International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT228458

Cloud Based Secure File Sharing Using Access Control

Jay Patel*, Akshit Trivedi
Department of Information Technology, Birla Vishvakarma Mahavidyalaya, Anand, Gujarat, India

Article Info

Publication Issue :
Volume 8, Issue 4
July-August-2022

Page Number : 337-348

Article History
Accepted: 10 August 2022
Published: 30 August 2022

ABSTRACT

In the current environment, data security is paramount, and any confidential files we must have with us everywhere we go raise the chance of losing such files. To prevent this inconvenient method of transporting private information, our suggested solution uses the cloud to store users' myriad of files of any size in encrypted form, encrypting them using the AES-256 version technique to ensure that your private data remains secret. The user no longer has to be concerned about security breaches when sharing his file with the intended recipient, since he may now provide permission to access his document to only the people he chooses, and it will only be accessible to him. With this approach, the saving and exchange of data is made possible. Our encryption algorithm uses the AES-256 version, in which all data is grouped into a string of bits and is subsequently converted into 128-bit blocks.

Keywords : File sharing, AES-256, SHA-256, Secure sharing, Software Requirement Specification.

I. INTRODUCTION

Nowadays, gadget storage is a big problem for smartphone users since we only have a certain amount of capacity, most of which is used up by entertainment or other programs that leave little room for essentials. Storage of any sensitive documents and files on mobile devices raises security issues. Data saved on mobile devices is not secure since it is easy to hack a device, infect it with a virus, cause damage, or steal it; in any of these scenarios, the user loses the contents. Therefore, we provide you with this technology that enables the server to save your material in an encrypted manner. The issue is that you'll need to retrieve the data first and then transmit it if you need to send it to someone. However, this solution also addresses this issue, allowing you to distribute the material without requiring a download. This is achieved by implementing an email to the recipient, who may then access the file by keying in the file ID while the system takes care of the rest. Here, the user is free to specify any length of key, and the output is saved in the cloud after the file has been encrypted using the AES-256-bit technique utilizing the SHA-256 algorithm on the backend side. [4][5][6][7][8].

Internet users have been drawn to web storage solutions because they allow them quick access anywhere, at any time. Numerous internet providers have flourished to support user profiles, company owners, and entrepreneurs in having their information in the cloud in a secure and reliable manner. The usage of mobile devices by those who

require access to information or operations from cloud-based systems while on the road with the aid of their mobile devices is continuously growing. Leveraging cloud services for data storage and transmission by smartphone subscribers is a difficult undertaking. The service providers' cloud environments come in the form of public, private, or hybrid clouds. The sort of public cloud is chosen by the user in accordance with their exposure or privacy policies. Numerous IT behemoths are embracing cloud services to save on their on-premises costs, which are more than what they charge internet service suppliers. [9][10][11]

Due to the frequent transfers of sensitive information between businesses, there is a chance that the information might be accidentally lost or stolen. This is unreliable because it poses a significant risk to the companies. The project is an effort to guarantee the security and privacy of data being transmitted via the Internet. To prevent any economic losses or cyberattacks that might hurt the firm, it is crucial that this data transmission stays out of the wrong hands. Additionally, only authorized individuals have access to the information transfer and storage, making it safe to manage and transmit.

Five sections make up the remainder of the essay. Section 2 discusses the past research on cloud storage systems and presents the approaches applied to solve the issues. The description of the suggested data hosting method is provided in Section 3. Section 4 presents the experimental outcomes based on the modeling and associated quantitative evaluations. Finally, Section 5 concludes the study by summarizing it.

II. LITERATURE SURVEY

[1] In this research paper, the author has developed an application that enables us to share files over the cloud securely. This application uses the AES algorithm for encryption and decryption. A user has the ability to share the files only with the users they wish to. AES uses large key values like 192, 256, and 128 bits, for encryption. Thus, the AES encryption has become more secure against hackers. AES is primarily used in activities including financial transactions, e-commerce, wireless communication, encrypted data storage, etc. The reason for choosing AES is that it is fast and, unlike DES, the number of shots in AES is flexible and is determined by the key length. Publisher: International Journal of Computer Application (IJCA) ((2250- 1797) Title: IMPLEMENTING CRYPTOGRAPHIC TECHNIQUES FOR SHARING FILES USING ACCESS CONTROL.

