Volume: 13 Issue: 05 Pages: 5143-5151(2022) ISSN: 0975-0290

Design of a Secure Virtual File Storage System on Cloud using Hybrid Cryptography

Bello A. Buhari
Department of Mathematics, Computer Science Unit, Usmanu Danfodiyo University, Sokoto, Nigeria
Email: buhari.bello@udusok.edu.ng
Aliyu Mubarak
Undergraduate Student, Department of Mathematics, Computer Science Unit, Usmanu Danfodiyo University, Sokoto, Nigeria
Email: aliyumubarak448@gmail.com
Bello A. Bodinga
Department of Mathematics, Computer Science Unit, Usmanu Danfodiyo University, Sokoto, Nigeria
Email: bello.bodinga@udusok.edu.ng
Muazu D. Sifawa
Management Information System Usmanu Danfodiyo University, Sokoto, Nigeria
Email: mzsifawa@gmail.com

ABSTRACT

As Security is becoming more and more useful in the field of computing, users would like to be sure of how secure their files are on a system, as security is one of the most crucial fields in networking and file storage. Dependable file storage and access establish several security issues in a cloud computing. This research designed and implemented virtual secure file storage system on cloud using hybrid cryptography. The cryptography method used for file encryption and decryption is AES and SHA-2 hash function. It is implemented using Cloud APIs with REST calls or client libraries in PHP. The system interfaces were developed using HTML, CSS and JAVASCRIPT. Back end development was done using PHP, MYSQL and GCP Cloud Storage Library then the file encryption and decryption was achieved through PHP classes which includes `open_ssl_file_encryption` and decryption (AES) and also MCRYPT function. The proposed virtual system is also compared with some latest related works.

Keywords - **Cloud Computing, Secure file storage, hybrid cryptography, AES, SHA-2, Google Cloud Platform**

Date of Submission: Mar 31, 2022 Date of Acceptance: Apr 18, 2022

I. INTRODUCTION

The simplest activity that a computer does is to store files in its vacant space and retrieve these files whenever needed by the user and is the basic services offered by Cloud Computing [1]. These files can contain the category of data we use in our day to day life from simple photographs, written documents, favorite songs, or even save movies to huge bulk amounts of data which are confidential.

As Security is becoming more and more useful in the field of computing, users would like to be sure of how secure their files are on a system, as security is one of the most crucial fields in networking and file storage. Users are always faced with the problem of theft of devices either a phone or system, and mostly important files kept in these devices are gone, so implementing cloud based platforms or software may somehow reduce the loss of files and information.

Cloud Computing utilizes network of huge amount of servers usually running lower cost customer PC technology with unusual connections to perform data processing tasks across end users. Cloud storage service avoids the cost pricey on software, personnel maintenance and ensures better performance, less storage cost and scalability [2]. Dependable file storage and access establish several security issues in a cloud computing [3]. Even digital library can be more effective, secure and efficient by incorporating security methods and cloud storage [4] and secure online civil registration [5].

None of the current cloud storage services providers provide security guarantees in their Service Level Agreements (SLAs) [6]. These providers include Microsoft Azure, Google Cloud, Alibaba Cloud, IBM Cloud, Oracle, Salesforce, SAP. Therefore, storing confidential data on cloud comes with severe security risks because cloud can leak the data, modify the data, or return inconsistent data to different users.

II. RELATED WORKS

Kute and Javheri in [7] implemented a Secure File Storage on Cloud with owner-defined Attributes for Encryption. Whenever the owner uploads a file, it is labeled with a set of attributes that includes: department, work profile, branch, experience which is called as access structure. After this time period, date and location are also added. The user cannot decrypt and download the file if the time interval, date location and attributes do not matches with the owner set attributes. File is split into multiple

Volume: 13 Issue: 05 Pages: 5143-5151(2022) ISSN: 0975-0290

fragments according to file size and stored on multiple nodes instead of being stored on a single node to get more security.

Swarna and Eastaff in [8] presents the file security model which uses the concept of hybrid encryption scheme to meet up security needs. Encryption and decryption of files at cloud servers accomplished using blowfish and modified version of RSA and it is tested in Open Nebula cloud environment.

Poduval et al. in [9] introduced a new security method that uses a combination of multiple cryptographic algorithms of symmetric key and steganography. 3DES, RC6 and AES algorithms are used to ensure security to data. All the algorithms utilize 128-bit keys. LSB steganography method is employed to store the key information securely. File is divided into three parts during encryption. These individual parts of the file will be encrypted using dissimilar encryption algorithm simultaneously with the help of multithreading method. The key information is inserted into an image using the LSB method. Their methodology ensures better security and safety of customer data by storing encrypted data on a single cloud server, using AES, 3DES and RC6 algorithm.

Wang et al. in [10] proposed a new secure cloud storage scheme with access control by using the Ethereum blockchain technology. Their framework is a combination of Ethereum blockchain and ciphertext-policy attribute-based encryption (CP-ABE). In their scheme there is no trusted third party in the system. Also, the data owner can store ciphertext of data through smart contracts in a blockchain network, can set legal access periods for data usage so that the ciphertext can only be decrypted during legal access periods and the formation and invocation of each smart contract can be stored in the blockchain.

