, access control, user revocation mechanism, cryptographic tools, AES encryption, RSA encryption.

2. INTRODUCTION

The capacity to use a collection of computing facilities owned and managed by a mediator over the web is known as cloud based computing. Parallel, distributed, and grid computing all contributed to the creation of cloud computing. The distribution of computing is known as cloud computing where network resources, software, and data are given to a workstation, and further computing resources are provided on demand as a utility over a network, as a service rather than a product. Security is the main issue with cloud computing. Owners of data keep their information on external computers. As a result, there is a growing need for data confidentiality, authentication, and access management. Cloud service companies are putting a lot of effort into addressing the issue of cloud security. In addition, security is increasingly used to separate various cloud service providers.

Initially, the data owners upload the files over the cloud provided by any cloud agent. The files are encrypted with content hash keying system as cloud agents may not be reliable. Later when users want to access this data it needs to use the key provided by the Group Manager to decrypt it.

3. LITERATURE SURVEY

Referring to a paper by Muthi Reddy P and other authors which deals with securely sharing data in cloud, the Key Aggregate cryptosystem is used to protect both confidential and public data in the cloud. In cloud storage, it is a consistent size aggregate for flexible cypher text choices. Equipping of host machines (VMs) allowed cloud service providers to more efficiently use their resources and reap greater rewards. Any data that is kept across various cloud data centres is corrupted, recoverable using regenerative coding, and energy-saving host machines are scheduled in multiple data centres to save energy consumption.Subsequently, referring to another paper by Jianghong Wei, and authors which deals with securely sharing data using Reversible-Repositry Identification-Based Encryption. This paper introduces the functionalities of user revocation and cipher text and revise it parallelly to provide the forward/backward security of ciphertext. Likewise, this article shows an effective application of the RS-IBE and demonstrates its security within the specified security model. The performance comparisons show that the suggested scheme has benefits in terms of performance, making it possible for a useful and affordable data sharing systemWhile dealing with Multiple Users in Cloud Computing, a paper addresses the aforementioned privacy concern for cloud storing, this paper provides the privacy-preserving authentication protocol.End user, a cloud serving platform, and a reliable mediator make up the three major network entities discussed in this paper. A end user is a person who utilises the cloud for data storage and other cloud-based tasks. Multiple users may access the cloud, and various users may be affiliated with the same company. Self authority will be granted to special feature categories. A cloud service provider is the company that controls a cloud computer. The principal goal of a cloud server is to keep user data on a distant server. It is also stated to be an individual that will offer limitless storing and a variety of services. An elective organization is available that provides latest powers is a trusted third party. They carry out tasks on the users' behalf. Likewise, they are used to audit user data and support conflict.

In this literature survey the researchers educated various techniques to deal with data sharing in cloud but large distributed systems such as public cloud which are being used for data sharing and data processing are increasingly becoming vulnerable to attacks such as the leak of private and valuable information. This information can be sold to various organizations which can manipulate the private information of the users to their own benefit. Virtualization can be leveraged to increase the security of such systems by providing integrity and confidentiality. So, a transparent cloud protection framework is demanded to increase the protection of the data being shared via public cloud.

Proposed System
The aim of the work is to safeguard shared data, assure the confidentiality and integrity of data owners and authorised users, and ensure data confidentiality by guaranteeing backward and forward privacy.
• **Forward Privacy**: This functionality ensures that an end user cannot view saved information prior to join a group.
• **Backward Privacy:** This system guarantees that the secrecy of encrypted data stored in the future is unchanged by the compromise of the secret key. As a result, after leaving the group, the invalid group member is cannot be able to view data that was uploaded onto the cloud.

By accounting for the limited storage and processing power of user devices, the objective is to support efficiency and resilience. The following conditions must be met:
Data confidentiality: Our system must guard the confidentiality of the contents of external data against both nosy mediators and nefarious end users.

Users belonging to different groups and having different permissions should be subject to variable security measures thanks to flexible access control. These access control guidelines ought to provide assurances on the privacy of the information contained in the outsourced data.
Effective user removal: End users in a group should not be affected by the leaving of a group member. In other words, the task is to describe a smooth leaving of a group member that does not need updating the secret keys of the non-revoked members, in contrast to conventional fine-grained access control schemes.
Low computation overhead: The suggested algorithms must also have a low level of client-side processing complexity.
Inarticulate overhead should reduce bandwidth consumption by depending on cheap communication.
Low storage costs: When developing our approach, the constrained storage capacities of user devices were crucial. Low client side storing costs are therefore strongly advised

