DUAL ACCESS CONTROL AND SHARING CONTROL

Shubham Upadhyay, Katakam Kiran Syam Kumar, Konduru Sriram Tarun Gupta, Korikana Manikanta, Vankayala Sri Harshit

Guide, Student, Student, Student, Student

Computer Science and Engineering,
Parul University , Vadodara , INDIA

shubhamupdhyay09@gmail.com , kirankatakam25@gmail.com , sriramkonduru0@gmail.com , manikanta76414@gmail.com , harshitvankayala118@gmail.com

ABSTRACT

Cloud-based data storage has recently gained significant attention in both academic and industrial domains due to its efficient and cost-effective management. Since these services are offered over open networks, it becomes essential for service providers to ensure secure mechanisms for data storage and sharing in order to safeguard user privacy and maintain data confidentiality. Encryption is the most widely used method to prevent unauthorized access to sensitive information. However, encryption alone is not sufficient to fully address the practical needs of data management. Additionally, it is important to implement robust access control over download requests to defend against Economic Denial of Sustainability (EDoS) attacks, which can hinder legitimate users from accessing the service. To address this, dual access control systems have been proposed, each designed for specific application scenarios. We introduce an approach that regulates download requests while maintaining both security and efficiency. Furthermore, experimental evaluations and security assessments of the proposed systems are also discussed.

Keywords: Cloud-based data sharing, Access Control, Cloud Storage Service, Attribute Based Encryption.

Introduction:

In recent years, cloud-based storage has gained considerable attention, with many businesses opting to outsource their data to remote cloud platforms rather than investing in upgrades to their local infrastructure or hardware.

To tackle the challenges mentioned above, we propose a new approach called **dual access control**. One promising technique for securing data in cloud storage services is **Advanced Encryption Standard (AES)**,which ensures Uses the same key for encryption and decryption. And **decryption Rivest-Shamir-Adleman (RSA)**, Uses separate public and private keys for encryption and decryption, thereby determining which users are permitted to access the data.

It is important to note that this study focuses on the integration of AES and RSA into our proposed framework. However, relying solely on AES and RSA is not sufficient to design an advanced system that can effectively manage both data access and control of download requests.

Objective:

The primary goal of this project is to offer uploaded data. Data owners alone will be in charge of dual access control; no outside parties are involved. They will obtain the data that was uploaded to the cloud at the request of datausers Users with the assistance of data owners. The cloud service encrypts the data and distributes the key to the data consumers.

Scope of the Project:

The project's primary objective is to provide a privacy-enabled and secure cloud project that will be used by AMAZON WEB SERVICES (AWS).
Services include preventing Denial of Sustainability assaults and providing privacy, encryption, and encryption.

1. Key management and encryption
2. Portability and interoperability
3. Management of Identity, Entitlement, and Access
4. The architectural framework for cloud computing.
5. Security as a service

Existing System:

The current method compromises security and privacy by utilizing standard servers to store and share data. There is a danger that our data will be stolen. This is the primary flaw in the current method, thus to get around it, we may use the suggested solution.

Drawbacks :

- It makes extensive use of arithmetic.
- Reduced security.
- Analysis of the saved document is challenging.
- The length of time it takes to search through the database of saved documents is linear.

Literature Review:

For flexible data exchange, attribute-based encryption and searchable encryption are combined.

Safe cloud storage is regarded as one of the most critical problems that both enterprises and end users must address before transferring their sensitive data to the cloud. Recently, we've seen several intriguing techniques based on either the promising notion of Symmetric Searchable Encryption (SSE) or the wellstudied topic of Attribute-Based Encryption (ABE). In the first scenario, researchers are attempting to build protocols that would safeguard users' data from both internal and external threats while ignoring the issue of user revocation. In the second scenario, however, current alternatives handle the issue of revocation.

The overall efficiency of these systems is jeopardized, however, because the suggested protocols are purely based on ABE schemes, and the quantity of the created ciphertexts and the time required to decrypt rises in direct proportion to the complexity of the access formula. In this research, we present a protocol that combines SSE and ABE while using the fundamental benefits of each approach. The proposed protocol allows users to search directly over encrypted data using an SSE method, while the matching symmetric key required for decryption is safeguarded using a CiphertextPolicy Attribute-Based Encryption scheme. and searchable encryption to allow for more flexible data exchange.