Wu et al. in [11] proposed privacy-preserving deduplication cloud storage with public cloud auditing (CPDA). Their framework achieves secure file deduplication on encrypted file, utilizes the convergent encryption and random masking method to achieve data privacy during the file deduplication and integrity auditing process and not only supports each data owner to separately launch the integrity auditing of their own files, but also supports cloud server to occasionally assign the third party auditor to concurrently handle multiple auditing tasks to ensure the integrity of the outsourced files. The security of their scheme is properly proved and its performance is verified by numerical analyses and simulation experiments.

Mohammed and Ibrahim in [12] developed several levels of multi-coding levels using more than one technique to obtain more privacy through DNA encryption and adding a higher level of privacy by adding Advanced Encryption Standard (AES) and then loading it into the cloud storage. They used MATLAB to implement and analyzed the proposed scheme. It has shown that the proposed scheme has a realistic level of security, efficiency, complexity and speed.

Fan et al. in [13] proposed a Trusted Execution Environment (TEE) based secure deduplication scheme. In their scheme, each cloud user is assigned a privilege set; the deduplication can be done if and only if the cloud users have the right privilege. It also augments the convergent encryption with users' privileges and relies on TEE to offer secure key management. This improves the capability of such cryptographic system to resist chosen plaintext attacks, chosen ciphertext attacks and is secure enough to maintain data deduplication and to defend the privacy of sensitive data. They implement a prototype of their scheme and evaluate its performance which shown that the overhead of their scheme is practical in reasonable environments.

Sharma et al. in [14] due to security issues like information security, protection, secrecy, and verification they employ of hybrid encryption with crossbreed cryptographic calculations to improve the security of information or data file on cloud.

Sharma et al. in [15] proposed the employment of a Blockchain to provide secure administration and investigation of health services enormous information that rely on their conveyed nature and other extra protection and security properties of the system and distributed storage. They utilize data security to ensure better security of structure. They also perform security assessment and expansive appraisals on various particular pieces of the proposed structure, and showing positive conditions of their suggestions over existing courses of action.

Seth et al. in [16] present insights into the implementation of a fresh architecture that can convey an enhanced level of security for outsourcing information in a cloud computing environment while involving several independent cloud providers. Their framework consists of dual encryption and data fragmentation techniques that visualize the secure distribution of information in a multi-cloud environment. All simulations and scrutiny have been accomplished on an Oracle virtual machine Virtual-Box and a Fog environment on an Ubuntu 16.04 platform. They showed that their projected proposal is greatly proficient and satisfies the security prerequisites of secure data sharing and can efficiently withstand security attacks.

Kumar and Shafi, in [17] proposed an effective mechanism with a distinctive feature of data integrity and privacy in order to guarantee that the user's data in the cloud are secure. They recommend a technique for providing data storage and security in cloud using public key Cryptosystem. The public key cryptographic algorithm used is modified RSA algorithm to provide enhanced security for the data stored in the cloud.

Abdel-Kader et al. in [18] proposed an efficient two-stage cryptography scheme to access and store data into cloud

Volume: 13 Issue: 05 Pages: 5143-5151(2022) ISSN: 0975-0290

securely. It consists of both user authentication and encryption processes. A two-factor authentication scheme one-time password is proposed to defeat the weaknesses in the existing authentication schemes. This proposed authentication technique does not need specific extra hardware or additional processing time to identity the user. Their plaintext is divided into two parts which are encrypted separately using a unique key for each plaintext. The keys are generated using logistic chaos model theory. Their scheme guarantees high level of by introducing different security processes with different stages. Their simulation results shown that the proposed scheme reduces the size of the ciphertext and both encryption and decryption times than previous schemes.

Rashmi et al. in [19] proposed an enhanced RDPC protocol for cloud storage is which checks the data integrity. Their research is based on homomorphic hash algorithm; also supports dynamic operations such as insert, update, delete and modify at block level, for data dynamics. Merkle Hash Tree is used which helps in finding the location of each dynamic operation and third party auditor checks the user's data for correctness and gives the accuracy of the data that is stored in cloud server. Deduplication method is used to ensure whether the file that user need to store in cloud storage is already exist at cloud server or not. Their scheme is effective and withstand replace attack launch by malicious server.

Sheeja et al. in [20] proposed new secure cloud storage system to ensure that the details of companies are secure from all data operator and third-party inspector, respectively by protected sharing of information utilizing RSA and AES calculation to keep up security inside the cloud server. The proposed system strengthens the intensity of authentication with the help of AES and time-stamping algorithms. Their experimental results showed the efficiency of the proposed method when auditing the shared data integrity.

Viswanath and Krishna [21] developed an encryption algorithm for storing big data in the multi cloud storage for restricting the insider attacks. The proposed framework consists of data uploading, slicing, indexing, encryption, distribution, decryption, retrieval and merging process. The Simulation analysis is carried with real time cloud storage environments and recorded around 2630 KB/S for the encryption process. The results prove that the proposed algorithm superior compared to the related mark algorithms.

