International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 04 | April - 2023 Impact Factor: 8.176 ISSN: 2582-3930

SECURE DATA SHARING IN AWS CLOUD ENVIRONMENT

Dr. M. Saraswathi
Assistant Professor,

Nettem Hemanth, Moola Sai Muralidhar Reddy
B. E, 4th Year,

Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya
Enathur, Kanchipuram

ABSTRACT:

The main aim of this paper is to provide secure data sharing to authenticated users by the data owners by checking authorization. As it is effective and low-cost management, cloud-based data storage services have the interest in both academia and industry in recent years. Service providers must utilize secure data storage and sharing mechanisms to maintain data confidentiality and service user privacy because they deliver service through an open network. Encryption is the most widely used for protecting sensitive data from being hacked. However, simply encrypting data (for example, using AES) is insufficient to meet the actual need for data management. Also, effective access control over download requests must be considered to prevent Economic Denial of Service (EDoS) attacks from preventing users from using the service. In this work, we look at secure cloud-based storage, in the sense that we create a control mechanism that can handle both data access and download requests while holding security and efficiency using an attribute-based encryption algorithm.

Keywords: Cloud, Economic Denial of Sustainability (EDOS), Advanced encrypted standard(AES), Denial of Service(DOS).

I. INTRODUCTION:

Both academia and industry have been paying attention to cloud-based storage systems in current decades. It might be widely employed in a variety of Internet-based saleable applications (e.g., Apple iCould) due to a large number of advantages, including access flexibility and the elimination of local data administration. A growing number of people and businesses are opting to outsource their data to the cloud to save money on reworking their local data management facilities and devices. However, one of the biggest obstacles preventing internet clients from using cloud-based storage services generally is the fear of data security breakings. In many practical examples, outsourced data may need to be shared with others. Alice, for example, may share photos with her Dropbox friends. Without employing data encryption, Alice must create a sharing link and then share it with friends before sharing the images. The sharing link may be disclosed within the Dropbox management level, albeit providing access restrictions over unauthorized users (e.g., those who are not Alice's friends) (e.g., the administrator could reach the link). Because the cloud (which is built on an open network) cannot be trusted, it is generally suggested that data be encrypted before being uploaded to the cloud. One similar option is to encrypt the outsourced data before uploading it to the cloud using encryption technology (e.g., AES) so that only a single cloud user (with a valid decryption key) can decrypt the data. To prevent "insiders" of the system from accessing shared photos, a simple solution is to select a group of approved data users before encrypting the data. However, Alice may have no idea who the photo recipients/users will be in some cases. It's possible that Alice just knows about the picture receiver's qualities. Traditional public key encryption (e.g., Paillier Encryption) cannot be used in this circumstance because it requires the encryptor to know who the data receiver is ahead of time.

II.LITERATURE SURVEY:

Secure cloud storage is one of the most important concerns for both businesses and end users when moving their private data to the cloud. Recently, various new approaches have appeared, based on either the good concept of Symmetric Searchable Encryption (SSE) or the well-studied topic of Attribute-Based Encryption (ABE). In this research, we present a mixed encryption system that uses the advantages of both SSE and ABE to combine the two algorithms. Unlike many other techniques, we create a cancellation process that is fully independent of the ABE scheme and purely based on SGX functionality. [1].

Searchable Encryption (SE) has been much investigated by academic and commercial researchers. While many academic SE algorithms have proven security, to achieve great implementation, they frequently leak some questioning information (e.g., search and access patterns). Several assumption attacks, such as a query recovery attack that uses previous information to convert unclear query trapdoors to their corresponding keywords, have taken advantage of this leakage. Many proposed SE methods, on the other hand, need an extensive change of current applications, making them less possible, usable, and deployable. In this paper, we present IDCrypt, a secure and practical searchable symmetric encryption system for cloud applications that have been proven to improve search efficiency and maintain the security of SE using symmetric cryptography. We also go into the challenges of exploring safely across several indexes and providing encrypted data to multiple users. To solve the previous challenges, we propose a token-adjustment search method to sustain search functionality among multi-indexes, as well as a key-sharing scheme that combines identity-based encryption with public-key encryption. According to our findings, the overhead of the key-sharing mechanism is somewhat low[2].

