

A SOPHISTICATED TRAIT SUPPORTED ENCRYPTION IN PUBLIC CLOUD STORAGE FOR RESTRICTED SHARED ACCESS CONTROL MECHANISM

Abdul Aqeel¹, Mohammed Abdul Rawoof² and Mohammed Sanaullah Qaseem³

Research Scholar, Dept. of Computer Science & Engineering, NSAKCET, HYD ¹

Associate professor, Dept. of Computer Science & Engineering, NSAKCET, HYD ²

Professor, HOD, Dept. of Computer Science & Engineering, NSAKCET, HYD ³

Abstract: - - In recent times, with the emergence of cloud technologies and exponential growth of cloud storage systems, many cloud-based services have evolved and public cloud storage is gaining prominence due to the availability of cloud storage at much cheaper prices when compared to private cloud systems. However as cloud storage is public, there is a risk of privacy loss and other security concerns. In general, the public cloud storage systems give subcontracts to semi-trusted cloud servers for the storage of data. These cloud servers do not fall into the scope of trust of the data owners and there is a risk that the service providers might get access to the data owners' sensitive data. The main challenge in this project is how to regulate the access of cloud servers over the data owner's sensitive data. In general, the data that is stored on such public cloud storage systems is encrypted using cryptographic techniques to prevent unauthorized usage. A sophisticated trait supported encryption has emerged as a novel technique that can enforce access policy restrictions over the data traits. But existing trait-based techniques give no support for collaboration. In this project, we try to discover and come up with a new trade-based cryptographic technique that can help Multiple users with multiple sets of trade data to collaborate and access the data owners' data. Also, we tried to implement a collusive detection scheme that identifies any collaboration that is not part of the access policy ask allusion or unauthorized access, and such access requests would be denied. When multiple user groups collaborate to ensure data there is a high chance that there would be duplicate documents in the same environment hence we proposed a technique for the deduplication of content present on the cloud nearby reducing the costs incurred due to duplicate data. Security analysis and performance analysis of the recommended technique show that the proposed system fetches good results and it is implementable in real-time environments.

Keywords: - *Public cloud storage, Access control, CP-ABE, Collaboration.*

I.INTRODUCTION

In recent times, cloud computing is seen as one of the fastest-growing markets as it helps to reduce the investment in hardware and infrastructural resources by leasing them out to many small scale IT companies that cannot afford the investment in infrastructure [1]. It is both a win-win situation for the cloud service provider as well as the cloud service receiver. Cloud computing has made its presence in many fields including grid computing, utility computing, distributed computing, etc. The consumers of the cloud resources can utilize the cloud infrastructure charging them as per their use. As the prices are relatively low many small-scale enterprises add individuals are using the cloud services and cloud platforms [3]. Despite having many advantages with the cloud environment, privacy and security are still aspects of concern when it comes to sharing of data. In general, the cloud customer or the data owner stores this data on the cloud on reliable cloud storage servers controlled and administered by a fully trusted

administrator [5]. But in public service cloud storage systems, the storage services are outsourced to semi-trusted third party vendors to take care of the administration activities. Data owners do not have their data in their trusted domain and the third-party cloud service administrators can gain access to the sensitive information uploaded by the data owner [7]. Hence this is a serious security threat and privacy issue that needs to be addressed. Traditional security mechanisms cannot be applied in this scenario. Nowadays, trait-based encryption has emerged as one of the key techniques to provide a resolution but the above-mentioned privacy issue. Trait-based encryption is appropriate in this scenario as the data owner can have direct control over his data and can provide fine-grained access services [9]. In a trait-based encryption mechanism, every user will have a set of traits over which the access policies are defined and the ciphertext is fixed on certain traits That belong to the access policy [2]. However, the trait-based mechanisms can only permit access to those users who have the traits