[2] In this research paper, the author has developed an application where we can send files from one computer to another in encrypted form. The entire computer is connected through one server, and we can share files and even access files from other computers through the PC folder. When we login in this study, the essential difficulty is the key. If we press the wrong key three times, the admin will block that user. So always remember the key given at log-in time. This improves file transmission security. After you've finished uploading the file, first encrypt and then decrypt the file to show the received data. And through the PC folder, we can access files from different connected computers. Publisher: International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE) Title: Design of Secure File Transfer over Internet.

[3] In this research paper, the author discusses smartphones, cloud technology on the go, and Android technology. They put up a plan for using two techniques to share data and files securely across public networks: the AES data is transferred, five algorithms and message digests are used. This security system is implemented on Android mobile devices. With this architecture, we can quickly retain and fetch the information while transferring it safely on both mobile and cloud devices. Publisher: ACEIT

Conference Proceeding 2016 Title: Secure Data Transfer & File Sharing Use of Cloud Service for Mobile Application.

[4] In this research paper, the author has implemented symmetric key techniques that are used in a safe storage mechanism. Our test findings demonstrate that a variety of network variables, including access time, acknowledgement time, and bandwidth consumption, have an impact on how files are processed in a cloud infrastructure. The current proposal addresses file security from unauthorized attackers by splitting documents into alternative complements and stashing parts in different clouds, but once again, the main issue is that the system required resources are too strong and the cloud service services are too reliant on one another. In our research, we put forth a strategy to guard against malicious user attacks across a particular cloud infrastructure. Additionally, the system can prohibit storing by malevolent users by disabling the component that stores file info. Publisher: INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR.ORG) Title: Design and Implementation of Secure File Storage using Distributed Cloud Mechanism.

[12] In this research paper, the author has proposed a system for sharing files in a safe way to store, transfer, and retrieve information in the cloud using public key infrastructure by assuring and meeting the three crucial security requirements. The suggested design depends heavily on CKMS and protects the file, which is accessible by many clients and secured against outside or internal intrusion. Additionally, a little comparison is provided between the various authentication and key agreement techniques in terms of how long it takes to produce the key and how long it takes to encrypt and decrypt data. The core transfer mechanism is used to exchange the key between the administrator, users, and CKMS, ensuring legitimacy. We concentrated primarily on guaranteeing security for the files or data saved in the cloud in the proposed system, not even on the regulation of levels of access. Therefore, managing the revocable process, traceability, and access controls effectively represents the most unsolved difficulty. Finally, we evaluated parameters such as Memory and cpu utilization as well as time spent performing decryption and encryption for various PKI techniques. Publisher: Journal of Computer Networks and Communications (JCNC) Title: An Efficient Framework for Sharing a File in a Secure Manner Using Asymmetric Key Distribution Management in Cloud Environment.

Advantage	Limitation
Users can share their confidential files securely using the AES algorithm which uses 128 bit keys.	The proposed research work has limitations of sharing limited data.
If a user inputs the wrong key 3 times then the admin automatically blocks that user	This application only works with computers connected to the server. It cannot work on any other devices. So it is not portable.
Proposed research includes splitting files into the problem of file safety from malevolent users is resolved by splitting files into many portions and storing them in various clouds.	Resources are insufficient, and cloud services storage services are required providers come across hence cost increases.
Researcher has implemented the entire secure file sharing app on android platform (which is the most used OS in the world)	Again it is platform dependent application means it can only run on android platform.

Table 1: Advantages and Limitations of State-of-the-art

How does our application tend to solve some limitations imposed by revived research papers ?. The application which we are aiming to develop will be Portable so it can be made available on every platform. Our application uses Simple Storage Service (S3) of AWS so the limitation of limited storage of storing files can be overcome. Cloud Resource requirement is not so high as we are storing all the encrypted files on a single instance of S3 Cloud service of AWS.

