A Framework for Encryption and Authentication Cryptosystem (EAC) for Shared Cloud Storage

Saurabh Singhal,GL*A University,* Mathura, India
Karan Balani, B.Tech 4th Year – CSE, GL*A University,* Mathura, India

Dhruv Baslas, B.Tech 4th Year – CSE, GL*A University,* Mathura, India
Tanya Gangwar,B.Tech 4th Year – CSE, GL*A University,* Mathura, India

Gayatri Gupta,B.Tech 4th Year – CSE, GL*A University,* Mathura, India
Apoorva Yadav,B.Tech 4th Year – CSE, GL*A University,* Mathura, India

Article Info
Volume 83
Page Number: 10553 - 10557
Publication Issue:
May - June 2020

Article History
Article Received*: 19 November 2019*
Revised*: 27 January 2020*
Accepted*: 24 February 2020*
Publication*: 18 May 2020*

Abstract:
The cryptosystem provides a secure way to share files among other users without worrying about the size or number of files and the leakage of data in between of transit that underpins the adaptable assignment of reduced rights for users and data. The key-size for encryption or decryption is autonomous of the number of data or files such that the cost of our plot is steady no matter how habitually clients transfer records to the cloud server. In expansion, the verification prepare in our plot tackles the leakage issue of information sharing. The cloud server gets the token of download-applicant and after that controls download right. This paper proposes a framework by which the data can be shared with users seamlessly without worrying about security. To realize proficient and secure information sharing in energetic cloud capacity, the proposed strategy ought to be steady in cost, and ought to be leakage-resilient.

Keywords*: cloud, EAC, Encryption, Decryption.*

I. INTRODUCTION

Information leakage could be an authentic issue adjusting the commerce, commercial and individual grounds [9]. This paper is proposing an arrangement to not as it was get freed of that issue but moreover, a cover layer for solid credibility of if i.e. in case information leakage happens it won't be an issue at all. Even though there is overwhelming and compelling calculations input for that reason, the end-users are not so beyond any doubt they know what happens at backend driving to gaps of understanding. One can get it this approach with an unimportant intrigued in innovation and it is forthright fetched is much less than the standard procedures. The issue of information leakage is getting to be more genuine in social organize applications presently [8]. Cloud servers may be inquisitive so that private data on cloud servers should be encrypted on our own. Clients often scramble their private information before uploading. Information sharing through the cloud securely and effectively gets to be troublesome for cryptosystem creators, to form a practical application for secure and proficient information sharing in the cloud. Subsequently by proposing a novel concept of encryption and verification cryptosystem. The conspire assumes proficient(e.g. little key estimate) and secure (e.g. leakage-resilient) information sharing through energetic cloud capacity with compact keys. The objective of this framework is to plan and execute a system framework that can unravel issues like information leakage from cloud capacity or cloud overflow. Data leakage is the transit of internal data of an organization to an external person or group which can misuse it for their benefits. Data leakage can also be done via exchanging data digitally or physically. Data or Information leakage usually happen through the internet, public servers and e-mail, but can also happen through data storage devices such as Hard Disks, USB Drives, and portable computers. A cloud overflow is when hazardous trade information put away in a private cloud or open cloud occasion is incidentally exposed to the web [10]. To avoid cloud overflow, we are going be proposing a calculation that can work with cloud capacity to guarantee that indeed after the leakage of information the data will be blended and will not result in any misfortune for the clients of that cloud.

II. Related Approaches

Encryption may be a preparation of turning information into a non-understandable frame to keep it secure in case of information breaches and information loss. Encryption of information that is hazardous and critical to different sorts of businesses that can be in shape of anything (content, pictures, recordings, other media, and records, etc.) to decrease the overall risk of information breaches from the stock. It'll diminish the chances of the utilization of information in a scrambled frame and will make it less demanding for benefit suppliers to share information as per require more securely and centering more on other aspects of security.Maintaining the Integrity of the Specifications In [2], to derive the keys for the lower level in a hierarchy from a higher level, the authors created a rigid hierarchy. In [3] the authors proposed a model which was flexible inthe hierarchical requirements. The main purpose of these schemes was to minimize the expense of storing data and thus the keys were manage as per the predefined hierarchies. In [4] the authors describes an extended key-aggregate cryptosystem (KAC) using cloud storage for frequently updating the ciphertext. In this if the ciphertext is to be classified in more than n classes, then additional pairs of keys may be registered by the data owner.

