IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

CRYPTOVAULT

[1]Mrudul Arvind Katti, [2]Shailja Bajaj, [3]Rashi Padroo, [4]Ojaswi Athghara

[1]Student, [2]Student, [3]Student, [4]Student
[1]Computer Science and Engineering,
[1]The National Institute of Engineering, Mysuru, India

Abstract: There is a lack of a secure storage solution for users today. The digital vault solution, called 'CryptoVault,' is designed to provide a highly secure storage capability to its users. The solution uses file encryption, file fragmentation, and cloud storage to ensure utmost data security. The users can access and store the documents with the provision of a user-friendly and interactive interface. TLS is used to ensure secure data transmission over the webserver. This work aims to provide the desired document storage for users to store their important documents.

Index Terms **- Advanced Encryption Standard (AES), Transport Layer Security (TLS), One Time Password (OTP), Secure Sockets Layer (SSL), Secure Hash Algorithm (SHA), Unique Identity Number (UIN).**

I. INTRODUCTION

People have a series of essential documents in today's world, including birth certificates, passports, marks cards, and drivers' licenses. Due to the copious number of documents, people are highly prone to forgetting or misplacing important ones when required. This problem presented the need for a secure virtual vault to maintain these documents, ensuring that people do not lose their documents and always have access. During the inception phase, a survey on similar existing systems revealed that they lack the desired level of security in maintaining essential documents.

Existing applications have several drawbacks, leading to less secure systems. The work presented in this paper aims at overcoming the shortcomings of the current systems by providing a safe and secure platform for everyone to store their important documents.

CryptoVault achieves this with the use of top-notch login security and the provision of a two-factor authentication system which adds an extra layer of protection, allowing only the intended user to login to their respective account. The Advanced Encryption Standard algorithm (with 128-bit key) is used to encrypt data and documents, and Transport Layer Security for secure transfer of data through web servers. CryptoVault also uses the Secure Hashing Algorithm (SHA) to ensure data integrity.

To store the encrypted data, CryptoVault uses cloud storage, which offers backup and security, hence protecting against ransomware. Therefore, it becomes more difficult to access the system without proper authentication. Cloud storage also provides other benefits like flexibility, scalability, and affordability, giving it an upper hand over other forms of storage.

In few renowned existing applications, encryption costs are very high, making it inaccessible to the ordinary person, whereas CryptoVault additionally provides a free encryption service for all documents.

II. EXISTING SYSTEM AND DRAWBACKS

Following are the drawbacks of the existing products:

- Available only to the users with a Unique Identity Number. [1][2]
- Time taken to access the stored documents is high. [1]
- UIN and OTP (One Time Password) enabled authentication are not entirely secure. In many cases, illegal access of OTPs happen, and UINs are generally easy to determine since users are required to submit the UIN documents to a lot of organizations, including companies, colleges, schools, etc. [1][2]
- Additionally, the documents stored are not themselves encrypted before storing them in the cloud. Data can easily be stolen or misused from online repositories, which would release thousands of personal records of the individuals online. [1][2]
- Authentication of documents uses digital signatures, using e-KYC based authentication, which is not highly secure [8].
- The absence of two-factor authentication poses another problem in few existing systems since getting access to the users' passwords can potentially mean data theft, a significant security threat [3][4].
- The provision of file encryption, if present, is given at a very high price (nearly 200 USD) [9].
- Data Privacy is the biggest concern; everything could be at risk without the addition of other defenses [5].
- The customer support team of the current systems can be slow to respond and hard to reach [9].
- One of the existing systems facilitates data sharing by operating in a Certificate Authority capacity. In this capacity, the system adopts the role of both certificate issuer and certificate authorizer (as signified in the Public-Key Infrastructure (PKI) scheme), enabling them to view user data, violating their 100% data confidentiality policy [6].
- Systematic evaluation reveals a serious breach to the security architecture of another current system: showing that the whole security of the product relies on the trust in the product, independent of trusting Azure [7].