III. PROPOSED METHOD

3.1) Detailed Module Description:

1) Register User: In order to use this system, users need to get themselves registered with the system by providing basic credentials. You are only allowed to access the full system and use it fully if you authenticate with it. This can be done by logging in to the system.

2) My Files: With the help of this module, users are able to see all the files uploaded by them to the server/cloud. Users can select files of any type, like img, docs, pdf, etc., which can be uploaded to the server and are encrypted utilizing a range of cryptographic techniques like AES and will be stored in the cloud service S3 bucket of Amazon Web Services (AWS).

Delete Files: Users can delete their uploaded files from the system.

Share Files: With the help of this functionality, users can share their uploaded files with other authorized users who are authenticated to the system by giving access rights to the intended user. By sharing a file, the intended user will get the necessary key for decrypting the shared file via registered email using the AWS Simple Email Service (SES) service.

3) Shared Files: Users can access all files shared with them by other authenticated users after being decrypted on the server, and they will be downloaded on the user's system. A user can access that file by specifying the received decryption key for that particular file.

3.2) Project Software Requirement Specification SRS:

1) Use Case Diagram:

Use case diagrams are a standard way to communicate the key functions of a software system. At its most basic level, a use case diagram may be a depiction of an interaction of the user with such a technology that demonstrates the relationship among the user and the system and, as an outcome, the several use scenarios where the user is engaged. A use case diagram, which is frequently among several other types of diagrams, may be used to examine the different kinds of users of both systems.

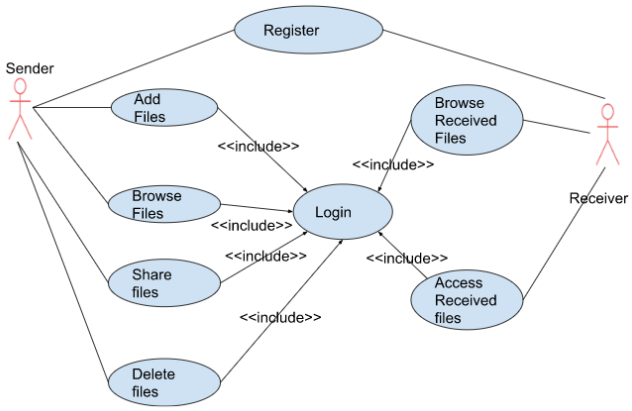

Figure 1: Use Case Diagram

2) Data Flow Diagrams (DFD) provide the program's functional layout. Any research problem between a system's "user and system analyst" is readily bridged by its graphical depiction. It includes everything from a system description to a hierarchy-based exploration of the system's intricate design. DFD displays both the opposing flow of data inside a system and external entities from whom data enters the process. It also covers the way that the process alters the stream of

data, and as a result, the data storage used to spell correct documentation.