To efficiently manage the encryption keys, the authors in [5], [6] focused on design of such encryption schemes. A key-aggregate searchable encryption was proposed by them. In this scheme, the user who want to download the file from multiple users can select files by performing keyword search.

For attaining security and reliability to access file control, the authors [7] proposed Attribute-Based Encryption in the cloud storage scheme. This approach offers Cipher-text Policy ABE (CPABE) framework for the cloud storage scheme. To decrease the computational cost of CPABE, the authors without disclosing secret keys and file contents outsourced high-level computational load to a cloud service operator cost for the local device.

A. Disadvantages

1) Time and Taken a toll Costly: : Encryption of information depends on the setup of the framework which chooses how more grounded encryption you'll prepare on your system which is fetched costly and it'll be an overwhelming errand to do in case of a huge sum of information which can be time expensive.

2) Security: : We possess a framework that may well be defenceless to handle information for encryption due to Infections, Trojans and other sorts of security vulnerabilities.

3) Sharing of Information: : The sharing of information is hazardous since of burglary and time and taken a toll costly unscrambling of information by the clients you shared it with. We require a centralized framework for it. Information security could be an exceptionally imperative issue and conventional IT way to guarantee or give it is to depend on the nearby frameworks and server to enforce the control rules after confirmation, which suggests any blemish and startling benefit acceleration in this framework will uncover all information and can permit information breaches and information robbery. In a dynamically modern shared-tenancy cloud computing environment,

things ended up indeed more awful since of different viewpoints such as accessibility of records, Cloud Spill - a cloud spill is when delicate commerce information put away in a private cloud or open cloud occasion is incidentally uncovered to the web. Moreover, cloud clients likely will not hold the solid trust on providers that the cloud server is doing a great job in terms of secrecy. Local Encryption Local Encryption is a straightforward however imperfect way to keep the information secure in this we add our claim layer of encryption which is done on our possess framework which may be an unsafe and time-costly assignment.

B. Disadvantages

1) Complexity: : The fetched and complexities of this are exceptionally tall and the number of keys included in encryption and decoding of information is subordinate on the number of records in cloud framework which can for the most part increment quickly and make the framework time costly and can too influence promptly accessibility of data. 2) Splillage: : Another issue is related to Spillage of Information from Cloud Servers.

C. Secure Storage and Access of Data in Cloud Computing [1] According to this approach the data in cloud is separated into two parts in cloud. One is Private Section where sensitive data of users is stored and can only be read by trusted users and other refers to Public Section where less sensitive and publically accessible data is stored. Data in both sections are encrypted and needs to be decrypted at the time of download.

III. Proposed Approach

A. *Problem Statement*

Public Cloud Storages are very convenient to use and solve various issues however they are having worries around Data Security and Data Privacy attached to them. For instance, “Alice has multiple files which she wants to encrypt and keep on a cloud storage and then she aspires to share few of them with her accomplice Bob but the obstacle with this state of operations is both she encrypts all documents with an equal key and offers the key to Bob but this could poses a risk to Alice records privacy because Bob can see other documents as accurately which she does not need Bob to see because of encryption using identical key or Alice uploads files encrypted with extraordinary and different keys but this is a complete time luxurious challenge because the quantity of key will increase as quantity of documents and sharing of different keys for special documents is a tedious and not appropriate assignment. Also, the keys are prone to theft that’s why we require a stable channel to alternate the keys and share statistics with users as in line with their need.”