III. PROPOSED METHOD BENEFITS

1. Use of two-factor authentication, leading to enhanced security.
2. Free of cost.
3. Use of Advanced Encryption Standard (AES) algorithm for encryption, one of the most secure encryption algorithms for big files.
4. Use of cloud storage leading to decreased latency, high scalability, and reliability, and affordability.
5. Use of fragmentation before uploading onto the cloud to improve security against unauthorized access.
6. Use of Transport Layer Security for secure transfer of data through web servers. TLS offers more robust security, increased performance, and better protection than currently adopted encryption methods.
7. Provision of an interactive, attractive, and up-to-date user interface.

IV. SYSTEM ARCHITECTURE AND MODULES

The solution is divided into four modules, whose functionality is as mentioned below:

Registration and Login Module: The Registration and Login module is provided to access the system initially. The users accessing the system for the first time will be asked to register themselves to the system by providing basic information like name, contact number, address, etc. Following that, the module will return a unique user I.D. and password. For subsequent access to the system, the users can directly log in by entering the user I.D. and the password mentioned above.

Fragmentation Module: This module is responsible for fragmenting the files as desired by the user and storing them in the users' local system. The users must choose the file to fragment from their computer. The user must plan the fragmentation by giving each part's pathname and size. The parts are stored in the assigned folder along with a merger file. The merger file contains the number of bytes in each component and the start and end bytes.

Upload File Module: This module handles the encryption of the fragments in addition to the upload onto the cloud itself. Here, the 'Cloud Replication' option on the main screen gives the users access to the module. Subsequently, the users must first select the local path to the stored pieces of the file, following which the pieces are loaded into the module. Next, the encryption algorithm is applied, and the metadata XML file is created to aid future retrieval of the file. Finally, the module uploads the file fragments onto the cloud (i.e., AWS) server.

Download File Module: The users must click on the 'Get Files from Cloud' option on the main screen. The download begins once the module gets access to the metadata file that contains the encryption key. Then a list of the parts is displayed for user reference. The next step is applying the decryption algorithm to decrypt the individual parts, and on merging them, the whole file gets downloaded with a filename having the prefix "download."

V. FLOW OF SEQUENCE

The flow of execution of the project is as follows (written concerning the below figure):

(1) The user enters his name, mobile, email I.D., and address for registration. These details are stored in the local database. Upon registration, a unique user I.D. and password are returned.

(2) The user can then use the user I.D. and password mentioned above to log in to the application in the future successfully.

(3) To upload any document, image, or file (file size must be below 800 MB), the files must first be split into fragments. To do so, the user selects the filename and path of the file to fragment on entering the fragmentation module in the displayed prompt. Then, the number of fragments is entered (>2), and the path to store the fragmented files is defined.

(4) The user must then define the size of each fragment (always less than the remaining file size), after which the fragment is added into the users' local system. This process is repeated until the entire file has been fragmented as required.

(5) The user must first ensure proper internet connectivity and then browse the path of the file fragments to apply encryption. Once the module has extracted all the fragments, the user must click on the 'Apply Security' icon to thus generate the metadata.

(6) Ensuing metadata generation, the user will have to provide the XML file name with the .xml extension and click on the 'Upload' icon to upload it to the cloud. Once the uploading process is over, the module returns an acknowledgment. The user can later view the XML file.

(7) To retrieve the uploaded file from the cloud, the user must first choose the XML file.

(8) The metadata (including the file details) is then extracted from the XML file.

(9) Now, the user must click on the 'Download' icon to decrypt, merge and download the file, and then specify the path to store the downloaded file.

(10) To view the uploaded file details, the user must enter the module.

