A PARALLEL AND ASCENDABLE OF ERASURE WRITING SUPPORT IN CLOUD OBJECTIVE STORAGE SYSTEM

Mrs.B.Dhanalakshmi[1], M.Banupriya[2], T.C.Janani[3]

[1]Assistant professor, [2,3]Students
Department of CSE,
Jeppiaar SRR Engineering College, Chennai.

***Abstract*: Cloud computing is a concept that treats the resources on the internet as a unified entity, a cloud. The data is generally stored in virtualized pools of storage, hosted by third parties. A decentralized erasure code is suitable for use in a distributed cloud storage system. In this project we construct a secure cloud system that supports secure data storing system. Algorithms such as AES algorithm can be used. The data is uploaded by the user and encrypted using AES algorithm. For more secured purpose the random image registration is occurred. Files can be uploaded only when the user enter the correct image name. Uploaded files are split into different pieces and stored in different location using proxy re-encryption scheme. Using secret key the user can retrieve the data forwarding under this decryption form. In this reversible manner the data can retrieve by the data user.**

***Keywords*: AES algorithm, proxy re-encryption.**

I. INTRODUCTION

High-speed networks and ubiquitous Internet access become available to users for access anywhere at any time. Cloud computing is a concept that treats the resources on the Internet as a unified entity, a cloud. Cloud storage is a model of networked online storage where data is stored in virtualized pools of storage which are generally hosted by third parties. Hosting companies operate large data centers, and people who require their data to be hosted buy or lease storage capacity from them. It provides extra storage devices over the network such that a user can access the storage devices via network connection. Afterward, many improvements on scalability, robustness, efficiency, and security. A decentralized architecture for storage systems offers good scalability, because a storage server can join or leave without control of a central authority. To provide robustness against server failures, a simple method is to make replicas of each message and store them in different servers. However, the method is expensive as z replicas result in z times of expansion. A message is encoded as a codeword, which is a vector of symbols, and each storage server stores a codeword symbol.

A storage server failure is modeled as an erasure error of the stored codeword symbol. Random linear codes support distributed encoding, that is, each codeword symbol is independently computed. The system has a light data confidentiality because an attacker. In their system, stored messages are encrypted and then encoded. To retrieve a message, key servers query storage servers for the user.

II. EXISTING SYSTEM

In Existing System a straightforward integration method is used. In order to provide strong confidentiality for messages in storage servers, a user can encrypt messages by a cryptographic method before applying an erasure code method to encode and store messages. When user wants to use a message, needs to retrieve the Codeword symbols from storage servers, decode them, and then decrypt them by using cryptographic keys. General encryption schemes protect data confidentiality, but also limit the functionality of the storage system because a few operations are supported over encrypted data. A decentralized architecture for storage systems offers good scalability, because a storage server can join or leave without control of a central authority.

III. PROPOSED SYSTEM

In our proposed system we address the problem of forwarding data to another user by storage servers directly under the command of the data owner. We consider the system model that consists of distributed storage servers and secret key servers. These key servers are highly protected by security mechanisms.

For more secured based we proposed a random image registration scheme. Here Storage system has allocates by different data container using proxy re-encryption. Once owner uploads the data using AES the data can be encrypted and stored in different server location using proxy re-encryption scheme. In these secured storage system we used proxy re-encryption of distributed cloud storage. All the data pieces will be save in different location in cloud storage.

In these encryption data can be stored in different server and location. In these downloading occurred all files in original form using a 16bit secret key, it use to retrieve all information in decrypt form. When a proper data user asking the data, cloud system will provide the data in reversible manner. So our system will prevent our data from both Inside and Outside attackers

.

A. proxy re-encryption

Proxy re-encryption schemes are crypto systems which allow third parties (proxies) to alter a cipher text which has been encrypted for one user, so that it may be decrypted by another user. By using proxy re-encryption technique the encrypted data (cipher text) in the cloud is again altered by the user. It provides highly secured information stored in the cloud. Every user will have a public key and private key. Public key of every user is known to everyone but private key is known only the particular user.

By mistreatment proxy re-encryption technique the encrypted information (cipher text) within the cloud is once more altered by the user. It provides extremely secured data hold on within the cloud. Each user can have a public key and personal key. Public key of each user famous is understood is thought to everybody however non-public secret's known solely the actual user.

