Empowering Data Deduplication With User Dynamic Revocation In Cloud Storage

R.Gomathipriya[1], Dr.M.Nithya[2],Dr.K.Sasikala[3]

[1]*PG Scholar/Department of Computer Science & Engineering/VMKV Engineering College Salem/TamilNadu/India*

[2]*Professor/ Department of Computer Science & Engineering/ VMKV Engineering College Salem/TamilNadu/India*

[3]*Professor/ Department of Computer Science & Engineering/ VMKV Engineering College Salem/TamilNadu/India*

Abstract

Cloud computing is give a powerfully versatile assets provisioned as a benefit over the webpage. There are more cloud hubs for single cloud supplier. From the trusted specialist, the cloud hub gets mystery labels for record pieces so that the pieces can be prepared by the cloud hubs. Different designs are presented and examined concurring to their security capabilities and prospects. In this paper, empowering open review capacity for cloud is of basic significance so that clients can resort to a third party inspector (TPA) to check the judgment of outsourced information. A secure cloud capacity framework supporting Isolation-preserving open inspecting. It assist expands the result to empower the TPA to perform reviews for different clients at the same time and efficiently.This paper points at viably stores and recovers the records from the cloud space database server.

Keywords—*Cloud Computing, Muticloud, Integrity, Isolation Preserving Auditing, TPA.*

1.INTRODUCTION

Cloud computing creates a large number of security issues and challenges. A list of security threats to cloud computing is presented in [8]. These issues range from the required trust in the cloud provider and attacks on cloud interfaces to misusing the cloud services for attacks on other systems. The main problem that the cloud computing paradigm implicitly contains is that of secure outsourcing of sensitive as well as critical data and processes. When considering using a cloud service, the user must be aware of the fact that all data given to the cloud provider leave the own control and protection sphere. Hence, a strong trust relationship between the cloud provider and the cloud user is considered a general prerequisite in cloud computing. Depending on the political context this trust may touch legal obligations. An attacker that has access to the cloud storage component is able to take snapshots or alter data in the storage. This might be done once, multiple times, or continuously. An attacker that also has access to the processing logic of the cloud can also modify the functions and their input and output data.These developed multi cloud architectures allow to categorize the available schemes and to analyze them according to their security benefits. An assessment of the different methods 1) Replication of applications, 2) Partition of application System into tiers, 3) Partition of application logic into fragments and 4) Partition of application data into fragments is given in particular.

- **Replication of applications** allows to receive multiple results from one operation performed in distinct clouds and tocompare them within the own premise. This enables the user to get evidence on the integrity of the result.
- **Partition of application System into tiers** allows separating the logic from the data. This gives additional protection against data leakage due to flaws in the application logic.
- **Partition of application logic into fragments** allows distributing the application logic to distinct clouds. This has twobenefits. First, no cloud provider learns the complete application logic. Second, no cloud provider learns the overall calculated result of the application. Thus, this leads to data and application confidentiality.
- **Partition of application data into fragments** allows distributing fine-grained fragments of the data to distinct clouds. Noneof the involved cloud providers gains access to all the data, which safeguards the data's confidentiality.

Cloud security is discussions to date mostly focus on the fact that customers must completely trust their cloud providers with respect to the confidentiality and integrity of their data, as well as computation faultlessness. The attacker gains immense potency over the customer's data. This attack vector is a novelty as the result of the control interface (alongside with virtualization techniques) being a new feature of the Cloud Computing paradigm, as NIST lists On-demand self-service and Broad network access as essential characteristics of Cloud Computing systems [1]. The main goal of this paper [2] is the investigation and evaluation of security and privacy threats caused by the unawareness of users in the cloud. Although the methods and techniques described in this paper are applicable to arbitrary IaaS providers, they focused on one of the major cloud providers.

2.SYSTEM MODEL, ENCRYPTIONAND DECRYPTION, BATCH AUDITING PROTOCOL

2.1System Model

The system model consists of four objects: The data blocks is stored and retrieved in different cloud locations based on the storage and computational capability. Thus the proposed system explores such issue to provide the support of variable-length block verification. . Likewise, session based de-duplication is considered. Here if the user provides the session duration, from date and to date, then only with the data range, proof of ownership can be allowed in server on those dates.Different trust level is set to different cloud providers and encryption/decryption is varied based on the clouds computational capability.

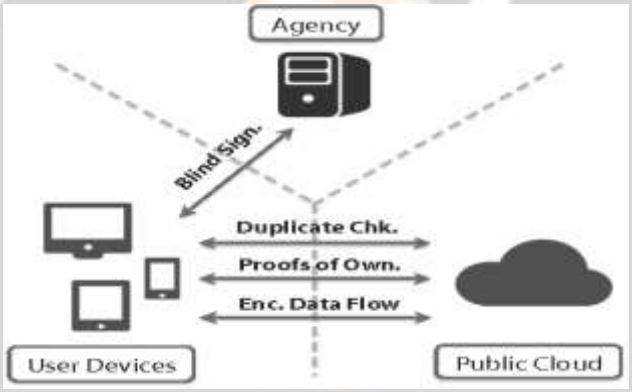

Fig 1.System Model

2.2 Encryption and Decryption

Fig. 2 describes the encryption process. There are three objectives in this process.