To summarize the purpose of our algorithm EAC can be stated as:

“Proposing an algorithm which can work with cloud storage to ensure security of data that even if in case leakage of data the information will be encrypted and will not result in any loss for the users of that cloud and data can also be shared with users which require that data without any hassles and also solving the problem of sharing multiple keys for multiple files.“

B. *The Framework of EAC – Encryption and Authentication Cryptosystem*

The information possessor will get the server together with their Setup Function and record/entryway will be enrolled as a client of Shared Cloud Storage. At that point the information possessor will transfer the information utilizing Upload Function which they need to store in Shared Cloud Storage, the information will be consequently blended utilizing the Symmetric Algorithm and afterwards, it will be put away in Storage. Different clients can demand the information utilizing Request Function on encoded documents to demand the entrance of records from information possessors. The solicitation will go to the information possessor’s entryway and they can give access to document utilizing Authorize Function. After getting the entrance of document at whatever point client will utilize Download Function the server will start formation of Secure Channel and will utilize Verify

Algorithm to confirm the two finishes client and server, on fruitful check server will begin decoding of the record utilizing Symmetric Algorithm and will send the mentioned document utilizing the protected channel to the client. As working is shown in the above figure Flow Diagram and mentioned above are 8 (eight) major steps in Encryption and Authentication Cryptosystem (EAC):

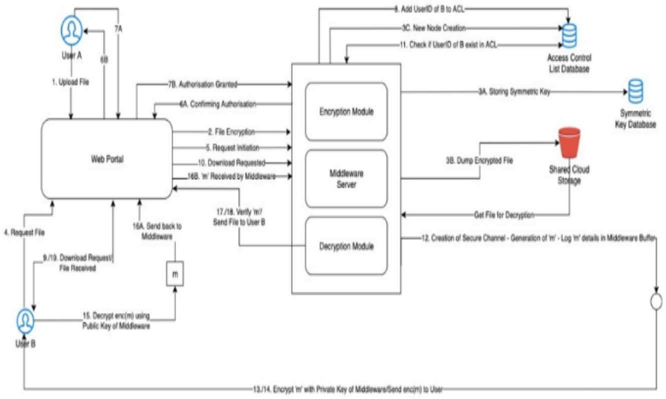
8. Add UserID of B to ACL
3C. New Node Creation
11. Check if UserID of B exist in ACL
Access Control List Database
User A
7A
6B
1. Upload File
7B. Authorisation Granted
6A. Confirming Authorisation
3A. Storing Symmetric Key
Symmetric Key Database
Encryption Module
Web Portal
2. File Encryption
5. Request Initiation
10. Download Requested
16B. 'm' Received by Middleware
Middleware Server
3B. Dump Encrypted File
Shared Cloud Storage
Get File for Decryption
4. Request File
16A. Send back to Middleware
17./18. Verify 'm'/ Send File to User B
Decryption Module
12. Creation of Secure Channel - Generation of 'm' - Log 'm' details in Middleware Buffer
9./19. Download Request/ File Received
m
15. Decrypt enc(m) using Public Key of Middleware
User B
13./14. Encrypt 'm' with Private Key of Middleware/Send enc(m) to User

Fig. 1. The Proposed Framework for EAC.

As working is shown in the above Flow Diagram and mentioned above are 8 (eight) major steps in Encryption and Authentication Cryptosystem (EAC):

- Setup: This is executed by the data owner to create their own portal to access the features of Shared Cloud Storage; it will take *Display Username, Security Key and Email Address* as input parameters to successfully create user's portal.
- Upload: This is executed by data owner who will use this function to select the files they want to store in Shared Cloud Storage and after selection, these files will be encrypted automatically using Symmetric Algorithm and after encryption files will be stored in Cloud Storage.
- Symmetric Algorithm: This algorithm will be triggered by the user's actions which will trigger the server to Encrypt or Decrypt the files uploaded or downloaded respectively using *Single Key Encryption Algorithm*. Example: AES, DES, etc.
- Request: This is executed by the user who wants to request some files of other users which are stored in Shared Cloud Storage; this will send the request to data owners for getting the access of those files.
- Authorize: This is executed by the user who owns the data; after getting request from the user for giving them access rights of the file, the user will have an option to Accept or Decline it.
- Download: This is executed by the user who requested the files; if the request by user is authorized by data owner then the user gets downloading functionality for the files which they have access to over a secure channel.
- Secure Channel: This is executed by server initiating creation of this channel and after verifying both ends i.e. the identity of the user who used *Download Function* for that file and server. If it is a successful verification, the server will decrypt the file and will use the secure channel to let the user download the decrypted file.
- Verify Algorithm: This is executed by server on the phase of file download to verify both ends for secure channel to pass the data requested to be downloaded; when server will get the request for downloading file the server will cross check the ID of user requesting download in file's Access Control List and then to verify the user server will send a random number which will be encrypted by servers private key and then encrypted random number will be sent to user, which will be decrypted by servers public key on client-side and sent back to server to match it for matching of random number which will verify both ends.