Karati et al. [3] develop a secure file storage and access protocol for cloud-enabled Internet of Things environment. The robust protocol proposed ensures data sharing among geographically isolated physical devices and efficient access to cloud data for clients. They consider a 3-level hierarchical indexing technique and offer efficient data access by utilizing the functionalities of the bilinear pairing operation. Their formal security analysis shows that their proposed protocol resists the chosen-ciphertext attack and other essential attacks and empirical study confirms that the proposed protocol has better functionalities as regards to security aspects, operational attributes, and performance cost than other related schemes.

Chinnasamy et al. in [22] introduced a new hybrid method to achieve high data security and confidentiality in cloud computing environment. They combine ECC and Blowfish to implement a hybrid algorithm. The performance of their hybrid system is compared with the current hybrid approaches and shows that the proposed approach ensures high security and confidentiality of patient data

III. PRELIMENARIES

This section discusses the useful backgrounds of this research. These include Advanced Encryption Standard (AES) and SHA-2 Hash function..

A. Advanced Encryption Standard (AES)

AES is symmetric key algorithm and is measured as a type of the block cipher. It is commonly used in many industry standards and in numerous commercial systems like IPsec, the internet Skype, the IEEE 802.11i and TLS [23]. AES has a unchanging block size of 128 bits and a key size of 128, 192, or 256 bits and the block-size has a maximum of 256 bits, but the key-size has no theoretical maximum [24].

The sizes of data matrix is 128 bits and have 10, 12 and 14 rounds which rely on key length. AES operates on the Galois fled (2^8) with the primitive irreducible polynomial:

$$m(x) = x^8 + x^4 + x^3 + x + 1 \quad (1)$$

This mathematical expression used in substitution box, mix columns and also in the creation of key. Each round is comprised of the four functions [25]. The functions can be discussed as follows [26]:

SubBytes: This is an invertable non-linear transformation. It is a substitution procedure of each byte of the input state by another one from a predefined table called S-box. The size of the table is 256 different elements of bytes. The design decisive factor of the S-box values is to be resistant against the known differential and linear crypto-analysis. Each possible element of the S-box is generated by computing the multiplicative inverse in GF (2^8) and then applying an affine transformation.

ShiftRows: This is an operation of cyclically shifting each row with a different offset.

MixColumns: This is an operation on the different columns by performing a polynomial multiplication in GF (2^8).

AddRoundKeys: This is an operation of bit-wise XORing the round key (sub-key) with the current state. Each round key is derived from the previous sub-key. This requires the encryption algorithm to schedule the key for each round.

Volume: 13 Issue: 05 Pages: 5143-5151(2022) ISSN: 0975-0290

AES-128 encryption algorithm is as follows [27]:

```
Input: The 128-bit plaintext blocks P
and key K.
Output: The 128-bit ciphertext block C.
X ← AddRoundKey(P,K)
for i ← 1 to 10 do
    X ← SubBytes(X) X ← ShiftRows(X)
    If i ≠ 10
      X ← MixColumns(X)
    end
    K ← KeySchedule(K)
    X ← AddRoundKey(X,K)
end
C ← X return C
```

B. SHA-2 Hash Function

The Secure Hash Algorithm (SHA) is actually a set of cryptographic hash algorithms defined by the National Institute of Standard and Technology (NIST) in the Secure Hash Standard (SHS) for being employed by the U.S. government agencies. SHA-2 hash algorithms are well-known by the length of the output they produce [28]. The two basic variants are SHA-256 and SHA-512, which are the same algorithm, applied to dissimilar word lengths. SHA-256 operates on 32-bit words, whereas SHA-512 works on 64-bit words. The two variants differ also in some constant parameters and values, and employ different initialization values.

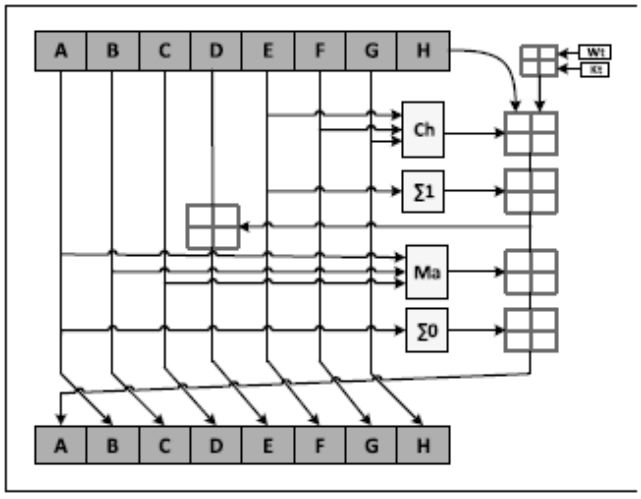

Figure 1. - The Function Block of SHA-2 [29].