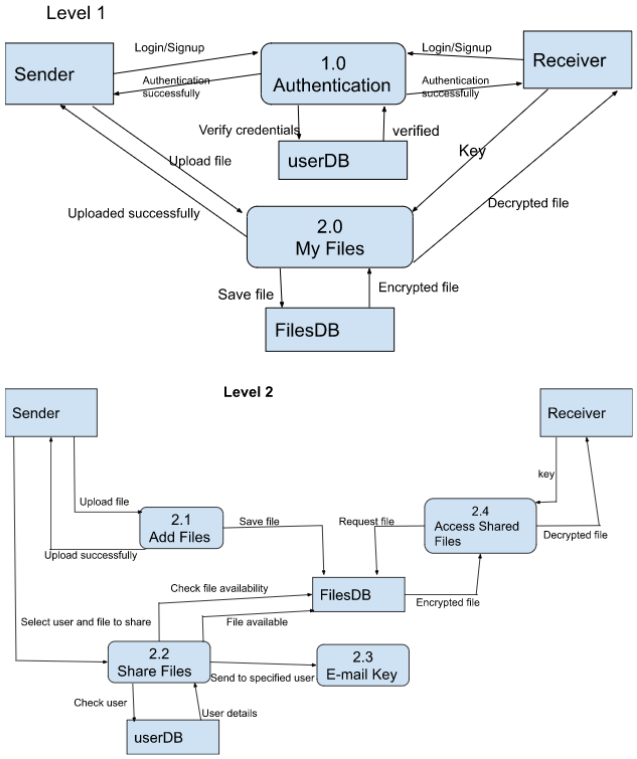

Figure 2: Data Flow Diagram

3) Entity Relationship Diagrams:

The design phase of a future database is represented using an object model. The ER model abstracts individual objects as units and models various potential links between them as connections. The ER diagram is used to depict the properties, entities, and relations. Table layouts are generated using ER diagrams, together with the appropriate restrictions. Finally, excess is removed from these tables, and integrity of data is maintained. Therefore, the ER diagram needs to be created as exactly and properly as possible in order to have data saved effectively.

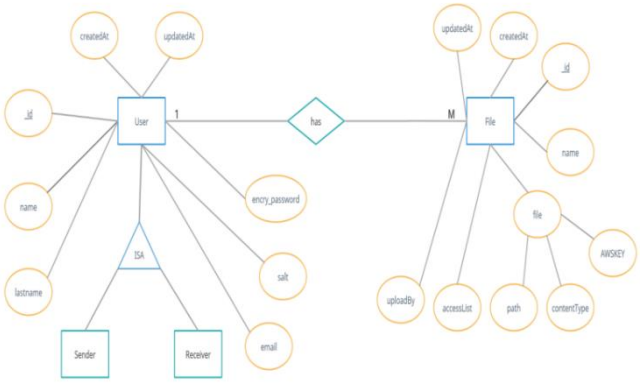

Figure 4: ER Diagram

4) Event Trace Diagram:

An engagement chart that places emphasis on the timing of information is known as an action trace diagram. It displays a set of items together with the information those items have sent and received. A table with items placed all along the X axis and communications sorted in extending the time along the Y axis might represent a flow chart graphically.

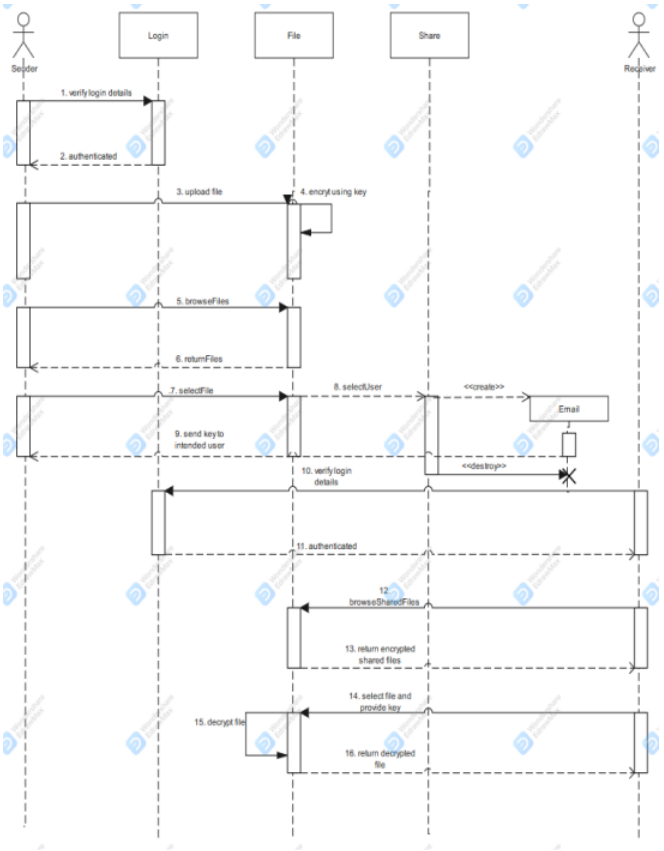

Figure 4: Event Trace Diagram

5) State Diagram:

The state diagram models the lifespans of items in each class to represent the transient characteristics of objects across time. Each item is thought of as an independent being that interacts with the rest of the world by recognising events and reacting to them. The sorts of modifications that entities may recognise are represented by events. An event is frequently defined as something that could have an impact on an item. Throughout the course of an object's lifespan, the object must always be in a particular state. Due to the obvious effects of an event that it is affected by, an item might travel from one point to another. In a state diagram, there is frequently just one initial state, but there may be several intermediate and end states as well.

Figure 5: state Diagram

6) Class diagram:

The basic edifice of entity modeling is the class diagram. And is used for both precise modeling, which converts the models into computer code, and for oriented modeling of the device's classification. Data modeling may also employ class diagrams. The class diagram serves as a representation of the programming classes as well as the user's most important components and interaction. A class diagram is used in the system design to identify different classes and organize them together to determine the static relationships among them. The categories of the design concept are frequently split into a series of subcategories when using thorough modeling.

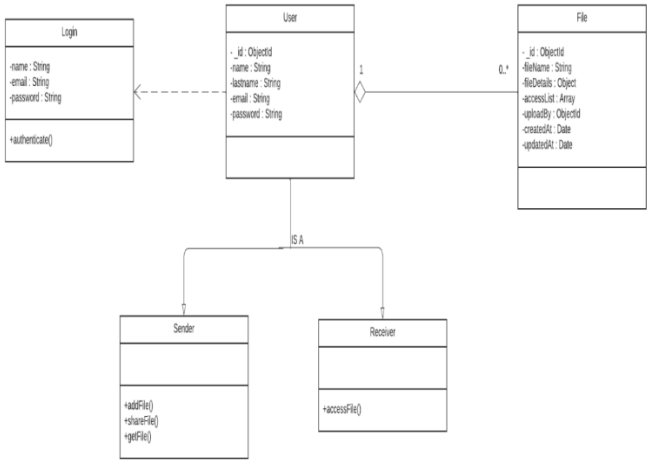

Figure 6: Class Diagram

3.3) Database Design:

1. Table Name: User Table

   Description: To store the data about the users of the system

   Primary Key: _id

Sr.No.	Name	Datatype	Constraint	Description
1	_id	ObjectId	Primary Key	To uniquely identify each user
2	name	String	Not null	To store the user name
3	Last name	String	Not Null	To store the user last name
4	email	string	Not null	To store the email
5	Salt	string	Not null	To store the salt
6	Encry_password	string	Not null	To store the encrypted password
7	createdAt	Data	Not null	To store the data of creation of record
8	updatedAt	Date	Not null	To store the data of last updation of this record

Table2. User Table in Database

2. Table Name: Files Table

   Description: To store details about file uploaded by users

   Primary Key: _id

   Foreign Key: uploadBy , which is the field in Files Table and Primary Key of the User Table.

Sr.No.	Name	Datatype	Constraint	Description
1	_id	ObjectId	Primary	To uniquely identify each file
2	name	string	Not Null	To store the file name
3	file	Object	Not Null	To store details about file
4	accessList	Array	Not Null	To store the id of users who has right to access this file
5	uploadBy	ObjectId	Foreign Key	To store the id of user who uploaded this file
6	createdAt	Date	Not Null	To store the data of creation of record
7	updatedAt	Date	Not Null	To store the date of last updation of this record

Tabel 3. Files Table in DataBase

Sr.No.	Name	Datatype	Constraint	Description
1	path	String	Not Null	To store the actual path of the file stored in AWS S3 Stroage
2	contentType	String	Not NUll	To store content type of the file uploaded
3	AWSKEY	String	Not NUll	To store the access key of the file which is stored in AWS S3

Table 4. Description of the 'file field' of Files table:

The software and tools used in the construction of secure file sharing control include Postman (which is used to test the REST APIs), Nodes JS (which offers a runtime environment for Java script), and Figma (a vector graphics editor and prototyping tool). This facilitates the creation of application template designs) and Visual Studio Code (Microsoft created Visual Studio Code, generally known as VS Code, a source-code editor for Windows, Linux, and macOS.) We save the information obtained from the browser using AWS. When you signed up, a key was issued for each file that you wanted to share with others, making it easy to access the website's management panel and store essential data.Although an AWS account is required to secure applications using FileMaker Server (Amazon Web Services, commonly referred to as AWS, is a subsidiary of Amazon.com, Inc.), the tools needed to construct this kind of application are free. However, when your memory runs out, this website is ideal for saving all of your information over a secure link to the cloud.

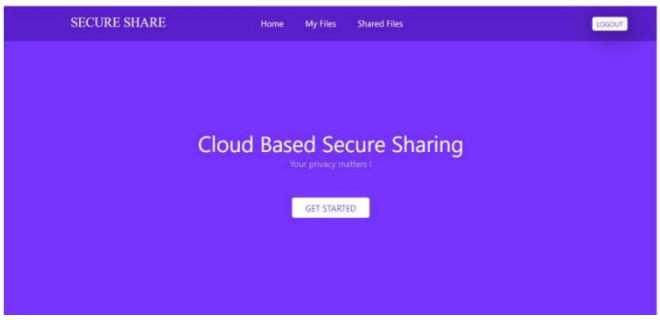

Figure 7: Home Design UI

The cryptographic algorithms that encrypt plaintext with almost the same private key and decode it with the same keys. There can be a straightforward conversion that connects the two keys, or the keys might be similar. In reality, the keys stand for a private key that two or more people might use to keep a link to confidential information open. One of the primary disadvantages of private-key encryption in comparison to public-key encryption is that both parties must have accessibility to the private key. But for mass encryption, symmetric-key techniques are often preferable. They require less storage capacity and transmit data more quickly since their keys are smaller. The Rijndael block cypher is the basis for AES, which was created by 2 Belgian cryptographers, Vincent Rijmen, and Joan Daemen who presented a suggestion to NIST as part of the AES recruitment process. Various key or block sizes are available in the Rijndael family of cyphers. NIST chose three Rijndael family members, each having a 128 bits, and three distinct key lengths: 128, 192, and 256 bits, for use with AES. A crucial component of contemporary computer security is encrypting. Data is encrypted using encryption techniques like AES 256 and PGP, which are then decrypted once it reaches its intended destination. But what if you need to obfuscate

information in a way that can never be undone? Hashing is used in this situation. This essay will look at the commonly used hash algorithm SHA-256 and how it fits into modern cybersecurity.

Data owners must first register with a reputable platform in order to save their files on a cloud infrastructure. This data is stored as objects inside containers, and it's saved on distributed storage nodes. The login information is created by the client and is used to submit client profiles and permits as well as files. When the TTP receives a specific file (F) from the data holder, it uses asymmetric key encryption to produce keys. It is expected that any normal asymmetric key generation algorithm be used for this reason, as asymmetric key creation is not described in this article. The TTP file directory is created for each user to ensure confidentiality of the data.

Node JS technology is used to execute the suggested technique in the cloud infrastructure. Although the test files aren't particularly large, performance has been guaranteed by carefully handling privacy and optimization issues. For file holders, a personality user interface has been created. A file owner creates their own credentials and registers with the Trusted Third Party system through the UI. After being added to TTP, the users receive an invitation from the owner. The invite that is issued to each user's email address contains instructions for setting up the credentials. TTP employs the suggested asymmetric key encryption technique to protect data in a cloud setting. The establishment of both specific users and bulk uploads of the setup may now be done.