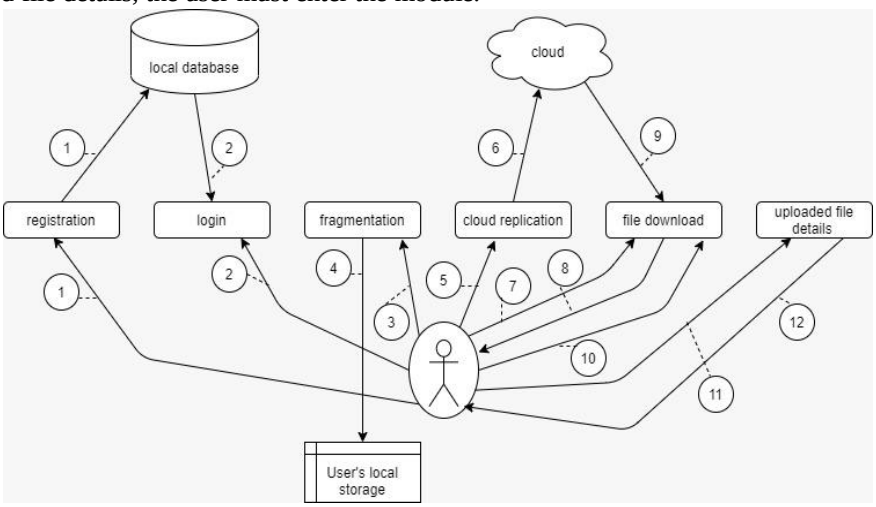

Fig. 1. Workflow sequence

(11) The module will then show a table with files' details, like serial number, document name, upload date, etc.
(12) To exit the application, the user must then click on the "exit" icon.

VI. CONCLUSION

The work presented in the paper is a secure storage solution called the 'CryptoVault.' This solution aims at providing its users a reliable and user-friendly storage capability. It employs strong authentication, encryption, other security, and integration facilities, making it a prospect for future use. The solution aims to alleviate the different problems that people face concerning their document usage and storage. The premise of it can be extended in the future, further improving the product's promise in the market.

VII. FUTURE ENHANCEMENTS

There are several opportunities in terms of improvement of the given solution to make the user experience better. Currently, the maximum file size is 800 MB, which can be made higher. The solution uses a single server and multiple folders to store the different fragments of a file, which can be made more robust with the addition of 2 or more servers to enable storage of the various fragments on other servers. The degree of automation in the modules can be increased to alleviate the users from major tasks, and finally, the user interface can be made more attractive to provide a more seamless experience to the users.

REFERENCES

[1] Dr. Vinay Kumar (VIPS/IT), Ms. Arpana Chaturvedi, and Dr. Meenu Dave (JIMS/IT) 2018. A Solution to Secure Personal Data When Aadhaar is linked with DigiLocker. MECS, 5, 37-44.
[2] Merlin Ann George (M.phil Scholar, Sacred Heart Autonomous College) and Dr. A M Viswambharam (Associate Professor, Sacred Heart Autonomous College) 2019. DigiLocker - an Overview. IJRAR, VOLUME 6.
[3] Nur Hayati Ahmad, Ameerah Saeedatus Syaheerah Abdul Hamid, Nur Solehah Sorfina Shahidan, and Khairul Akram Zainol Ariffin, 2020. Cloud Forensic Analysis on pCloud. EAI International Conference, iCETiC, 19–20.
[4] Dimitris Papadias, Spiridon Bakiras and Stavros Papadopoulos 2021. pCloud: A Distributed System for Practical PIR. IEEE, VOLUME 9.
[5] M. S. Kavitha and P. Damodharan 2021. pCloud implementing SaaS in distributed system. IEEE, VOLUME 9.
[6] Duane C. Wilson and Giuseppe Ateniese 2014. To Share or Not to Share in Client-Side Encrypted Clouds. ISC, Part of the Lecture Notes in Computer Science book series (LNCS, volume 8783).
[7] Martin Grothe, Christian Mainka, Paul Rosler, Johanna Jupke, Jan Kaiser, Jorg Schwenk 2016. Your Cloud in My Company: Modern Rights Management Services Revisited. IEEE, DOI 10.1109/ARES.2016.69.
[8] The official DigiLocker website. digilocker.gov.in.
[9] The official pCloud website. pcloud.com.