DM = DM_0 to DM_7 = IV
for each data_block i do

$W = expand$(data_block)
A = DM_0 ; B = DM_1 ; C = DM_2 ; D = DM_3
E = DM_4 ; F = DM_5 ; G = DM_6 ; H = DM_7

for t= 0, t≤ 63 {79}, t=t+1 do
$T_1 = H + \sum_1(E) + Ch(E,F,G) + K_t + W_t$
$T_2 = \sum_0(A) + Maj(A,B,C)$
$H = G$; $G = F$; $F = E$;
$E = D + T_1$
$D = C$; $C = B$; $B = A$
$A = T_1 + T_2$
end for

DM_0 = A + DM_0 ; DM_1 = B + DM_1
DM_2 = C + DM_2 ; DM_3 = D + DM_3
DM_4 = E + DM_4 ; DM_5 = C + DM_5
DM_6 = D + DM_6 ; DM_7 = E + DM_7
end for

Figure 2. - Pseudocode for SHA-2 Algorithm

SHA-2 compression function evaluation requires a preprocessing phase. A message is padded and divided into blocks (Al-Odat et al., 2019). The functional block of the SHA-2 hash function can be shown in Fig. 1 and the algorithm can be shown in Fig. 2.

IV. DESIGN OF THE PROPOSES SECURE VIRTUAL FILE STORAGE ON CLOUD

In our proposed system there are two main entities: an owner of a file and another with whom the owner has shared access with. The owner will upload the file that is required to be stored at a remote location or needs to be shared with other users. Owner gives access to other users by sharing required metadata to decrypt the file using an asymmetric crypto system. The high level view of the system can be shown in Fig. 3.

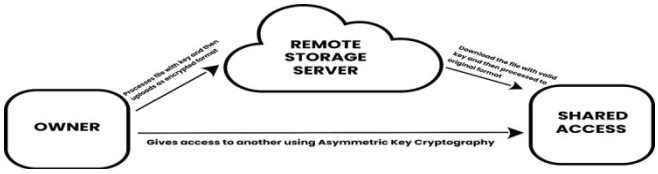

Figure 3. - High Level View of the System

A. *Architecture of the Proposed Secure Virtual File Storage on Cloud*

The architectural design of the proposed virtual secure file storage on cloud can be shown in Fig. 4.

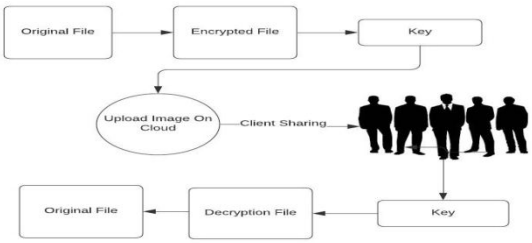

Figure 4. - Architecture of the proposed virtual secure file storage system on cloud

***Encryption Process*:** The Encryption process starts when the user wants to upload a file on the system, of course it does not upload unless a key is provided (user authentication). After the key is provided it starts by using the class.aes.enc.php file during a series of block size conversions. Rijndael is the family of ciphers with different key and block sizes. For AES, NIST selected each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.

a. Steps For Encryption

1. Register and login with correct login information.
2. Select a file which you want to upload.
3. Insert a key for encryption.
4. Applying hashing function on user key which will generate encrypted key.
5. Then applying AES (MAES) or AES with MCrypt Function (hybrid) on the selected file will generate an encrypted file.

6. Now, Store encrypted files along with encrypted keys in the cloud.
7. User file can then decide to share the key via email.

This can be shown in Fig. 5.

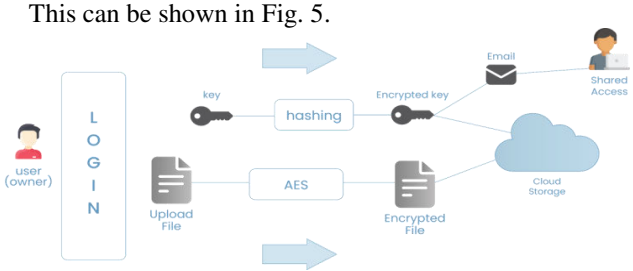

Figure 5. - Encryption Process

Decryption Process : For the decryption process, the system extracts and converts the encrypted data and changes that data into words and pictures which are easily understood by any reader and by any system. This process is done automatically after the user inputs the correct key for the file encrypted.

b. Steps for Decryption

1. Login with correct information.
2. Select a file which you want to download from the cloud.
3. Enter the correct key to download the file.
    a. If key is correct then allow access to download otherwise denied access to download.
4. Apply the correct AES key on the encrypted file.
    a. If key is correct then decrypt and allow access to the file otherwise denied accessibility.

This can be shown in Fig. 6.