Problem Statement:

As the key management server only contains the document metadata in encrypted format and the application server will have encrypted documents, a cloud administrator won't be able to decrypt any documents. The papers will remain secure as a consequence. For dual access control, we proposed an identity key verification technique. The goal of this project is to provide a Secured and Privacy-Enabled The privacy, encryption, and decryption services provided by cloud projects prevent Deny of Sustainability assaults.

Proposed System:

In the proposed system, we suggest a novel method in the proposed system called dual access control. One of the potential options for securing data in cloud-based storage services is attribute-based encryption, which allows for both fine-grained management and the secrecy of outsourced data.

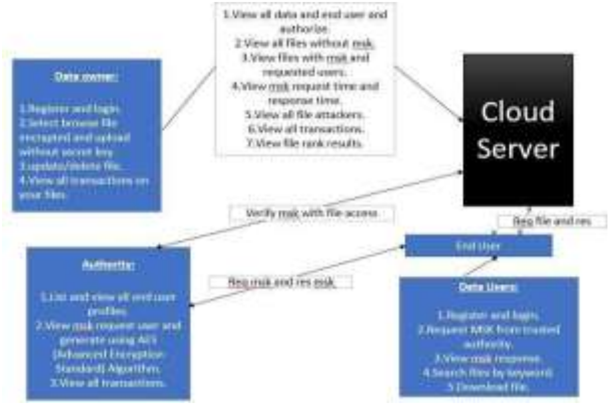
Process:

STEP-1: Install the necessary applications STEP-2: Defining the links to the databases. STEP-3: Establish all of the database tables needed for this project.
STEP-4: Redesign the HTML and CSS pages.
STEP-5: Construct the project using the modules.
STEP-6: Launch the Python program (app.py),

copy the link, paste it in any browser, and then proceed as directed.

Architecture:

Our dual access control system topologies for cloud data sharing are displayed. The systems specifically include the following entities.


Data owner:
1.Register and login.
2.Select browse file encrypted and upload without secret key.
3.update/delete file.
4.View all transactions on your files.
1.View all data and end user and authorize.
2.View all files without msk.
3.View files with msk and requested users.
4.View msk request time and response time.
5.View all file attackers.
6.View all transactions.
7.View file rank results.
Cloud Server
Verify msk with file access
Req file and res
End User
Authority:
1.List and view all end user profiles.
2.View msk request user and generate using AES (Advanced Encryption Standard) Algorithm.
3.View all transactions.
Req msk and res msk
Data Users:
1.Register and login.
2.Request MSK from trusted authority.
3.View msk response.
4.Search files by keyword.
5.Download file.

- Initializing system parameters and data user registration are the responsibility of the authority. Also, the initial suggested construction deals with the cloud call request.

- Data owner wants to outsource his data to the cloud and currently retains the data. Particularly, data owners only like to disclose their data with those who meet specific requirements (e.g., professors or associate professors). After their data has been transferred to the cloud, they will be offline.

- A user of data wishes to download and decode encrypted data that has been shared in the cloud. The encrypted file may be downloaded by those with permission, who can then decode it to view the plaintext.

- Both data owners and consumers may save their data easily in the cloud. In particular, it manages the download requests made by data users and maintains the data users' outsourced data.

- The cloud's call request is handled by Enclave.

Module Description:

1. **Data Owner :**

Register:
Data owner can Register and login with valid credentials

Upload File:
Data provider can upload the file.

View File:
Data Owner can view uploaded file once means whether the file is correctly uploaded or not.

2. **Data User : Register:**

Data user can do registration with his details.

Login:
The user needs to register and the data stored in MySQL database.

Search a File:
Data user can search a file based on the keyword ,if file is available then user can view file and send request to cloud to download the file.

Get Key & Download
Once User Request can accept get the key to cloud provider user can download the file.

3. **Cloud Provider:**

Login
Cloud provider can login with his/her credentials.

View Files:
Cloud can view all uploaded files.

View Users:
Cloud can view all the users details to give permission for login the website.

View Data Providers:
Cloud can view all the data providers details to give permission for login the website.

Send Key request to Authority:
Cloud gets a key from authority and send to the authority.