mentioned in their access policy. There would be multiple scenarios or situations where access to certain secret information should be obtained by a single user and this is not possible with the above-mentioned trait-based mechanisms [4]. For instance, some files might have to be shared across people in a single organization with different responsibilities, roles, and positions. All these individuals have the responsibility of protecting the confidentiality of the document. Currently, fine-grained trait-based mechanisms are still under development in the study and do not support collaborative environments[6]. Hence, we need to define trait-based encryption techniques for both collaborative and non-collaborative environments. The traditional trait-based mechanisms are suitable for non-collaborative environments. In this project, we try to recommend fine-grained access control policies using trait-based encryption in collaborative environments [8]. The responsible data owners define designated nodes for collaboration. The users within the same project or group can collaborate. A group is considered as a single entity and it can collaborate with other groups by using the designated nodes defined by the data owner. In this way, unauthorized access can be avoided across groups [10]. Also, there would be a high chance for data duplication when multiple users collaborate. We also propose a novel mechanism to deduplicate the data and maintain single copies of the files present in the cloud environment thereby reducing the cost that is incurred due to duplicate data.

II LITERATURE SURVEY

“Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption,”

In the medical domain, an electronic health record is private sensitive information back will be maintained at the health department it which may be under the control of the patient, this private sensitive information needs to get redirected to another third party, we call it a service provider. There is a security requirement to protect private sensitive electronic patient record which is kept under service through Cloud Service Provider and may be kept available for some unauthorized parties, so we may need to adopt high confidential control access to the patients over their health records by converting the source plain text information into to an appropriate cipher-text based text to an ultimate encryption methodology which includes privacy preservation utmost level. In this article, an innovative patient-driven model along with the set of functional operations h are defined well towards data access control over personal health records in semi-transparent cloud Servers. Along with the above set mechanism we may also need to adopt a key management technique to maintain the secret keys related to all sensitive private information stored in the server and to manage both i.e Data contributors and Data consumers most effectively. In some abnormal situations, we

may need to provide user revocation and need to facilitate a trapdoor mechanism to address certain abnormal scenarios most effectively and efficiently.

“DAC-MACS: Effective data access control for multi-authority cloud storage systems,”

To maintain Data integrity in data as a service cloud environment, data access control has to be well organized in such a way privacy levels are maintained in high grades which we need to address several abnormal issues on cloud data storage systems. In the primitive approaches of controlling sensitive information access over the cloud data stores, women need to take a step to handle an abnormal situation like duplicate copies of similar data will get repeated over completely authorized cloud data servers. So we adopt a highly secured fine-grained, flexible data availability of personal health records by using an advanced methodology like attribute oriented encryption to enhance the security standards in desirable modes. The cipher-text-based attribute-based mechanism facilitates high security over sensitive information as well establishes a restricted access control over unauthorized parties and facilitates an effective consumption of data on to the authorized parties most effectively and efficiently. So the newly constructed system may need to establish itself in a sophisticated manner for data access control power multi-authority cloud data storages and revocation of users over abnormal calls which enhance the security in a backward manner.

“RAAC: Robust and auditable access control with multiple attribute authorities for public cloud storage”

A fine-grained secure and flexible encryption mechanism has to be adopted to deal with data access control over cloud data storages in a most effective way wherein this cipher-text basic trait-based encryption may fulfill the above said requirement. This proposed mechanism works with attribute level authorization and evaluates user identity by using the secret key that got distributed in an earlier state, thus the conclusions made using this cipher-text policy attribute-based scheme performs well and addresses many security issues to generate trustable environments in cloud data storages. In this recommended system authorized people are independent and have a wide scope to organize a variety of attribute keys as well try to eradicate the issue of single point contribution which will lead to establishing an effective access control mechanism with the proper auditing technique so that the user legitimacy can be verified properly.

III SYSTEM ANALYSIS

Existing System:

In the public storage cloud servers, As the data of the data owner is entrusted to third party authorities, there is a high risk of privacy and security breach up data in the cloud

environment. There would be multiple scenarios where the users need to collaborate and share data. the existing trade based encryption mechanisms cannot ensure secure sharing of data in the collaborative environment and cannot handle the duplicate data that gets generated while collaboration. hence there is a need to come up with a novel technique to handle data deduplication and secure sharing in collaborative environments within public clouds.