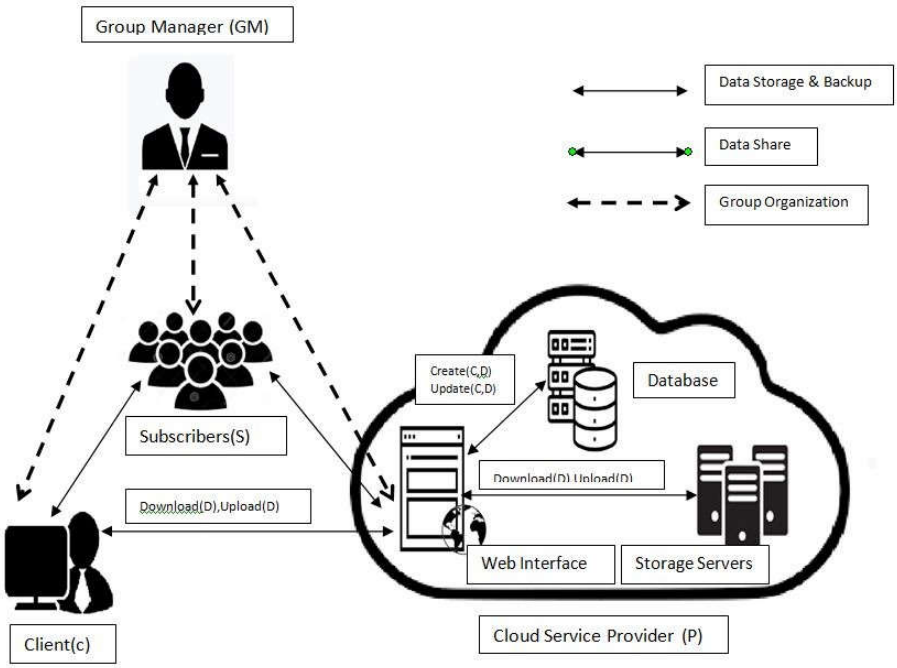
3.1. **Base Paper Explanation**

Our suggested system offers the data owner several cryptographic tools to ensure the protection of the external data and to assure that only authorised tenants can acquire the decrypting data keys in order to safeguard outsourced data in public cloud servers from unauthorised parties. The system is built on convergent encryption, a data hashed keying cryptographic technique. In other words, it shows data encrypting level and key encrypting level as two different encryption levels:
• Symmetric data encryption level – The data owner encrypts the files using a symmetric AES algorithm before outsourcing data to cloud computers. That is, the enciphering data key is derived from the file's content using a one-way hashing algorithm. Consequently, there are several options for convergent cryptography.
First, data storage is not lost because multiple users producing the same encrypted data is only kept once. Accordingly, to maintain the storage service's efficiency, redundant files are therefore kept to a minimum. Second, Convergent encryption creates a per-data ciphered key when content sharing is necessary, solving the common problem of key sharing. Lastly, knowing the plaintext is a need for producing the decoding data key.
• Asymmetric key encryption level – Using the recipient's public key and an asymmetric method, the depositor encodes the decryption key. Then, to ensure flexible access rules, the Group Manager zips the encrypted key and the encrypted data. The encoded file can then be decoded by the user.
The Dual encryption, where data is encrypted with one key and the decrypting keys are encrypted with another, ensures data confidentiality. This also allows for flexible access control policies, as only those with the appropriate decrypting keys can access the data.. The processes incorporate a joint layer for data and another for management. We present data operations and the associated encryption keys in the data layer, including GenerateParameters, EncryptData, DecryptData, EncryptKeyOneToOne, EncryptKeyOneToMany.
Proposed System Architecture

This section presents a illustrative network framework for the proposed architecture, outlining the essential entities required for effective client data management.
• Cloud Service Provider (CSP): A CSP has the ability to govern and manage allocated cloud storing servers and databases, and offer virtual infrastructure for hosting application services. Clients can utilize these services to manage their data stored in the cloud.
• Data Depositor: A data depositor utilizes the services of the mediator to upload, and access data with numerous users. A data proprietor may be a person or an organisation.
• Group Manager (GM): A group manager is in control of creating groups, creating system parameters, registering users, and revoking users. Consequently, we presume that the other organisations have faith in the group manager.

• Users: Access privileges determine a user's permissions to read, write or modify data in the cloud, based on permissions granted by the data owner. Different user groups can be identified based on their access privileges.

Block Diagram



3.2. Advantages of Proposed System

The proposed methodology employs symmetric encryption on the data and asymmetric encryption on the key to guarantee the confidentiality of cloud-stored information, ensuring secure and reliable data management..

Ensuring the security of data against forward and backward security issues, as well as access control concerns caused by insider threats, is critical.

Data Confidentiality, access control as well as smooth user revocation is obtained.

System Methodology

In this section, the desired framework based on public key in the cloud environment securely allows sharing and forwarding of files within a group without involving re-encryption .

3.3. Entities

The proposed framework has the following entities:

(i) Cloud

Cloud services offer users a convenient way to store their data, but privacy breaches are a concern. To ensure confidentiality, data is stored encrypted on the cloud without requiring changes to the cloud protocol or implementation. Basic file upload and download operations are all that is needed.