With the use of a proprietary sharing algorithm, the key pair is divided into two halves. Part of the public key is kept by the TTP, while the other half is split here between the device's user and a third party who cannot access the key at the same moment as the tool's owner. This configuration enables the user to retrieve their personal key in the event that it is misplaced or stolen. Users connect to the TTP and ask for a certain file and the matching key component. TTP safely rewrites the files in favor of its user after distributing the secret portion of the public key. TTP obtains the encryption content via data storage and uses the newly created decryption key to decode it. The desired user is subsequently given access to the encrypted file. Prior to exchanging the public key with the user, TTP verifies the person's rights and recreates the key. TTP then erases the decrypted data and the disassembled public key when the procedure is finished.

IV. RESULTS AND DISCUSSION

Time may be a crucial factor when creating encryption, decryption, and key procedures. All tests are conducted using Node JS on an Intel (R) Core-5 with a 2.5GHz processor, 8GB of RAM, and a Windows 10 application framework. the quantity of your time a machine has spent could also be wont to gauge the performance of any method. Anybody who wishes to make an efficient system must guarantee that these 3 steps within the created cryptosystem will probably be quicker.

The cloud-based safe file sharing design template includes elements like home, files, shared files, and uploading files with signup and login within the right formats. Additionally, there was an enclosed file area that required me to pick out a file and kind an appropriate key in excess. The key we entered during file sharing helped ensure a secure share and kept the file private. Additionally, the file included an icon so we could download it with the suitable key and share it with others via key generation. The intended web service was tested using custom test cases, which include

1) When signing up, the required information must be provided; otherwise, a pop-up message stating that "Every field must be filled" for join up validation will appear on the screen.

2) The system displays "User already existent in DB" if the data is already available within the database.

3) The passwords we were inputting must be the identical, otherwise, "Confirm password is different" would seem.

4) The sign-up page's appropriate content is followed by a redirect to the sign-in page with the message "Success sign on."

5) The message "Provide Proper Email and Password Sixe should be greater than or capable 3" appears if the e-mail address or password were left blank.

6) If the supplied information is invalid, a pop-up message stating "User email or password doesn't exist" appears over the screen.

7) If the supplied information is valid, "You are signed in" appears.

8) an accurate name and key were necessary within the Add Files section; otherwise, it'd say "Enter file name and Enter Key."

9) For a secure procedure, "All fields must be filed" is stated within the shared files section.

10) The message "user not found with the email" appears if the entered email doesn't exist already within the database.

11.) a sound key must be accessible within the mail so as to access other users' files. When the proper pin is entered, a successful download begins.

For each account generated on the cloud surface, a special ID was produced that would be wont to access data throughout the cloud and will be accustomed to recognize the user when a selected file was given thereto ID. Additionally, to share across cloud users without using any third-party services, a sharing user's ID requires an energetic user email with multiple cloud accounts. Different users may view the identical data. If a user provides over one user, an inventory of group user IDs is stored within the cloud and made available for access by all users. The identical secret's then emailed to any or all users within the group. For example, if a user with ID 6 generates account data and shares it with all the cloud users, then this data is accessible by anyone using the following: 'https://cloudmongodb.com/open?ID=user-6 If we run out of space on our devices, the cloud saves the imported files and is necessary on a secure surface with an endless amount of space for storing for you and maintains the file secure and guarded within the cloud with key generation. Additionally, to share across cloud users without using any third-party services, a sharing user ID requires a full of life user email.

Figure 8: mongodb profile visual

Figure 9: mongodb user file sharing details

V. CONCLUSION

The application developed enables us to share files over the cloud securely. This application uses the AES algorithm for encryption and decryption. A user has the ability to share the files only with the users they

wish to. AES uses higher encryption key sizes, including 128, 192, and 256 bits. The reason for choosing AES is that it is fast and, unlike DES, the number of financial transactions, wireless communication, e-business, encrypted data storage, and more applications employing AES is growing. Rounds in AES are a changeable algorithm that is dependent on the key length. At the current stage, we are only using the AES algorithm, and to enhance security even further, we can use more hybrid versions of the AES algorithm to better meet user needs. We can also use distributed cloud storage to store encrypted files. Various databases can be used to store large amounts of data. This system can also be a part of many security-centric systems in order to address security related issues.