Disadvantages:

1. Cannot handle data duplication
2. Cannot ensure private sharing of data when multiple users collaborate

Proposed System:

The main objective of this project is to ensure successful collaboration in the public cloud environments and securely share data with the respective individuals or groups. We also aim to handle the data duplication issue that arises when multiple users collaborate and work on the same set of data as they belong to the same enterprise in our organization. Hence we define the access policies based on user roles and sub-roles. these rules are used for setting the traits of the encryption mechanism. To facilitate collaboration between multiple groups and within groups, the data owners of the groups define selected nodes that can interact and share data with the designated nodes of other groups. in this way we can avoid any collusive behavior and securely access the data. Using this process, we can also avoid data duplication by sharing data using the selected nodes.

Advantages:

1. Allows collaboration in a public cloud storage environment
2. Allows data deduplication.

IV IMPLEMENTATION

There are four modules in this project. They are:

1. Data Owner
2. Authority
3. Cloud Server
4. User

Data Owner:

In this module, the data owner is the person who is responsible for applying trait-based encryption on the files and permit users to access the files based on the defined access policies. To perform the relevant activities, the data owner has to first register on the system. Once the data owner registers on the system, the request is sent to the authority to approve. Once the cloud server approves the data owner registration, then the data owner will be allowed to login into the system and upload files and apply trait-based encryption mechanisms to them. He can

also share these files with the relevant users based on access policies.

Authority:

In this module, the authority has the responsibility of generating the content key requests and master secret key requests that are raised by the owners. The authority first logs in into the system and can view the request made by various owners to generate the content keys and master secret keys.

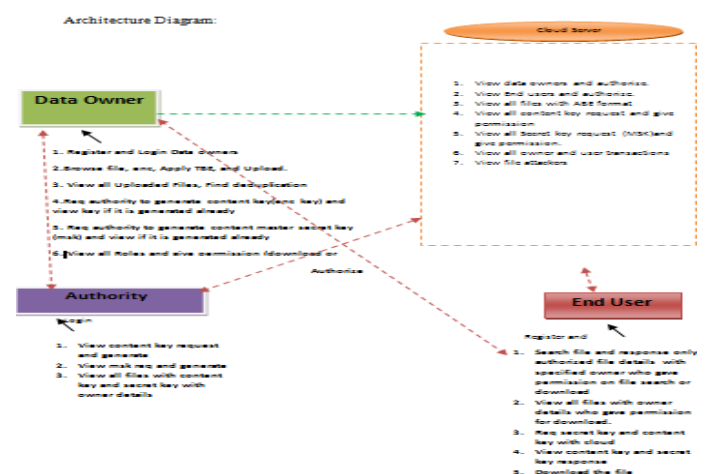
Cloud Server:

The cloud server is the administrator of the public cloud storage system. The responsibilities of the cloud server admin are to authorize the registration of data owners and data users, to view the data of the file that are encrypted using trait-based encryption, view and approve the request for the content key, created by the users, View and approve the request for secret keys created by the users, etc. The cloud server admin cannot view the file as it is in an encrypted format. they can only approve the requests for content keys and secret keys created by the users but they cannot view the key details corresponding to the files. Hence in this way, the data can be protected from an untrusted or semi-trusted third-party cloud service administrator on public clouds.

Users:

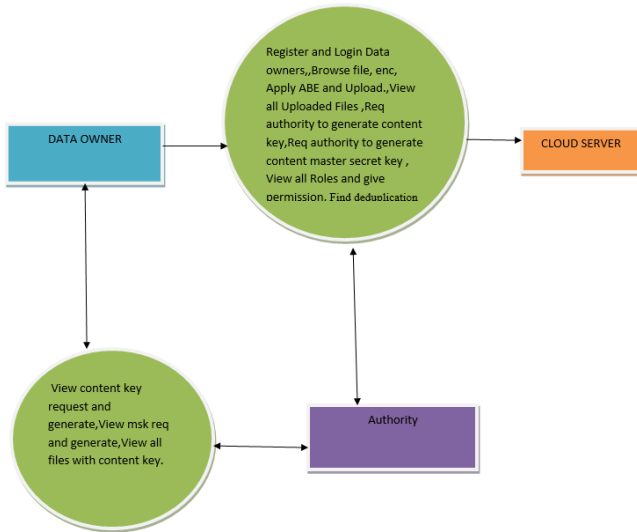
The users are the end-users of the application who wish to view the content that is created by the data owners. These users have to 1st register on the system to be able to view the shared content that is created by the data owners. They can create requests to view certain files that are uploaded by the data owners. Once the data owner permits to view the files, request to approve the content key and the secret key will be created for the cloud administrator to approve. The cloud administrator can then approve the request and these content keys and the secret keys will be shared with the user. The user can make use of the above-mentioned keys to download the files from the public cloud storage.

V SYSTEM DESIGN

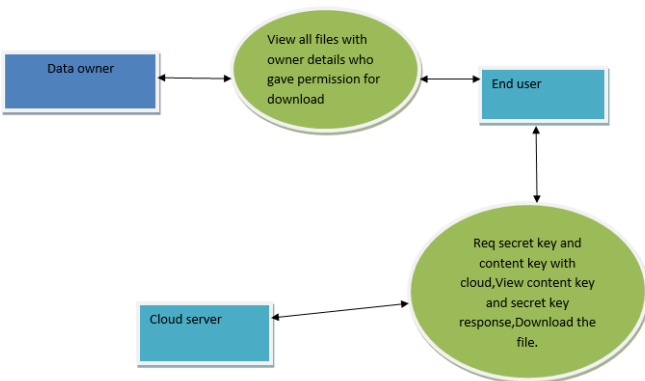


Data Flow Diagram:

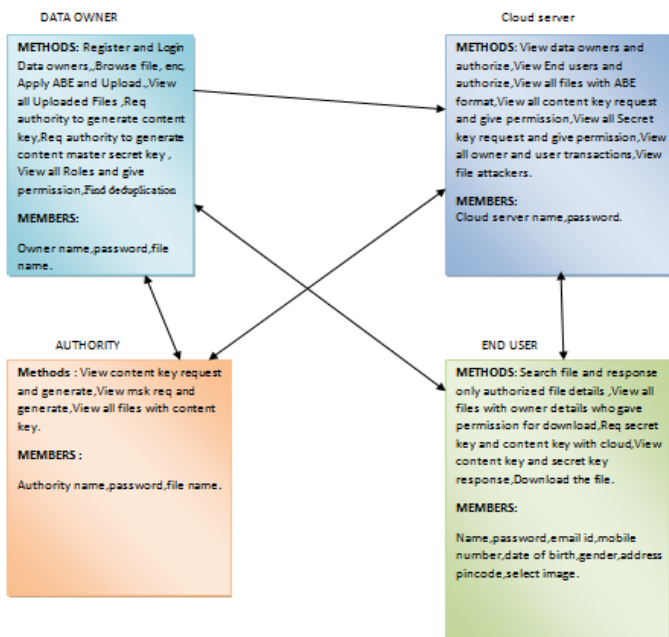
Level -0



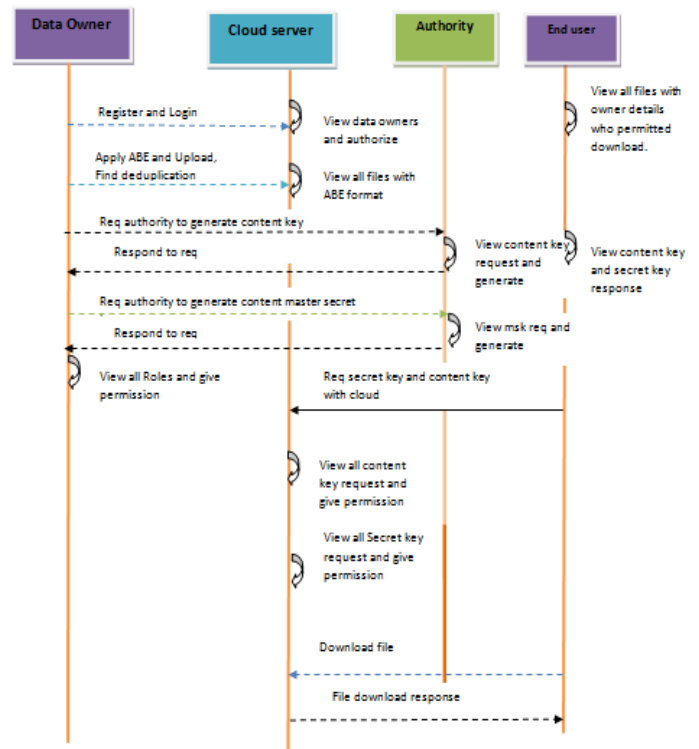
Level -1



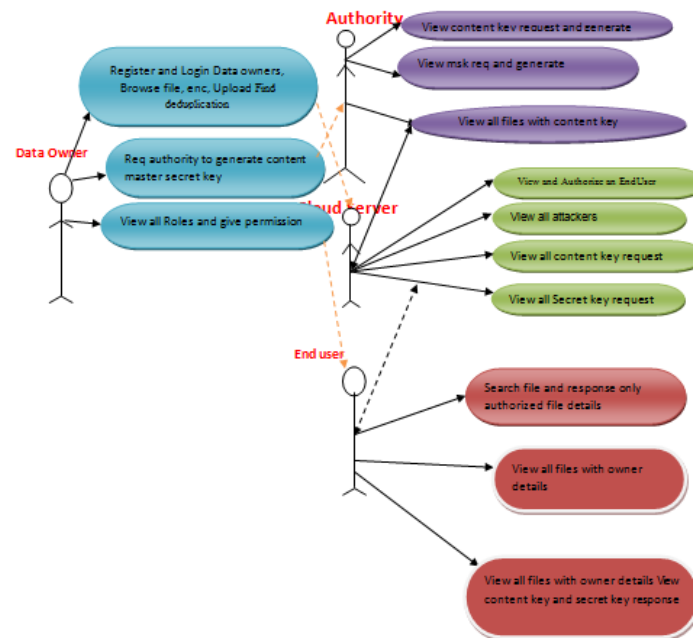
Class Diagram :



Sequence Diagram :



Use Case Diagram:

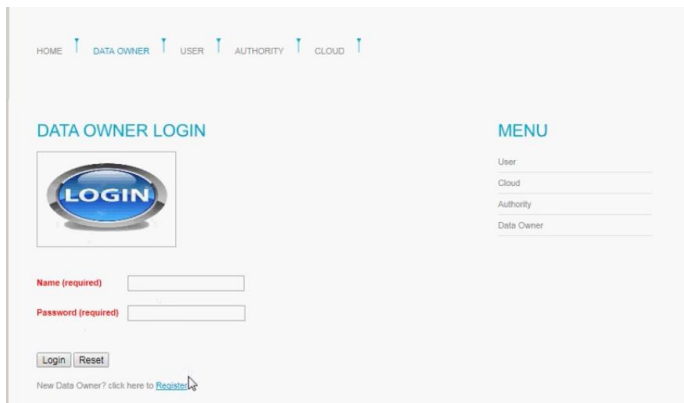


VI PROJECT EXECUTION AND TESTING

Data Owner Login:

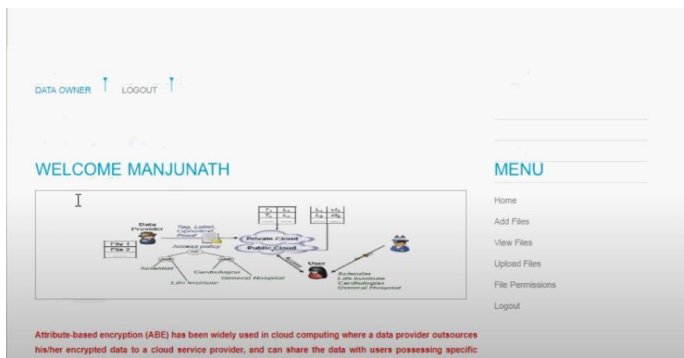
Using the below page the data owner can log in to the system by giving appropriate credentials after the cloud service provider approves the registration request. On successful authentication, the data folder will be redirected to the home

page where he can perform various activities that are entrusted to him.