B. Architecture Diagram

The system model that consists of distributed storage servers and key servers. Figure1 In this user can register the information like username, password, mobile no, email-id. In these registration 16bit secret key can be created. Here notification can be received to this user email-id for security purpose using SMTP protocol. These key servers are highly protected by security mechanism. Since the security mechanisms are highly prohibited by random image registration. In this image solution are occurred the images to use upload files in secured way. Then uploading file by these data owner. Owner can upload the files and encrypt the files using AES algorithm. All the data can be split in piece will be saving in different location in cloud storage using proxy re-encryption. The distributed systems require independent servers to perform all operations.

In these downloading occurred all files in original form using a 16bit secret key, it use to retrieve all information in decrypt form. When a proper client asking the data, cloud system will provide the data in reversible manner. So our system will prevent our data from both Inside and Outside attackers.

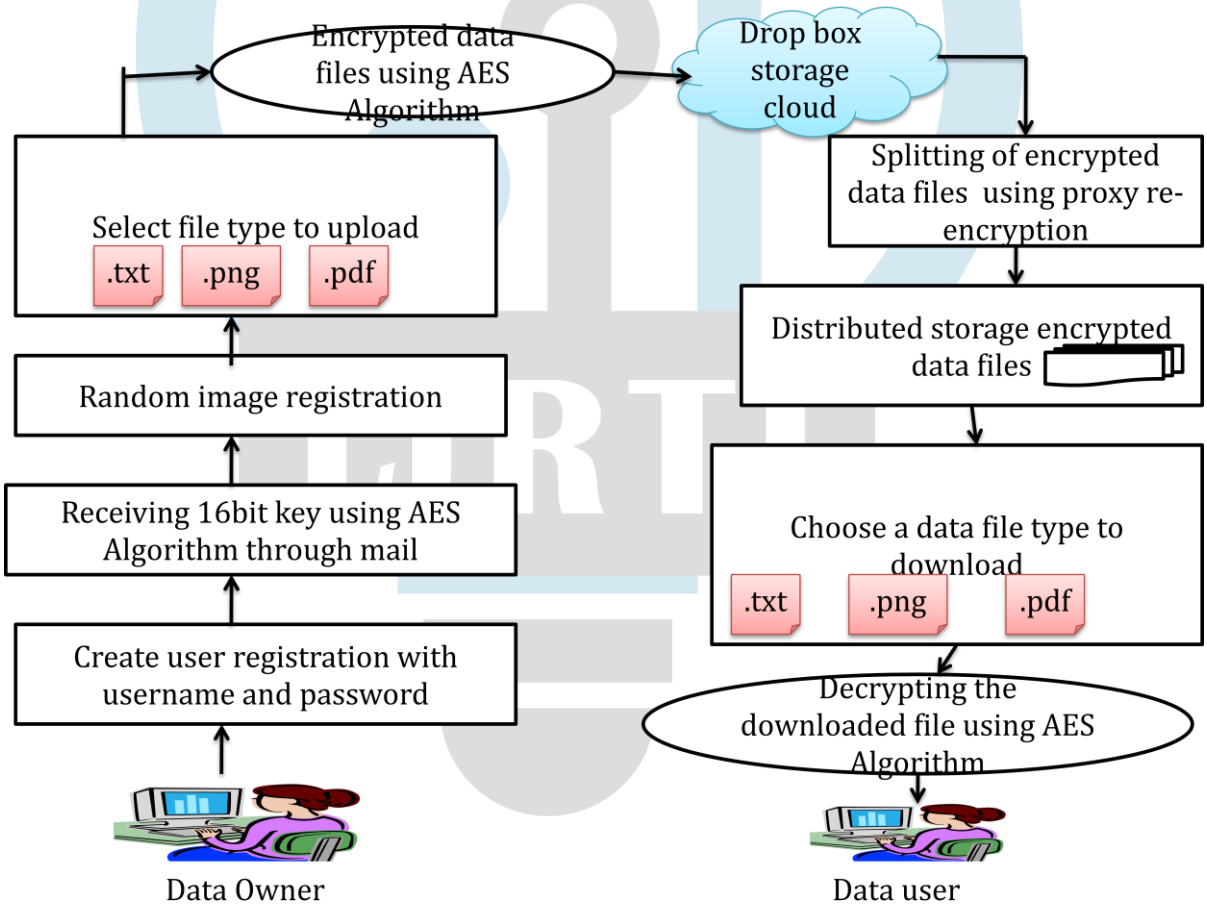

Figure 1. System architecture diagram

IV. MODULE DESCRIPTION

1. Authorization of data owner

2. Uploading and encrypting the data files in cloud
3. Splitting data by using proxy re-encryption
4. Decryption and retrieval of required data

1. Authorization of data owner

In cloud login module the user can login own details. Figure 2 The Registration process details are Username, Email, password, confirm password, mobile no and email-id.