4. **Authority:**

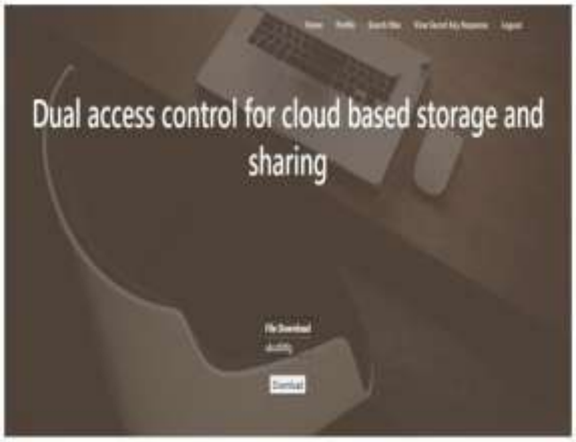
Login:
Authority login and view users and give authorization to users.

Generate key to users:
Authority generate key to users.

Methodology:

- Collecting and analysis of needs – at this stage, all potential system requirements are gathered and recorded in a requirement specification document.

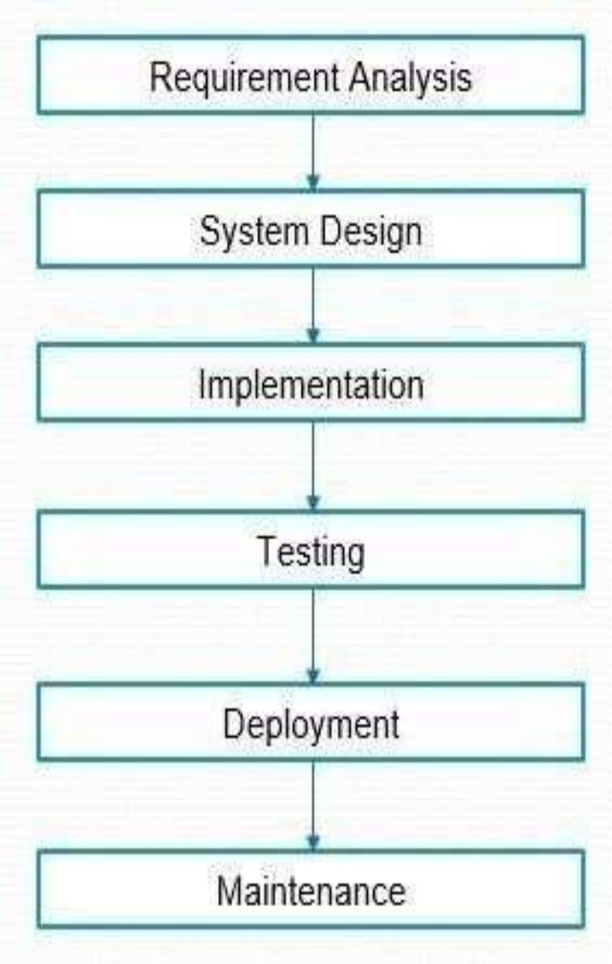
- System Design – In this step, the required specifications from the previous phase are examined, and the system design is created.



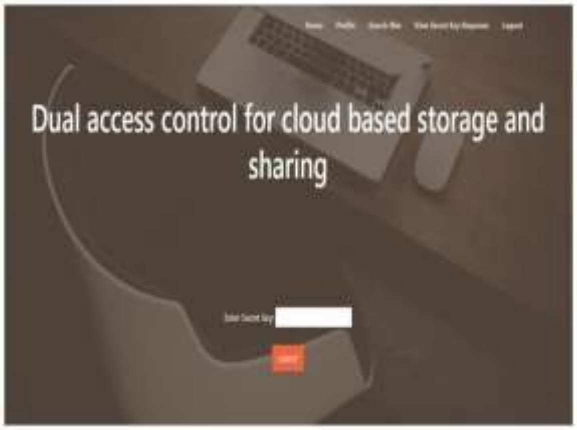
Dual access control for cloud based storage and sharing

In addition to describing the overall system architecture, this system design aids in identifying the hardware and system requirements.

- Implementation: Using feedback from the system design, the system is initially created as a series of compact programmes known as units. These units are then merged in the next step. Unit testing is the process of developing and evaluating each unit for functionality.
- Integration and Testing – Following the testing of each unit created during the implementation phase, the entire system is integrated. The entire system is tested for errors and failures after integration.
- System deployment – After functional and nonfunctional testing, the product is either provided to customers or deployed in their environments.
- Maintenance – The client environment occasionally experiences problems. Patches are published to address certain problems. Moreover, various improved versions of the product have been launched. To bring about these changes in the surroundings of the consumer, maintenance is performed.