People see the value of cloud computing, but they are reluctant to trust cloud providers with sensitive data because of the lack of user-to-cloud control over it. To maintain obscurity, data owners outsource encrypted data rather than plaintexts. When sharing encrypted files with multiple users, ciphertext-policy attribute-based encryption (CP-ABE) can be utilized to provide fine-grained and owner-centered access control. This, however, does not make you resistant to other forms of attacks. Many last systems did not allow the cloud provider to check whether a downloader could decrypt the data. As a result, these data should be accessible to anyone with access to cloud storage. A negative attacker can download thousands of files to perform financial rejection of service (EDoS) attacks, consuming significant cloud resources. The cost is delivered by the cloud service payer. Likewise, the cloud provider serves as both the accountant and the payee of resource consumption fees, leaving data owners in the dark. These issues should be addressed in public cloud storage in the real world. In this research, we present a method for protecting encrypted cloud storage from DoS assaults while also securing resource consumption clarity. It uses CP-ABE techniques in a black-box way and adheres to the CP-arbitrary ABE's access policy. Two protocols are shown for various situations, followed by performance and security analysis[3].

3. PROBLEM STATEMENT:

The use of cloud-based data storage services has seen a rise in favor among both academics and industries in recent years due to their price-efficient and effective management. However, as these services are provided through an open network, service providers must implement secure data storage and sharing techniques to preserve user privacy and maintain data confidentiality. Encryption is the most commonly used method to protect liable information. Despite its widespread use, just depending on encryption may not fully address the practical challenges of managing data.

4. PROPOSED SYSTEM:

To overcome these challenges, we used an dual access control in our system. A developed encrypted model Advanced Encryption Standard(AES) has been identified as a possible solution to enhance the data securely stored in cloud-based services. This encryption technique provides not only confidentiality for outsourced data but also a flexible means of managing the outsourced data with adequate control.

Advantages of Proposed System:

i. Confidentiality and Integrity
ii. Data sharing
iii. Accessible to only authorized users
iv. Data encrypted

SYSTEM ARCHITECTURE:

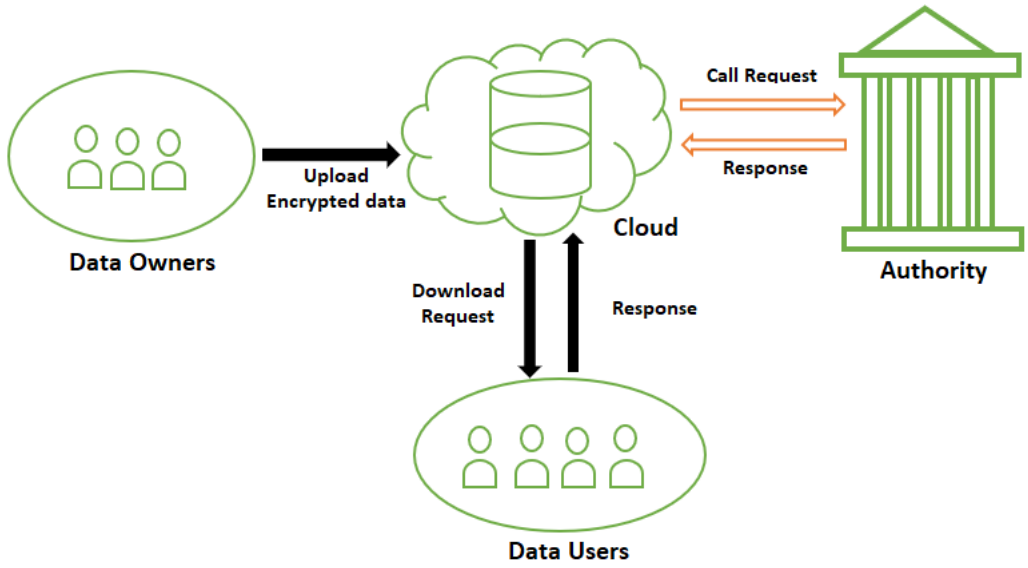

Fig 1. System Architecture

The above figure Represents that Data owners are the persons who upload files in the Cloud. Data users those who request the file. The Cloud is embedded with an Entity called Enclave & also the Authority. The Enclave or Authority will be checking the Access Policy.