After entering the registration process the details can be stored in database of the cloud system. figure 2 Then the user has to login to give his corrected username, password. Then the registration 16 bit secret key can be created by using AES algorithm. Then the user will go to open his account and view the code that can be generated from the cloud system. In these registration process user personal information are occurred. Here download process 16bit key is important hence the notification is send to this user email-id using under the SMTP protocol.

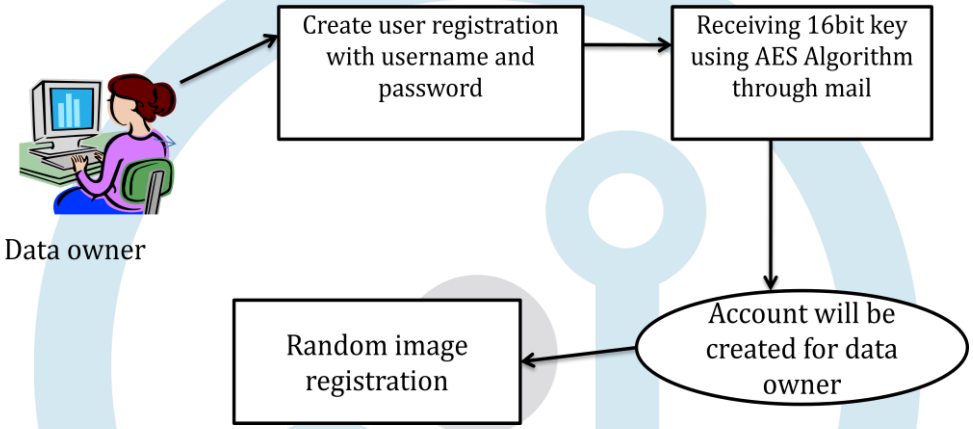

Figure 2. Authorization of data owner

A. Random image registration

In these user registered the information, then user receives a notification to these email-id for security purpose. For more security mechanisms highly protected by these images security purpose. During these users choose the images for security. Hence these images can randomly register during changes in upload files.

The conformation of username and password in these registrations. During this registration user personal details are included hence conform the username and password. The user only knows image file names hence it secured is high. During upload files, first image page is opened for secured purpose. Its helps to more secured the user information. All the user information is stored in database server.

2. Uploading and encrypting the data files in cloud

In this random image registration is occurred by user then user can upload the file. Figure 3Then uploading files can be encrypted by these AES algorithm. Encrypted data files a\can stored in different storage server by using these proxy re-encryption in these cloud storage system. Here Storage system has allocates by different data container. The schemes enable a content provider to share her data in a confidential and selective way, with a fixed and small cipher text expansion, by distributing to each authorized user a single and small aggregate key.

There have been many proposals of storing data over storage servers. One way to provide data robustness is to replicate a message such that each storage server stores a copy of the message.