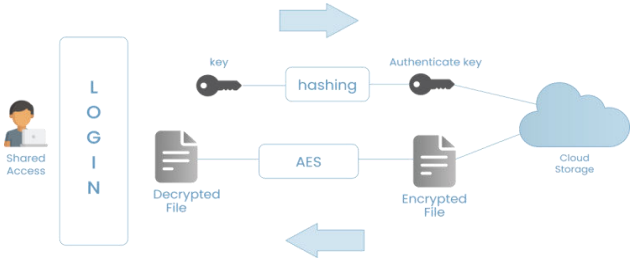

Figure 6. - Decryption Process

V. IMPLEMENTATION OF THE PROPOSES SECURE VIRTUAL FILE STORAGE ON CLOUD

The secure virtual file system is implemented using Cloud APIs with REST calls or client libraries in PHP. The system interfaces were developed using HTML, CSS and JAVASCRIPT. Back end development was done using PHP, MYSQL and GCP Cloud Storage Library then the file encryption and decryption was achieved through PHP classes which includes `open_ssl_file_encryption` and decryption (AES) and also MCRYPT function.

Finally, files are stored on Google Cloud Platform (GCP) using cURL function in PHP to POST and RETRIEVE data being stored there. All these are combine to provide elegant, responsive and user friendly user interfaces so as for users to be able to perform tasks seamlessly based on their roles. The system is designed using a single login page where each user is authenticating first before accessing the system. Below are some of the system interfaces.

B. *SIGN UP AND LOGIN IN*

The user sign in/log in is the first page that is displayed after the login button on the nav list of home page is clicked, here user are required to first provide username and password first before they can have access to system so if users are not registered already on the system the first have to sign up and a mail of confirmation is sent to their provided email address to confirm they are the one. Also the login page has other functionalities like the “Forgot Password?’ link where they can recover their password and have access back to their account. This can be shown in Fig. 7 and Fig. 8 respectively.

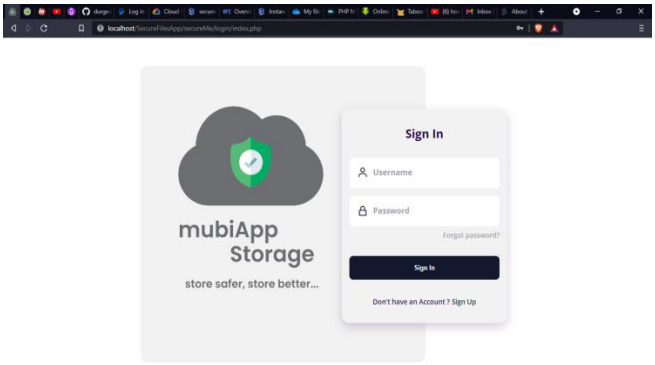

FIGURE 7. - Login Form

C. *USER’S DASHBOARD*

When user successfully login he/she will be directed to their user dashboard where he/she will be able to view functions/other pages like: My Files, Recycle Bin, Change Password and view storage space capacity and also logout, he/she can also search for files using the search bar on system. This can be shown in Fig. 9.

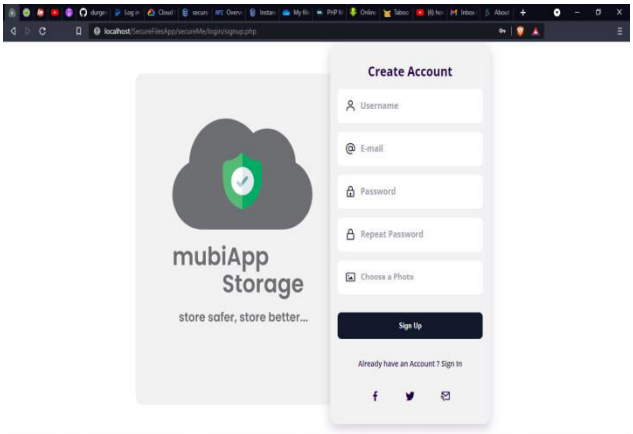

FIGURE 8. - Sign Up Form

Volume: 13 Issue: 05 Pages: 5143-5151(2022) ISSN: 0975-0290

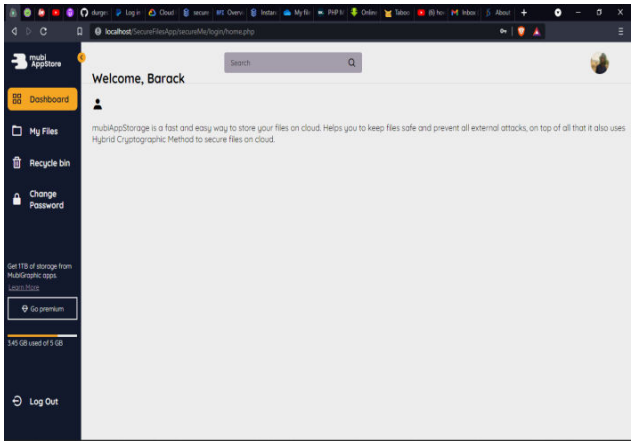

FIGURE 9. - User's Dashboard

D. *MY FILES PAGE*

Here files that are already encrypted and saved on both database and cloud platform are displayed. User can choose to decrypt or delete a file with a key. The key used to encrypt files that are uploaded in the upload page will have to be provided if you want to decrypt or even delete files so only users that have shared access that are able to perform this function without the key file cannot be decrypted or even deleted. This can be shown in Fig. 10.