4.1ALGORITHM USED:

AES

Round keys are a set of specially constructed keys used in the encryption process. These, along with other processes, are applied to an array of data that contains exactly one block of data to be encrypted. This array is directed to the state array.

For a 128-bit block, you perform the following aes encryption steps:

From the cipher key, create a series of round keys.

Use the block data to settle the state array (plaintext).

To the starting state array, add the first round key.

Nine rounds of state manipulation are required.

Complete the tenth and final state manipulation round.

Make a copy of the final state array as encrypted data (ciphertext).

Because the tenth round includes a somewhat different manipulation than the others, the rounds are given as "nine followed by a final tenth round."

The encrypted block is simply a 128-bit sequence. Because AES operates with byte values, we must first change 128 bits to 16 bytes. We say "convert," but it's almost definitely already saved in this format. RSN/AES uses a two-dimensional byte array with four rows and four columns for operations. The 16 bytes of data at the start of the encryption.

6.1 PROJECT DESCRIPTION:

a) DATA OWNER:

Data owners can create an account and log in using proper credentials. They can upload files to the platform. After uploading a file, the data owner can check it once to confirm that it was correctly uploaded.

b) USER:

Data users are required to register on the platform using their personal information, and this information is stored in a MySQL database. Once registered, a data user can search for a file using a keyword. If the file is found, the user can view it and send a download request to the cloud provider. After receiving the required key from the cloud provider, the user can download the file.

c) CLOUD PROVIDER:

The cloud provider can log in using their valid credentials. All uploaded files are visible to the cloud provider. The cloud provider has access to all of the users' information to grant access to the website. Similarly, the cloud provider has access to all of the data providers' information to give them access to the website. The cloud provider gets a key from the management and sends it to the appropriate party.

d) AUTHORITY:

As an authority, you can log in to view registered users and grant them permission to access certain resources. You can also generate keys that allow users to access these resources.

Data sharing can be achieved in two ways:

Data owner to data owner rights

and Data owner to data user rights

V.RESULTS:

The result of the proposed system is represented in fig 2 to fig 6.