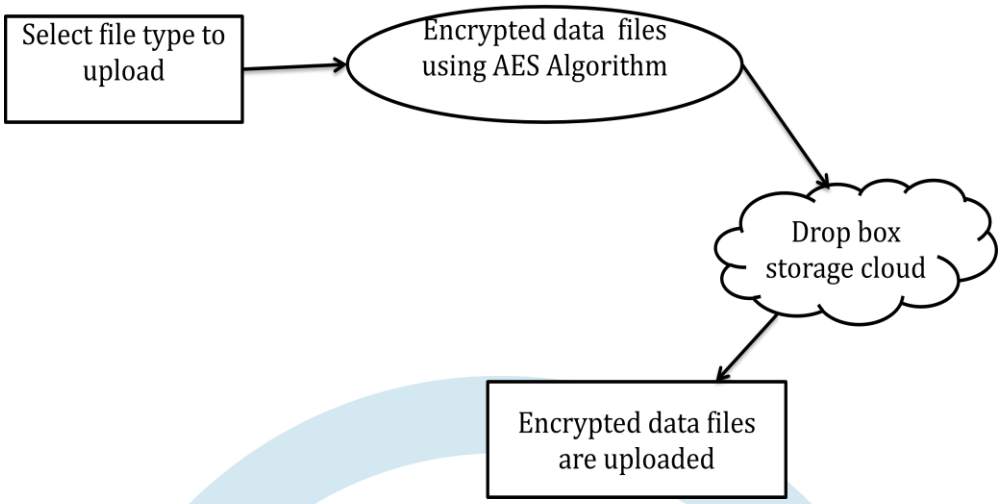

Figure 3. Uploading and encrypting the data files

3. Splitting data by using proxy re-encryption

Proxy re-encryption schemes are crypto systems which allow third parties (proxies) to alter a cipher text which has been encrypted for one user, so that it may be decrypted by another user. By using proxy re-encryption technique the encrypted data (cipher text) in the cloud is again altered by the user. Figure 4 It provides highly secured information stored in the cloud. Every user will have a public key and private key. Public key of every user is known to everyone but private key is known only the particular user.

In information theory, an erasure code is a forward error correction (FEC) code for the binary erasure channel, which transforms a message of k symbols into a longer message (code word) with n symbols such that the original message can be recovered from a subset of the n symbols. The fraction r = k/n is called the code rate, the fraction k'/k, where k' denotes the number of symbols required for recovery, is called reception efficiency.

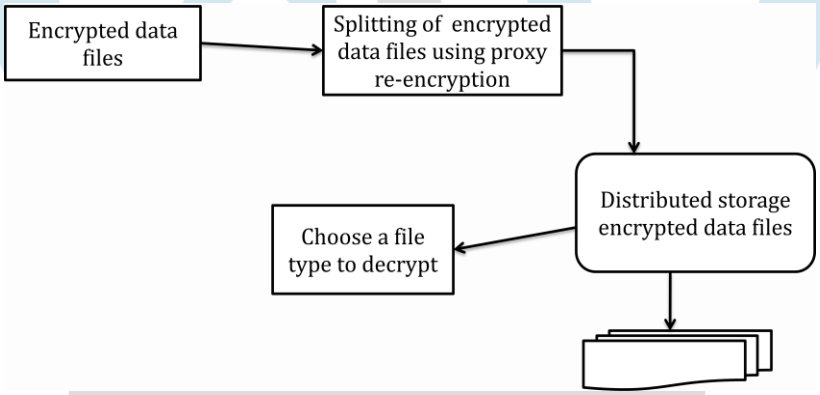

Figure 4. Splitting Data Files Using Proxy re-encryption

4. Decryption and retrieval of required data

There are some overlaps between them, but queries generally select a relatively small portion of the server, while reports show larger amounts of data. Queries also present the data in a standard format and usually display it on the monitor; whereas reports allow formatting of the output however you like and is normally retrieved. In Download module contains the following details. There are username and file name.

First, the server process can be run which means the server can be connected with its user. Now, the user has to download the file to download the file key. Figure 5 In file key downloading process the fields are username, filename, question, answer and the code. Now clicking the download option the client can view the encrypted key. Then using that key the client can view the file and use that file appropriately. Now enter the code which was sent by application admin while registration. This code is used to download the encrypted key file. Decrypt the encrypted file by giving encrypted key file as input and download the original file. Then the partial decryption takes place at multiple servers where pieces of data are stored. The algorithm is invoked to club all the pieces to get original file.

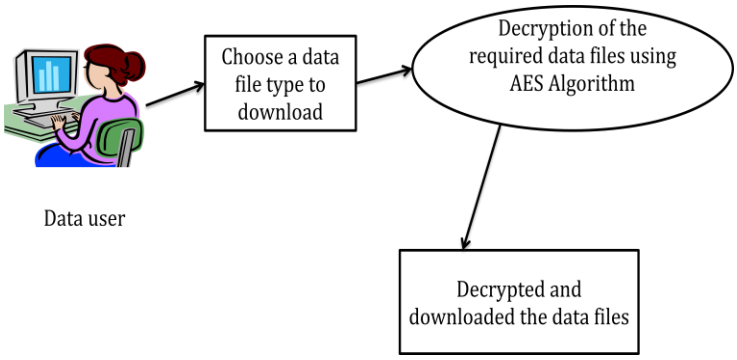

Figure 5. Decryption and retrieval of required data

V. ADVANTAGES

Tight integration of encoding, encryption, and forwarding makes the storage system efficiently meet the requirements of data robustness, data confidentiality, and data forwarding. The storage servers independently perform encoding and re-encryption process and the key servers independently perform partial decryption process. More flexible adjustment between the number of storage servers and robustness