C. *Requirements*

EAC ensures and provides these edges and advantages:

- Sharing of Data: The data can be shared with multiple users at a time without un-necessary replication of data.
- Security: The data is stored encrypted on Cloud Storage and provided to a user who has rights to access them only, which maintains the privacy of data owners and provides security even in cases of breaches.
- Speed and Availability: The usage of cloud storage is an essential key to this to provide fast access to data anywhere and anytime to users.
- Compactness: EAC uses a constant sized token to verify users and server connection which does not depends on size or number of files making system less complex and lightweight to use and needs less computational power and time to execute.
- Authentication: The users are always authenticated before entering to Shared Cloud Storage and they all are bound to their limited privileges in it.

CONCLUSION:

The problem of data sharing with others has always had many limitations such as data leakage, type of encryption of data, amount of data, availability of data, theft of data, breaches, and many other costs and time-bound complexities and limitations.

Our proposed scheme EAC – Encryption and Authentication Cryptosystem in Shared Cloud Storage reduces the complexities and solves the limitations to provide a reliable and efficient solution of these flaws. Usage of cloud and cross-checking of privileges provided to individual users over data and providing simple and easy to use system to apply.

EAC can be used on various aspects such as “Data Sharing between team members working on different modules of a project” and, any other such scenarios.

REFERENCES

1. Kumar, Arjun, et al. ”Secure storage and access of data in cloud computing.” 2012 International Conference on ICT Convergence (ICTC). IEEE, 2012.
2. Guo, Cheng, et al. ”Key-aggregate authentication cryptosystem for data haringindynamiccloudstorage.”FutureGenerationComputerSystems 84 (2018): 190-199.
3. Akl, Selim G.,and PeterD. Taylor. ”Cryptographic solution toa problem of access control in a hierarchy.” ACM Transactions on Computer Systems (TOCS) 1.3 (1983): 239-248.
4. Chu, Cheng-Kang, et al. ”Key-aggregate cryptosystem for scalable data sharing in cloud storage.” IEEE transactions on parallel and distributed systems 25.2 (2013): 468-477.
5. Cui, Baojiang, Zheli Liu, and Lingyu Wang. ”Key-aggregate searchable encryption (KASE) for group data sharing via cloud storage.” IEEE Transactions on computers 65.8 (2015): 2374-2385.
6. Liu, Zheli, et al. ”Verifiable searchable encryption with aggregate keys for data sharing system.” Future Generation Computer Systems 78 (2018): 778-788.
7. Li, Jiguo, et al. ”Flexible and fine-grained attribute-based data storage in cloud computing.” IEEE Transactions on Services Computing 10.5 (2016): 785-796.
8. Chow, Sherman SM, et al. ”Spicesimple privacy-preserving identitymanagement for cloud environment.” International Conference on Applied Cryptography and Network Security. Springer, Berlin, Heidelberg, 2012.
9. Chow, Sherman SM, et al. ”Dynamic secure cloud storage with provenance.” Cryptography and security: From theory to applications. Springer, Berlin, Heidelberg, 2012. 442-464.
10. Goyal, Vipul, et al. ”Attribute-based encryption for fine-grained access control of encrypted data.” Proceedings of the 13th ACM conference on Computer and communications security. 2006.