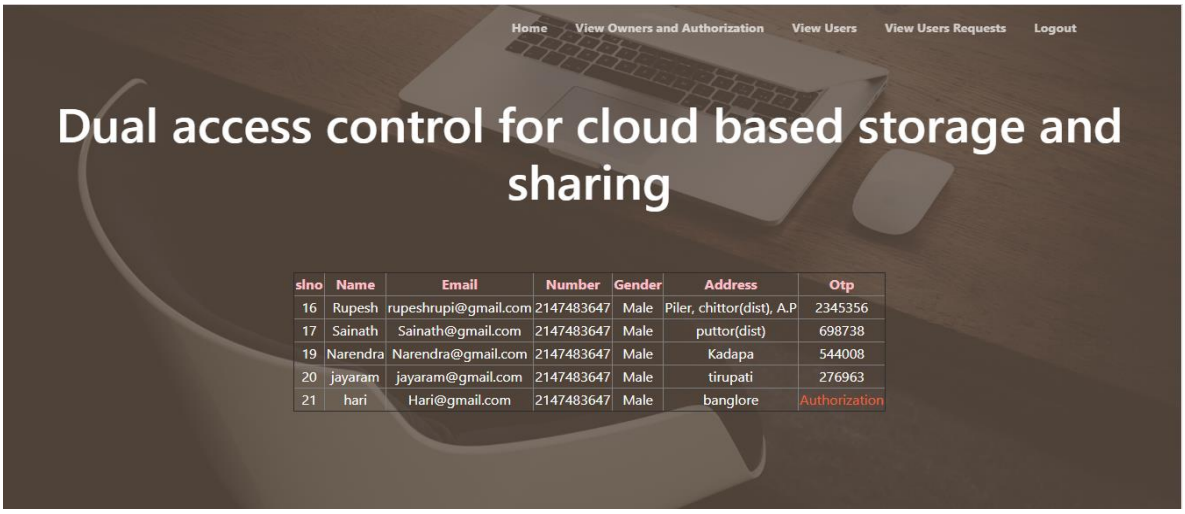

Fig 2. Owner's Information view

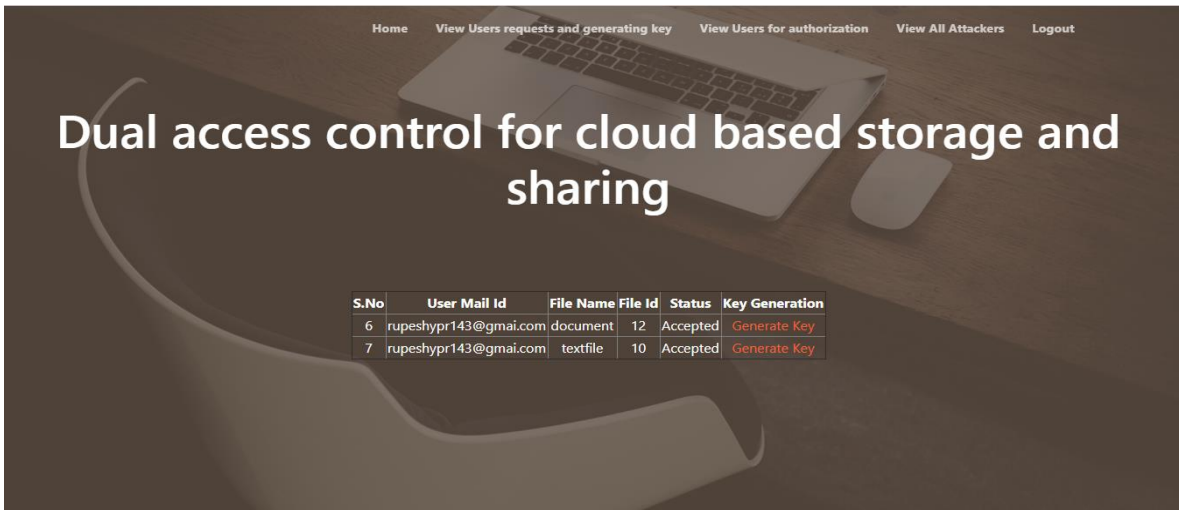

Fig 3. View User's request and Generate key

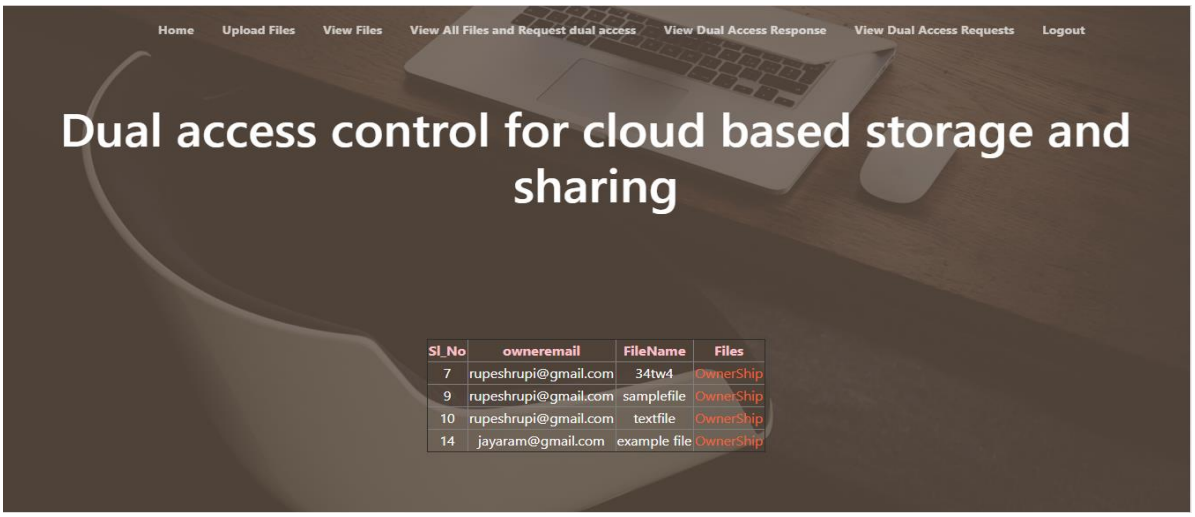

Fig 4. View All Files and request dual access

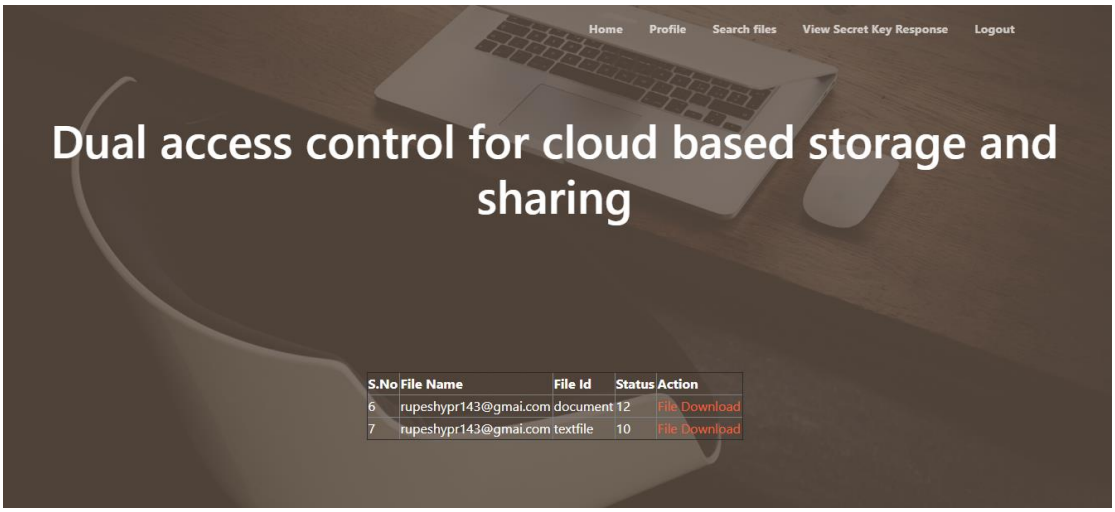

Fig 5. View Secret key response

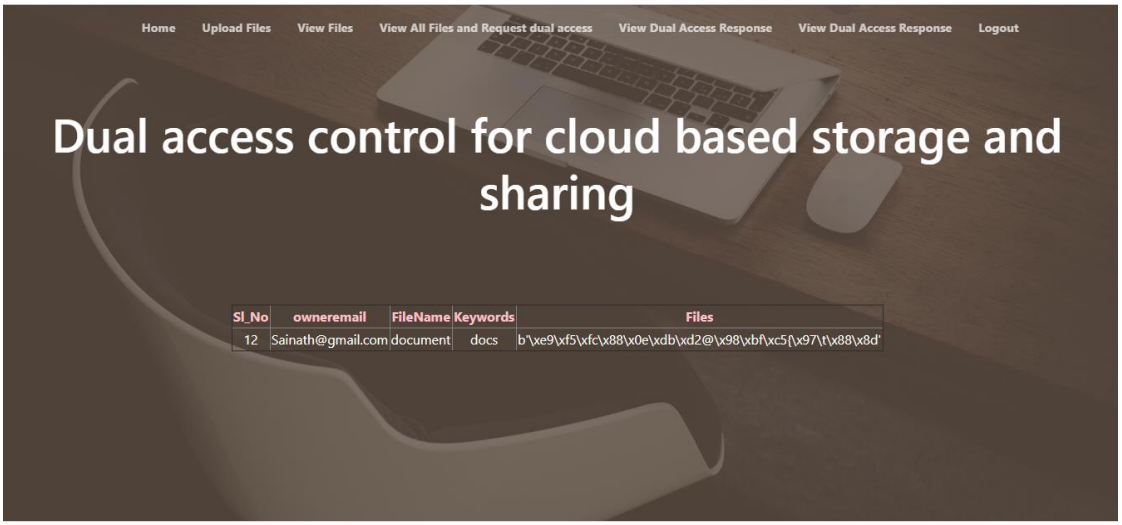

Fig 6. View Files

8. CONCLUSION:

In this paper, we present two novel dual-access control solutions to address a significant issue in cloud-based data sharing. Our proposed system Advanced Encryption Standard (AES) is resilient to DDoS and EDoS attacks. It is a symmetric encryption algorithm that is widely used to secure electronic data. AES uses a block cipher, which means it encrypts data in fixed-size blocks of plaintext, typically 128 bits long. It also employs a symmetric key, which means the same key is used for both encryption and decryption. AES uses a substitution-permutation network to carry out the encryption process, which involves substituting values in the plaintext with values from a lookup table, and then rearranging the resulting values using a permutation function. The method utilized to implement control on download requests is highly adaptable to various CP-ABE designs. Our upgraded system leverages the fact that private information entered into the enclave cannot be recovered. Recent research has raised concerns over the possibility of secret leakage from an enclave to a malicious host through memory access