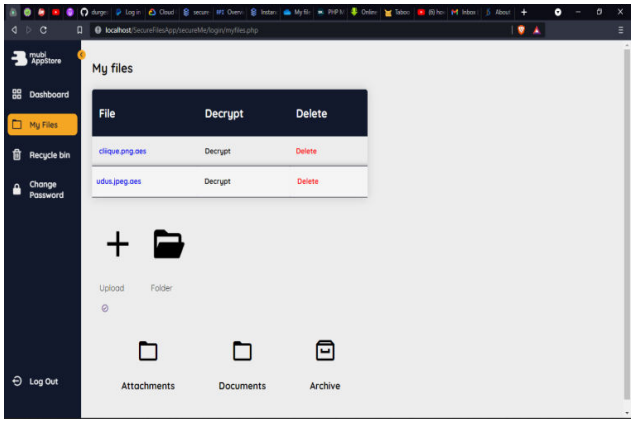

FIGURE 10. - My Files Page

E. *UPLOAD PAGE*

The upload page does the file encryption part just like how in the previous page which is the My files page deals with decryption. This is done when the user clicks on the upload button and then selects the file they want to encrypt after that the user is now prompted to enter a key to complete file encryption (i.e key that will be used to decrypt and download file on the My file page). The key is hashed and saved in the database and uploaded together with the encrypted file on the cloud. File is then stored on the GCP bucket storage with the extension ".aes". Shortly after it is done it shows up on the page with the status encrypted so that users know that file are indeed encrypted, also an option to mail key to shared users is provided. This sends shared access key information and file id to the shared users to enable them to download and decrypt files. This can be shown in Fig. 11.

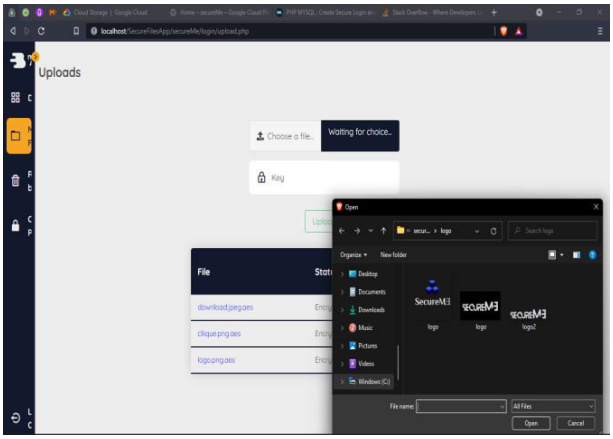

FIGURE 11. - Upload Page

F. *FILE ENCRYPTION AND DECRYPTION*

Files are encrypted using two different cryptography methods which are Advance Encryption Cipher and MCrypt Methods in PHP same goes for file decryption. In this project file encryption and decryption library is used here. Function used is the `open_ssl_file_encryption` and `open_ssl_file_dencryption`. This can be shown in Fig. 12.

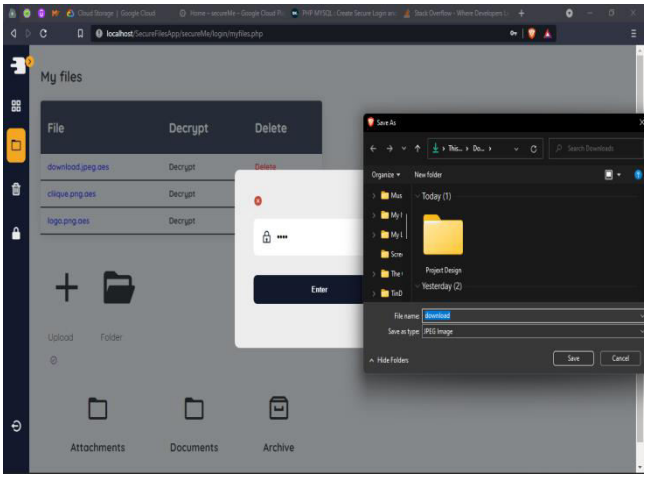

FIGURE 12. - File Decryption

G. *FILE STORAGE ON CLOUD AND DOWNLOAD*

File encrypted data is stored on the cloud platform onto a storage bucket. The storage buckets are storage location on the cloud provided by GCP as one their services in order to access this storage bucket we need a json file that references to the storage bucket credentials (username, password, id and other info) that's how the bucket is setup, then we used an "uploadObject" function provided by Cloud Client libraries to upload and store files on bucket location. The download part also used a function from library "downloadObject" to download specific files from the cloud. Definitely it does some key validation first before it fetches and downloads locally on the user device. This can be shown in Fig. 13.