VI. REFERENCES

[1]. Ravi, A., Raj, K. S., Reddy, M. G., & Srikar, M. S. S. IMPLEMENTING CRYPTOGRAPHIC TECHNIQUES FOR SHARING FILES USING ACCESS CONTROL.

[2]. Pragya Gupta et al., "Mobile Cloud Computing: The Future of Cloud" International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering Vol. 1, Issue 3, September 2012.

[3]. Suhas Holla et al., "ANDROID BASED MOBILE APPLICATION DEVELOPMENT and its SECURITY" International Journal of Computer Trends and Technology- volume 3 Issue 3- 2012.

[4]. Khan AN, Mat Kiah ML, Khan SU, Madani SA. Towards secure mobile cloud computing: A survey. Futur Gener Comput Syst. 2013; 29(5):1278–99.

[5]. Rajathi A, Saravanan N. A survey on secure storage in cloud computing. Indian Journal of Science and Technology. 2013 Apr; 6(4):1–6.

[6]. Kumar R, Rajalakshmi S. Mobile cloud computing: Standard approach to protecting and securing mobile cloud ecosystems. Proceedings of International Conference on Computer Sciences and Applications; 2013. p. 663–9.

[7]. Lee JY. A study on the use of secure data in cloud storage for collaboration. Indian Journal of Science and Technology. 2015 Mar; 8(S5):33–6.

[8]. Uddin M, Memon J, Alsaqour R, Shah A, Rozan MZA. Mobile agent based multi-layer security framework for cloud data centers. Indian Journal of Science and Technology. 2015 Jun; 8(12):171–8.

[9]. V.Malligai et al.," Cloud Based Mobile Data Storage Application System" ternational Journal of Advanced Research in Computer Science & Technology (IJARCST 2014).

[10]. Muhammad Shiraz et al.," A Review on Distributed Application Processing Frameworks in Smart Mobile Devices for Mobile Cloud Computing" IEEE communications survey & tutorials, vol 15 no 3, third quarter 2013.

[11]. P.Srinivas et al.,"Secure Data transfer in Cloud Storage Systems using Dynamic Tokens" International Journal of Research in Computer and Communication technology, IJRCCT, ISSN 2278-5841, Vol 2, Issue 1, January ,2013.

[12]. Pradeep, K. V., Vijayakumar, V., & Subramaniyaswamy, V. (2019). An efficient framework for sharing a file in a secure manner using asymmetric key distribution management in a cloud environment. Journal of Computer Networks and Communications, 2019.

[13]. ANDRADES, B., MICHAEL, P., VAZ, R. L., & Guide, A. A. (2018). Secure File Sharing using Access Control. ST. FRANCIS INSTITUTE OF TECHNOLOGY.

[14]. Tang, Y., Lee, P. P., Lui, J. C., & Perlman, R. (2012). Secure overlay cloud storage with access control and assured deletion. IEEE Transactions on dependable and secure computing, 9(6), 903-916.

[15]. Yu, S., Wang, C., Ren, K., & Lou, W. (2010, March). Achieving secure, scalable, and fine-grained data access control in cloud computing.

In 2010 Proceedings IEEE INFOCOM (pp. 1-9). Ieee.

[16]. Wang, S., Wang, X., & Zhang, Y. (2019). A secure cloud storage framework with access control based on blockchain. IEEE access, 7, 112713-112725.

[17]. Wei, J., Liu, W., & Hu, X. (2016). Secure and efficient attribute-based access control for multiauthority cloud storage. IEEE Systems Journal, 12(2), 1731-1742.

[18]. Fotiou, N., Machas, A., Polyzos, G. C., & Xylomenos, G. (2015). Access control as a service for the Cloud. Journal of Internet Services and Applications, 6(1), 1-15.

Cite this article as :

Jay Patel, Akshit Trivedi, "Cloud Based Secure File Sharing Using Access Control", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 8 Issue 4, pp. 337-348, July-August 2022. Available at doi : https://doi.org/10.32628/CSEIT228458
Journal URL : https://ijsrcseit.com/CSEIT228458