patterns or other side channels. To address this, we introduce the transparent enclave execution model. The development of a dual access control system from a transparent enclave for cloud data sharing is an exciting topic for future work.

9. REFERENCES:

[1] Jianting Ning, Xinyi Huang, Willy Susilo, Senior Member, IEEE, Kaitai Liang, Member, IEEE, Ximeng Liu Member, IEEE, and Yinghui Zhang, Member, IEEE, “Dual Access Control for Cloud-Based Data Storage and Sharing”, IEEE Transactions on Dependable and Secure Computing (Early Access), 2021.

[2] Ittai Anati, Shay Gueron, Simon Johnson, and Vincent Scarlata. Innovative technology for cpu based attestation and sealing. In Workshop on hardware and architectural support for security and privacy (HASP), volume 13, page 7. ACM New York, NY, USA, 2020.

[3] 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.

[4] Amos Beimel. Secure schemes for secret sharing and key distribution. PhD thesis, PhD thesis, Israel Institute of Technology, Technion, Haifa, Israel, 2018.

[5] JohnBethencourt,AmitSahai,andBrentWaters. Ciphertext-policy attribute-based encryption. In S&P 2007, pages 321–334. IEEE, 2018.

[6] Victor Costan and Srinivas Devadas. Intel sgx explained. IACR Cryptology ePrint Archive, 2016(086):1–118, 2016.

[7] Ben Fisch, Dhinakaran Vinayagamurthy, Dan Boneh, and Sergey Gorbunov. IRON: functional encryption using intel SGX. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, pages 765–782, 2017.