Volume: 13 Issue: 05 Pages: 5143-5151(2022) ISSN: 0975-0290

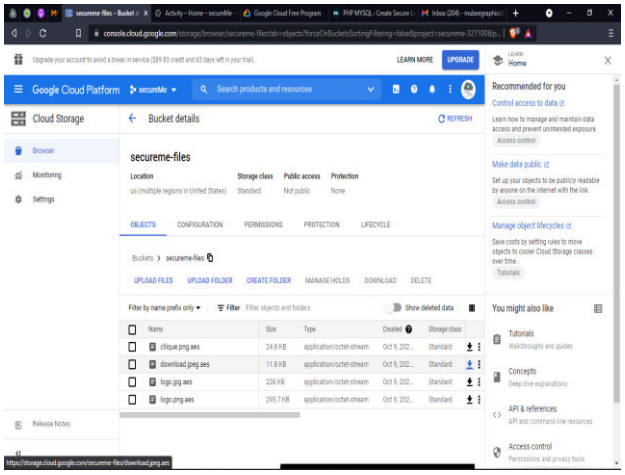

FIGURE 13. - Cloud Bucket

VI. COMPARISON OF PROPOSED SECURE VIRTUAL FILE STORAGE ON CLOUD WITH SIMILAR WORKS

This section compares the proposed secure virtual file storage on cloud with other reviewed works. The comparison is presented in tabular form as follows:

TABLE 1: Comparison of proposed system with some related works

System/Scheme	Purpose	Cryptographic Method(s)
Chinnasamy et al. (2021)	New hybrid method to achieve high data security and confidentiality in cloud computing environment	ECC and Blowfish to implement a hybrid algorithm
Viswanath & Krishna (2021)	An encryption algorithm for storing big data in the multi cloud storage for restricting the insider attacks	The framework consists of data uploading, slicing, indexing, encryption, distribution, decryption, retrieval and merging process
Karati et al. (2021)	A secure file storage and access protocol for cloud-enabled Internet of Things environment	Utilize the functionalities of the bilinear pairing operation
Sheeja et al. (2020)	New secure cloud storage system to ensure that the details of companies are secure from all data operator and third-party inspector, respectively	Utilize RSA and AES calculation to keep up security inside the cloud server
Rashmi et al. (2020)	An enhanced RDPC protocol for cloud storage is which checks the data integrity	Based on homomorphic hash algorithm
Abdel-Kader et al. (2020)	An efficient two-stage cryptography scheme to access and store data into cloud securely	A two-factor authentication scheme one-time password is proposed
Kumar & Shafi, (2020)	An effective mechanism with a distinctive feature of data integrity and privacy in order to guarantee that the user's data in the cloud are secure	Modified RSA algorithm to provide enhanced security for the data stored in the cloud
Seth et al. (2020)	Implementation of a fresh architecture that can convey an enhanced level of security for outsourcing information in a cloud computing environment while involving several independent cloud providers	Framework consists of dual encryption and data fragmentation techniques that visualize the secure distribution of information in a multi-cloud environment
Sharma et al. (2020)	Employment of a Blockchain to provide secure administration and investigation of health services enormous information that rely on their conveyed nature and other extra protection and security properties of the system and distributed storage	Utilize data security to ensure better security of structure
Sharma et al. (2020B)	Improve the security of information or data file on cloud	Employ of hybrid encryption with crossbreed cryptographic calculations

Volume: 13 Issue: 05 Pages: 5143-5151(2022) ISSN: 0975-0290

Proposed System	Designed and implemented virtual secure file storage system on cloud using hybrid cryptography	The cryptography method used for file encryption and decryption is AES and SHA-2 hash function

VII. CONCLUSION

Cloud Computing utilizes network of huge amount of servers usually running lower cost customer PC technology with unusual connections to perform data processing tasks across end users. Cloud storage service avoids the cost pricey on software, personnel maintenance and ensures better performance, less storage cost and scalability. Dependable file storage and access establish several security issues in a cloud computing.

REFERENCES

[1] Bindu, B. S., & Yadaiah, B. (2011). Secure data storage in cloud computing. *International Journal of Research in Computer Science*, *1*(1), 63-73.

[2] Rajathi, A., & Saravanan, N. (2013). A survey on secure storage in cloud computing. *Indian Journal of Science and technology*, *6*(4), 4396-4401.

[3] Karati, A., Amin, R., Mohit, P., Sureshkumar, V., & Biswas, G. P. (2021). Design of a secure file storage and access protocol for cloud-enabled Internet of Things environment. *Computers & Electrical Engineering*, *94*, 107298.

[4] Suresh, S. R. (2021). An Electronic Digital Library Using Integrated Security Methods and Cloud Storages. *International Journal of Advanced Networking and Applications*, *13*(1), 4839-4844.

[5] Olanrewaju, O., Oluwatoyin, A. A., & Mary, O. T. C. (2020). Secure Online Electronic Civil Registration Using Cloud Computing: A Conceptual Framework. *International Journal of Advanced Networking and Applications*, *11*(5), 4418-4422.

[6] Popa, R. A., Lorch, J. R., Molnar, D., Wang, H. J., & Zhuang, L. (2011, June). Enabling Security in Cloud Storage SLAs with CloudProof. In *USENIX Annual Technical Conference* (Vol. 242, pp. 355-368).