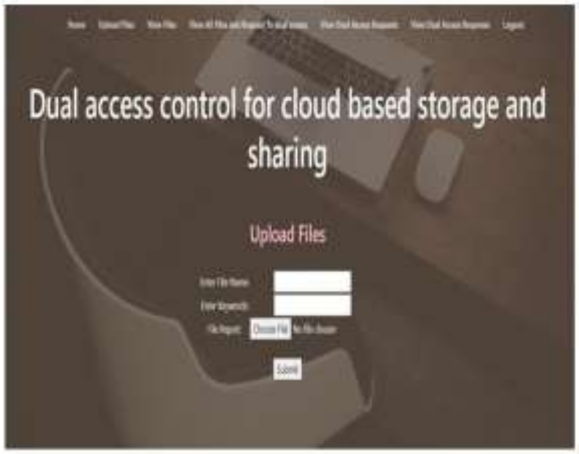


Result :



Dual access control for cloud based storage and sharing

Implementation :



Dual access control for cloud based storage and sharing

Upload Files

This explains how the system functions. When we have loaded all the libraries, the data owners will upload and store the data to the cloud. Afterwards, people who require the data will submit requests to the data owners. The data owners will then see the request.

Requirement specifications:

- Operating System : Windows 7/8/9
- IDE : Pycharm
- Server side scripts : HTML, CSS, JS
- Libraries Used : Numpy, IO, OS, sklearn, Flask
- Technology : Python

Conclusion:

We presented two dual access control mechanisms and tackled a critical and widespread challenge in cloud-based data sharing. The proposed systems are resistant to DDoS/EDoS attacks. Our approach demonstrates that various CP-ABE constructions can be adapted to incorporate the functionality of download request control. Experimental results further indicate that the proposed methods introduce only minimal computational and communication overhead compared to their underlying CP-ABE framework.

In our enhanced system, we rely on the property that sensitive information within the enclave cannot be retrieved. However, potential risks remain, as memory access patterns and other side-channel attacks may allow a malicious host to extract partial secrets from the enclave. To address this, we consider the **transparent enclave execution model**. A compelling open problem lies in designing a dual access control scheme for cloud data sharing within the context of a transparent enclave. We plan to explore effective solutions to this challenge in our future work.

Future Scope:

The Future scope of this project is to increase the security and get the data easily by email authentication.

References:

1. A. Bakas and A. Michalas, "Modern family: A revocable hybrid encryption scheme based on attribute-based encryption symmetric searchable encryption and SGX", *Proc. Int.* Conf. Secur. Privacy Commun. Syst., pp. 472*486, 2019.*
2. J. Li, Y. Wang, Y. Zhang and J. Han, "Full verifiability for outsourced decryption in attribute based encryption", *IEEE Trans. Services Comput.*, vol. 13, no. 3, pp. 478-487, May/Jun. 2020.
3. A. Michalas, "The lord of the shares: Combining attribute-based encryption and searchable encryption for flexible data sharing", Proc. 34th ACM/SIGAPP Symp. Appl. Comput., pp. 146155, 2019.
4. J. Ning, Z. Cao, X. Dong, K. Liang, H. Ma and L. Wei, " Auditable \$sigma\$ σ -time outsourced attribute-based encryption for access control in cloudcomputing ", *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 1, pp. 94-105, Jan. 2021.
5. F. Tramer, F. Zhang, H. Lin, J.-P. Hubaux, A. Juels and E. Shi, "Sealed-glass proofs: Using transparent enclaves to prove and sell knowledge", *Proc. IEEE Eur. Symp. Security Privacy*, pp. 19-34, 2020.
6. J. Han, W. Susilo, Y. Mu, J. Zhou and M. H. A. Au, "Improving privacy and security in decentralized ciphertext-policy attributebased encryption", *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 3, pp. 665678, Mar. 2015.
7. J. Ning, Z. Cao, X. Dong and L. Wei, "White-box traceable CP-ABE for cloud storage service: How to catch people leaking their access credentials effectively", *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 5, pp. 883-897, Sep./Oct. 2019.
8. V. Costan and S. Devadas, "Intel SGX explained",IACR Cryptol. ePrint Archive, vol. 2016, no. 086,pp. 1-118, 2016.
9. 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 Trans. Inf. Forensics Security, vol. 13, no. 8, pp. 2062-2074, Aug. 2019.
10. W. Susilo, P. Jiang, F. Guo, G. Yang, Y. Yu and Y. Mu, "EACSIP: Extendable access control system with integrity protection for enhancing collaboration in the cloud", IEEE Trans. Inf. Forensics Security, vol. 12, no.12, pp. 3110-3122, Dec. 2020.