[8] Alexandros Bakas and Antonis Michalas. Modern family: A revocable hybrid encryption scheme based on attribute-based encryption, symmetric searchable encryption and SGX. In SecureComm 2019, pages 472–486, 2019.

[9] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In ACM CCS 2006, pages 89–98. ACM, 2011.

[10] Jinguang Han, Willy Susilo, Yi Mu, Jianying Zhou, and Man Ho Allen Au. Improving privacy and security in decentralized ciphertext-policy attribute-based encryption. IEEE transactions on information forensics and security, 10(3):665–678, 2015.

[11] Christofer Hoff. Cloud computing security: From ddos (distributed denial of service) to edos (economic denial of sustainability). http://www. rationalsurvivability.com/blog/?p=66.

[12] Joseph Idziorek, Mark Tannian, and Doug Jacobson. Attribution of fraudulent resource consumption in the cloud. In IEEE CLOUD 2012, pages 99–106. IEEE, 2012.

[13] Simon Johnson, Vinnie Scarlata, Carlos Rozas, Ernie Brickell, and Frank Mckeen. Intel Rsoftware guard extensions: Epid provision-ing and attestation services. White Paper, 1:1–10, 2016.

[14] Sangho Lee, Ming-Wei Shih, Prasun Gera, Taesoo Kim, Hyesoon Kim, and Marcus Peinado. Inferring fine-grained control flow inside sgx enclaves with branch shadowing. In 26th USENIX Security Symposium, USENIX Security, pages 16–18, 2017.

[15] Jiguo Li, Xiaonan Lin, Yichen Zhang, and Jinguang Han. Ksfoabe: outsourced attribute-based encryption with keyword search functionforcloudstorage. IEEETransactionsonServicesComputing, 10(5):715–725, 2017.