VI. APPLICATIONS

It is used in places where documents and data's are required to be stored in a secured storage. This technique can be used in Banks, to store important files. It can also be used in military to secure the important files

VII. CONCLUSION

Erasure codes are promising for improving the reliability of the storage system due to its space efficiency compared to the replication methods. Traditional erasure codes split data into equalized data blocks and encode strips in different data blocks. This brings heavy repairing traffic when clients read parts of the data, since most strips read for repairing are not in the expected blocks. This paper proposes a novel discrete data dividing method to completely avoid this problem. The key idea is to encode strips from the same data block. We could see that for repairing failed blocks, the strips to be read are either in the same data block with corrupted strips or from the encoded strips. Therefore, no data is wasted. We design and implement this data layout into a HDFS-like storage system. Experiments over small-scale tested shows that the proposed discrete data divided method avoids downloading data blocks that are not needed for clients during the repairing operations. The traffic is reduced. The storage space is reduced. Since the files are spread across the cloud system it is difficult to be hacked, securing the file.

VIII. FUTURE ENHANCEMENTS

As a response, erasure coding as an alternative to backup has emerged as a method of protecting against drive failure. Raid just does not cut it in the age of high-capacity HDDs. The larger a disk's capacity, the greater the chance of bit error. And when a disk fails, the Raid rebuild process begins, during which there is no protection against a second (or third) mechanism failure. So not only has the risk of failure during normal operation grown with capacity, it is much higher during Raid rebuild, too. Also, rebuild times were once measured in minutes or hours, but disk transfer rates have not kept pace with the rate of disk capacity expansion, so large Raid rebuilds can now take days or even longer**.**

REFERENCES

[1] Huayu zhang, Hui li, member, IEEE, and Shuo-yen Robert li, fellow, IEEE transactions on reliability, vol. 64, no. 3, 2017

[2] Y. Fu, J. Shu, and X. Luo. A Stack-Based Single Disk Failure RecoverySchemefor ErasureCoded StorageSystems. InProceedings of IEEE SRDS'14, October, 2014.

[3] S. Xu, R. Li, P. Lee, Y. Zhu, L. Xiang, Y. Xu, and J. Lui. Single disk falure recovery for X-code-based parallel storage systems. IEEE Transaction on Computers, PP:99, January, 2013.

[4] X. Luo, and J. Shu. Load-Balanced Recovery Schemes for Singledisk Failure in Storage Systems with Any Erasure Code. In Proceedings of IEEE ICPP'13, October, 2013.

[5] O. Khan, R. Burns, J. Plank, and W. Pierce. Rethinking erasure codes for cloud file systems: minimizing I/O for recovery and degraded reads. In Proceedings of USENIX FAST'12, 2012.

[6] Y.Zhu, P. Lee, Y. Hu, L. Xiang, and Y. Xu. On the speedup of single-disk failure recovery in XOR-coded storage systems: theory and practice. In Proceeding of IEEE MSST'12, 2012

[7] B. Calder, J. Wang, A. Ogus, N. Nilakantan, A. Skjolsvold, S. Mckelvie, Y. Xu, S, Srivastav, J. Wu, H. Simitci, et al. Windows azure system: a highly available cloud storage service with strong consistency. In Proceedings of ACM SOSP'11, 2011

[8] L. Xiang, Y. Xu, J. Lui, and Q. Chang. Optimal recovery of single disk failures in RDP code storage systems. In Proceedings of the ACM SIGMETRICS'10, 2010.

[9] K. Greenan, X.Li, and J. Wylie. Flat XOR-Based Erasure Codes in Storage Systems: Constructions, Efficient Recovery, and Tradeoffs. In Proceedings of IEEE MSST'10, 2010.

[10] Z. Wang, A. Dimakis, and J.Bruck. Rebuilding for array codes in distributed storage systems. In IEEE GLOBALCOM Workshops, 2010.