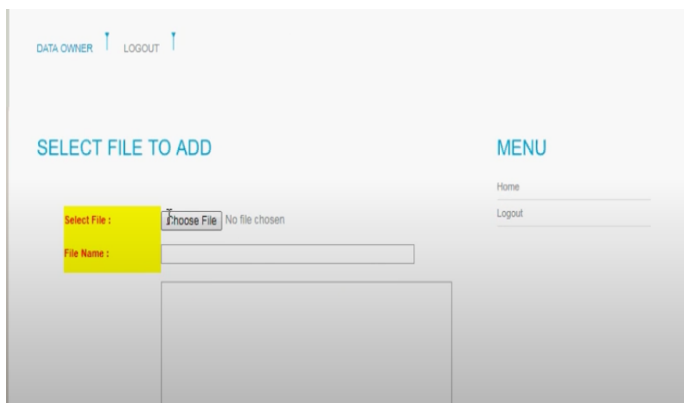
Data owner home:

Using the below page the data owner can perform various activities that are entrusted to him like adding files, viewing files, uploading files, setting file permissions, etc.



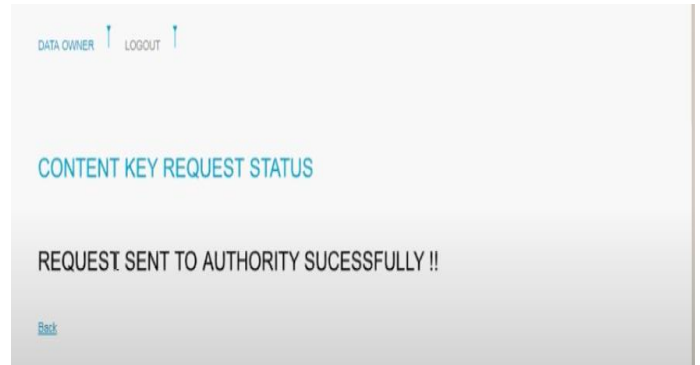
Add Files:

Using the below page the data owner can add the files to the system



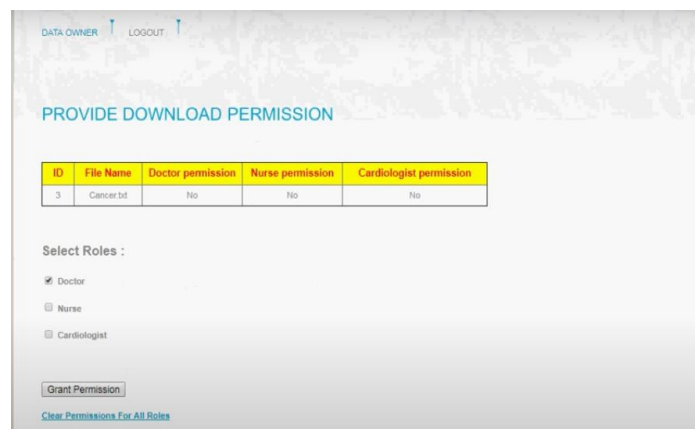
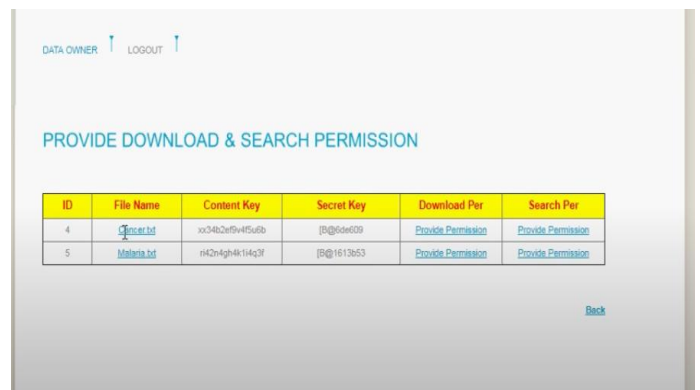
Request keys:

Using the below page, the data owner can request for content key and master secret key. This request would be sent to the authority. The authority will generate the content key and the master secret key for the file and share them with the data owner. Once the data owner receives the keys, the file can be uploaded to the public cloud storage.