(ii) Group Manager (GM)

The Group Manager (GM) is a reliable entity responsible for performing critical operations like key management, managing users' public keys, ensuring confidentiality, and facilitating secure data exchange among group members.

(iii) End Users

In the storage cloud, users can be both clients and data owners. Each file has one owner who controls access rights for other group members, managed by the GM.

3.4. Cryptographic Algorithms

The framework relies on dual encryption. That is, it proposes two encrypting steps: Data/File encrypting and subsequent Key encrypting as follows:

4.2.1 Encryption on the Data Owner side

SymmetricAES Encryption on the Data

Before uploading data/file onto the cloud, Data owner encrypts file contents, using a symmetric key based algorithm. That is, a key is generated to encipher the plain text file. It assures the confidentiality, integrity, and authenticity of data, making it a vital tool for securing information in today's digital age.

Asymmetric RSA Encryption on the Key

The data owner encrypts the decrypting data key, based on an asymmetric RSA algorithm, using the public key of the end user or the receiver which is maintained by the group manager (GM). Then, the encrypted file/data contents and the underlying encrypted key are zipped together, ensuring flexible access policies. Certainly, member in the group may obtain the zipped file, to decrypt the encrypted data key, using his/her private key.

4.2.2 Zipping the file and the Key

After the data and the secret key of the encrypted data has been encrypted, both the entities are zipped together and uploaded onto the cloud hence providing an additional level of security.

4.2.3 Decryption at the User Side

After the user logins into the cloud and wishes to download the file, the interface provided by our framework has a decrypt button which first initiates the unzipping task followed by decryption on two levels: Key decryption level followed by data decryption level.

RSA decryption on the Key

Upon receiving the zipped file, the user unzips the file and obtains the encrypted file and the encrypted Key. Only the authorized user can decrypt the encrypted key file. RSA decryption algorithm is applied to the file to get the deciphering secret key.

AES Decryption on the File

The decryption process requires the exact key that was used to encode the data/file. Upon receiving the deciphering key, the AES decryption is applied. Hence, the user will be able to get the desired data with a secure framework.

The system needs a group manager, client, and data owner; communication between these entities can be facilitated via Java socket programming. For the purpose of managing contact between customers and data owners, the group manager can serve as a central hub. Clients can seek access to data from the data owner by establishing a socket connection with the group manager. By forwarding communications between the two endpoints, the group manager can then facilitate the data flow between the client and data owner. A versatile and effective method of managing these connections and data transmissions is provided by Java socket programming. Developers may build dependable and scalable systems that can handle massive volumes of data and several connections at once by learning how to generate and manage sockets.

4. CONCLUSION & FUTURE SCOPE

The idea of presence of pair of private key and public key makes the framework more modular and ensures access control. It also eliminates the concept of rekeying where the same secret key was shared and newly joining member was able to access the previous data. The generation of a random key makes sure that it can't be predicted by an intruder and implementation of dual encryption ensures an extra level of security. With the presence of a trusted GM, the management of secret key in one-to-many communication can be easily handled in a secure way. The interface of our framework can be made more secure using two-factor authentication at the Time of login. There can be a library which stores the previously enciphered files, its corresponding plain text and key. So same plain text files need not be encrypted over again but the previous record can be shared. Making the framework more optimized.

5. REFERENCES

1. https://kinsta.com/blog/cloud-security/#threat-intelligence-monitoring-and-prevention

2. A comprehensive Review on Secure Data Sharing in Clouds by Muthi Reddy P & others

3. Data Sharing in cloud environment adressing on Confidentiality, Integrity & access control by V rajkumar & others.

4. Securely sharing data among multiple users by Aishwarya Shetty& others.

5. Securing distributed Storage with proxy- reccryption scheme by Ateniese & others.

6. securely sharing data using Reversible-Repositry Identification-Based Encryption by Jianghong Wei & others.

7. A framework for file sharing using asymmetric key in distributed storage by V vijayakumar .

8. Framework for cloud based sharing of data preserving the privacy by Felix Horandner & others.

9. Attribute based cryptography framework for sharing the data in Hybrid Cloud by E poornima & others.

10. A multi cloud framework using the IBBE key management system for secure data storage by Manreet Sohal & others.

11. Sharing data and access control using Aggregate key by Rahul Veeravalli.

12. Secure data sharing using web based cloud storage by Shuzhou Sun & others.

13. Privacy based framework for authenticated access by Sana Belguith & others.

14. Secure data sharing in dynamic groups using smooth access control policy .

15. Data Security framework in Inter cloud by Syeed Imran Akhtar & others.

16. A framework based on different algorithms for cloud security by Jannnatul ferdous & others.

17. Efficient Revocation Scheme based on attibute in clouds by Asmai Alotaibi & others.