[7] Kute, S., & Javheri, S. B. (2018, August). Implementation of Secure File Storage on Cloud with Owner-Defined Attributes for Encryption. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)* (pp. 1-6). IEEE.

[8] Swarna, C., & Eastaff, M. S. (2018). Secure File Storage in Cloud Computing Using Hybrid Cryptography Algorithm. *Iaetsd Journal for Advanced Research in Applied Science*.

[9] Poduval, A., Doke, A., Nemade, H., & Nikam, R. (2019). Secure File Storage on Cloud using Hybrid Cryptography. *International Journal of Computer Science and Engineering*, *7*.

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

[11] Wu, J., Li, Y., Wang, T., & Ding, Y. (2019). CPDA: A confidentiality-preserving deduplication cloud storage with public cloud auditing. *IEEE Access*, *7*, 160482-160497.

[12] Mohammed, N., & Ibrahim, N. (2019, March). Implementation of new secure encryption technique for cloud computing. In *2019 International Conference on Computing and Information Science and Technology and Their Applications (ICCISTA)* (pp. 1-5). IEEE.

[13] Fan, Y., Lin, X., Liang, W., Tan, G., & Nanda, P. (2019). A secure privacy preserving deduplication scheme for cloud computing. *Future Generation Computer Systems*, *101*, 127-135.

[14] Sharma, S., Singla, K., Rathee, G., & Saini, H. (2020B). A hybrid cryptographic technique for file storage mechanism over cloud. In *First international conference on sustainable technologies for computational intelligence* (pp. 241-256). Springer, Singapore.

[15] Sharma, S., Mishra, A., & Singhai, D. (2020, April). Secure cloud storage architecture for digital medical record in cloud environment using blockchain. In *Proceedings of the International Conference on Innovative Computing & Communications (ICICC)*.

[16] Seth, B., Dalal, S., Jaglan, V., Le, D. N., Mohan, S., & Srivastava, G. (2020). Integrating encryption techniques for secure data storage in the cloud. *Transactions on Emerging Telecommunications Technologies*, e4108.

[17] Kumar, Y. K., & Shafi, R. M. (2020). An efficient and secure data storage in cloud computing using modified RSA public key cryptosystem. *International Journal of Electrical and Computer Engineering*, *10*(1), 530.

[18] Abdel-Kader, R. F., El-Sherif, S. H., & Rizk, R. Y. (2020). Efficient two-stage cryptography scheme for secure distributed data storage in cloud computing. *International Journal of Electrical & Computer Engineering (2088-8708)*, *10*(3).

[19] Rashmi, R. P., Gandhi, Y., Sarmalkar, V., Pund, P., & Khetani, V. (2020, October). RDPC: Secure Cloud Storage with Deduplication Technique. In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)* (pp. 1280-1283). IEEE.

[20] Sheeja, R., Bibin, C., Krishnan, P. R., Nishanth, R., Gopinath, S., & Ashok, K. G. (2020, August). Secure File Sharing System in Cloud Using AES and Time Stamping Algorithms. In *IOP Conference Series: Materials Science and Engineering* (Vol. 906, No. 1, p. 012023). IOP Publishing.

[21] Viswanath, G., & Krishna, P. V. (2021). Hybrid encryption framework for securing big data storage in multi-cloud environment. *Evolutionary Intelligence*, *14*(2), 691-698.

[22] Chinnasamy, P., Padmavathi, S., Swathy, R., & Rakesh, S. (2021). Efficient Data Security Using Hybrid Cryptography on Cloud Computing.

Volume: 13 Issue: 05 Pages: 5143-5151(2022) ISSN: 0975-0290

In *Inventive Communication and Computational Technologies* (pp. 537-547). Springer, Singapore.

[23] Khan, M., & Munir, N. (2019). A novel image encryption technique based on generalized advanced encryption standard based on field of any characteristic. *Wireless personal communications*, *109*(2), 849-867.

[24] Riaz, M. N., & Ikram, A. (2018). Development of a secure SMS application using advanced encryption standard (AES) on android platform. *Int. J. Math. Sci. Comput.(IJMSC)*, *4*(2), 34-48.

[25] Zhang, X., & Wang, X. (2018). Remote-sensing image encryption algorithm using the advanced encryption standard. *Applied Sciences*, *8*(9), 1540.

[26] Daoud, L., Hussein, F., & Rafla, N. (2019). Optimization of advanced encryption standard (AES) using vivado high level synthesis (HLS).

[27] Buhari, B. A., Obiniyi, A. A., Sunday, K., & Shehu, S. (2019). Performance Evaluation of Symmetric Data Encryption Algorithms: AES and Blowfish.

[28] Martino, R., & Cilardo, A. (2019). A flexible framework for exploring, evaluating, and comparing SHA-2 designs. *IEEE Access*, *7*, 72443-72456.

[29] Al-Odat, Z., Abbas, A., & Khan, S. U. (2019, December). Randomness Analyses of the Secure Hash Algorithms, SHA-1, SHA-2 and Modified SHA. In *2019 International Conference on Frontiers of Information Technology (FIT)* (pp. 316-3165). IEEE.