Set permissions for files:

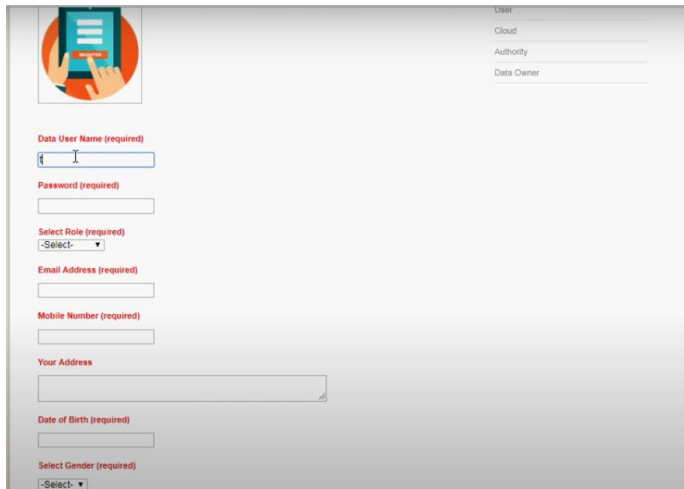
Using the below page the data owner can set sharing permissions to ask for the user roles for each file.



Data user registration:

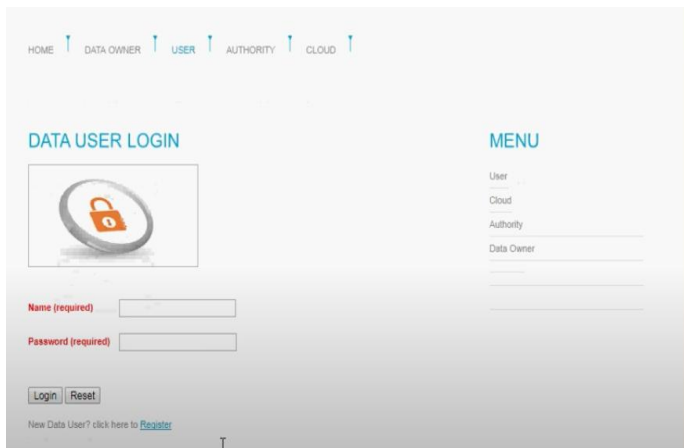
Using the below page, the data user can register on the system. Once the cloud service administrator approves the registration

he can log in to the system and perform various activities that are entrusted to him.



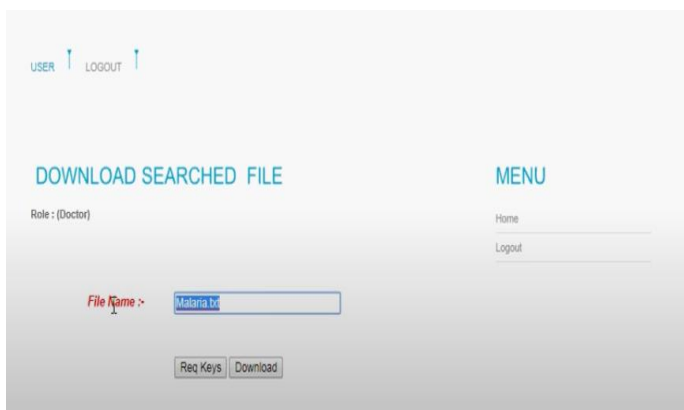
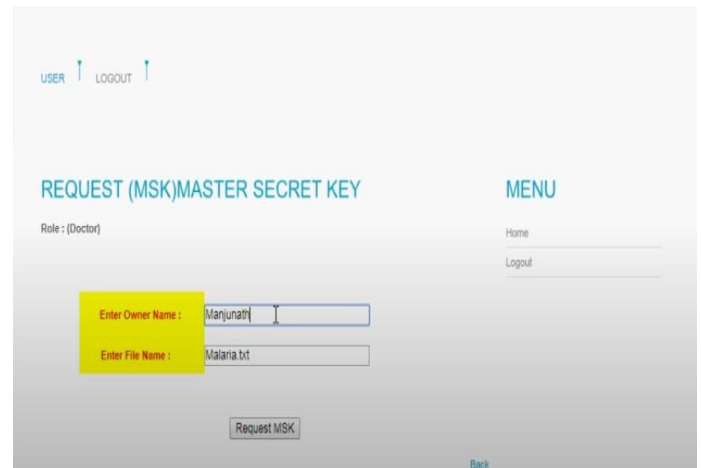
Data user login:

Using the below page the data user can log into the system by giving appropriate credentials on successful login the data user would be redirected to the user home page. otherwise, a message stating that the credentials are invalid would be displayed.



Request keys:

Using the links on the homepage, data user can search for files and request content key and secret key to gain access to the required files as shown below

Download files:

Using the links on the homepage, the data user can download the files after the cloud service provider has approved his request for content keys and secret keys corresponding to that file as shown below. On clicking the download button the file would get decrypted using the keys and the file would be available for download.

VII CONCLUSION

In this project, we Have come up with the trait-based encryption mechanism for collaborative environments in public cloud storage. In this mechanism, the owners chose selected users for data sharing and collaboration. Data owners can also define access policies to satisfy the collaborative requirements end cap and combine the trades in multiple ways to define the access policies. This methodology helps to avoid data deduplication as the data sharing happens through few selected nodes. the content keys and secret keys are generated by a third-party authority who is fully trusted, and they are shared with the data owners. Thus, collaboration and data deduplication are both achieved using this process in public cloud environments. The security and performance results are promising, and this process can be implemented in real-time public cloud storage systems.

Future Enhancement:

We would like to best this process in larger-scale environments and automate the generation of content and secret keys without the intervention of the third-party authority.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica et al., "A view of cloud computing," Communications of the ACM, vol. 53, no. 4, pp. 50–58, 2010.

[2] K. Yang, X. Jia, K. Ren, and B. Zhang, "DAC-MACS: Effective data access control for multi-authority cloud storage systems," in Proceedings of the 32nd IEEE International

Conference on Computer Communications (INFOCOM). IEEE, 2013, pp. 2895–2903.

[3] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, “Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 131–143, 2013.

[4] Y. Wu, Z. Wei, and H. Deng, “Attribute-based access to scalable media in cloud-assisted content sharing,” *IEEE Transactions on Multimedia*, vol. 15, no. 4, pp. 778–788, 2013.

[5] K. Xue, Y. Xue, J. Hong, W. Li, H. Yue, D. S. Wei, and P. Hong, “RAAC: Robust and auditable access control with multiple attribute authorities for public cloud storage,” *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 953–967, 2017.

[6] W. Li, K. Xue, Y. Xue, and J. Hong, “TMACS: A robust and verifiable threshold multi-authority access control system in public cloud storage,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 5, pp. 1484–1496, 2016.

[7] K. Xue, W. Chen, W. Li, J. Hong, and P. Hong, “Combining data owner-side and cloud-side access control for encrypted cloud storage,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 8, pp. 2062–2074, 2018.

[8] A. Shamir, “How to share a secret,” *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[9] T. Tassa, “Hierarchical threshold secret sharing,” *Journal of Cryptology*, vol. 20, no. 2, pp. 237–264, 2007.

[10] M. Li, X. Huang, J. K. Liu, and L. Xu, “GO-ABE: group-oriented attribute-based encryption,” in *Proceedings of the 8th International Conference on Network and System Security (NSS)*. Springer, 2014, pp. 260–270.