- Authorized cloud user can access this process in this level, and checks theauthorization.
- Valid user can enter the text with password for encryption and it will show their encrypted text below.
- The same user can decrypt that text in few seconds using AESalgorithm

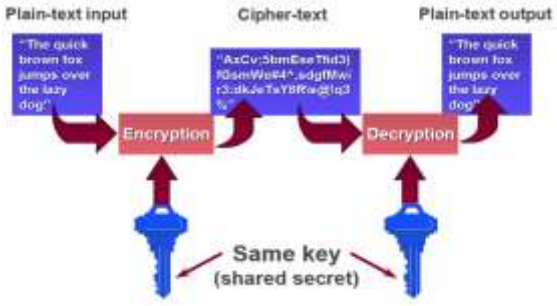

Fig 2.Encryption Process

2.3 Batch Auditing Protocol

In this batch auditing protocol, during auditing, two processes of same third party auditor randomly pick the two set of

segment ids and send corresponding prime number vectors to cloud server. If the credentials match, then the file integrity is said to be verified.

Fig 3.Batch Auditing Protocol

3.METHODOLOGY

3.1 AES Implementation

The AES encryption algorithm is a block cipher that uses an encryption key and several rounds of encryption. A block cipher is an encryption algorithm that works on a single block of data at a time. In the case of standard AES encryption, the block used is 128 bits, or 16 bytes, in length. The term "rounds" refers to the way in which the encryption algorithm mixes the data re-encrypting it ten to fourteen times depending on length of thekey.

In this paper, PHP version of AES a server-side complement to JavaScript AES implementation was implemented.

AES Cipher function:
encrypt 'input' with Rijindeal algorithm param input message as byte-array(16 bytes) param w key schedule as 2D byte-array (Nr+1 x Nb bytes)
generated from the cipher key by keyExpansion() return cipher as byte-array(16 bytes)

3.2 3TDES Implementation

3TDES, user first generate and distribute a 3TDES key K, which consists of three different DES keys K1, K2 and K3. This means that the actual 3TDES key has length 3×56 = 168 bits. The encryption scheme is illustrated as follows −

- Encrypt the plaintext blocks using single DES with key K1.
- Now decrypt the output of step 1 using single DES with key K2.
- Finally, encrypt the output of step 2 using single DES with key K3.
- The output of step 3 is the ciphertext.
- Decryption of a ciphertext is a reverse process. User first decrypt using K3, then encrypt with K2, and finally decrypt with K1.

Due to this design of Triple DES as an encrypt–decrypt–encrypt process, it is possible to use a 3TDES (hardware) implementation for single DES by setting K1, K2, and K3 to be the same value. This provides backwards compatibility with DES.

4. Units

DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key**.**The features of AES are as follows −

- Symmetric key symmetric block cipher
- 128-bit data, 128/192/256-bit keys
- Stronger and faster than Triple-DES
- Provide full specification and design details
- Software implementable in C and Java

Before using 3TDES, user first generate and distribute a 3TDES key K, which consists of three different DES keys K1, K2 and K3. This means that the actual 3TDES key has length 3×56 = 168 bits. Due to this design of Triple DES as an encrypt–decrypt–encrypt process, it is possible to use a 3TDES (hardware) implementation for single DES by setting K1, K2, and K3 to be the same value. This provides backwards compatibility with DES.Second variant of Triple DES (2TDES) is identical to 3TDES except that K3is replaced by K1. In other words, user encrypt plaintext blocks with key K1, then decrypt with key K2, and finally encrypt with K1 again. Therefore, 2TDES has a key length of 112 bits.Triple DES uses a "key bundle" that comprises three DES keys, K1, K2 and K3, each of 56 bits (excluding parity bits). The encryption algorithm is:

ciphertext = EK3(DK2(EK1(plaintext)))

I.e., DES encrypt with K1, DES decrypt with K2, then DES encrypt with K3.Decryption is the reverse:

plaintext = DK1(EK2(DK3(ciphertext)))

I.e., decrypt with K3, encrypt with K2, then decrypt with K1.

A naive approach to increase strength of a block encryption algorithm with short key length (like DES) would be to use two keys (K1, K2) instead of one, and encrypt each block twice: EK2(EK1(plaintext)). If the original key length is n bits, one would hope this scheme provides security equivalent to using key 2n bits long. Unfortunately, this approach is vulnerable to meet-in-the-middle attack: given a known plaintext pair (x, y), such that y = EK2(EK1(x)), one can recover the key pair (K1, K2) in ~2n steps, instead of ~22n steps one would expect from algorithm with 2n bits of key.

5.Results

AES is an iterative rather than Feistel cipher. It is based on ‘substitution–permutation network’. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations).Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix－Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.Triple DES is another mode of DES operation. It takes three 64-bit keys, for an overall key length of 192 bits. In Stealth, you simply type in the entire 192-bit (24 character) key rather than entering each of the three keys individually. The Triple DES DLL then breaks the user-provided key into three subkeys, padding the keys if necessary so they are each 64 bits long. The procedure for encryption is exactly the same as regular DES, but it is repeated three times, hence the name Triple DES. Like DES, data is encrypted and decrypted in 64-bit chunks. Although the input key for DES is 64 bits long, the actual key used by DES is only 56 bits in length. The least significant (right-most) bit in each byte is a parity bit, and should be set so that there are always an odd number of 1s in every byte. These parity bits are ignored, so only the seven most significant bits of each byte are used, resulting in a keylength of 56 bits. This means that the effective key strength for Triple DES is actually 168 bits because each of the three keys contains 8 parity bits that are not used during the encryption process.

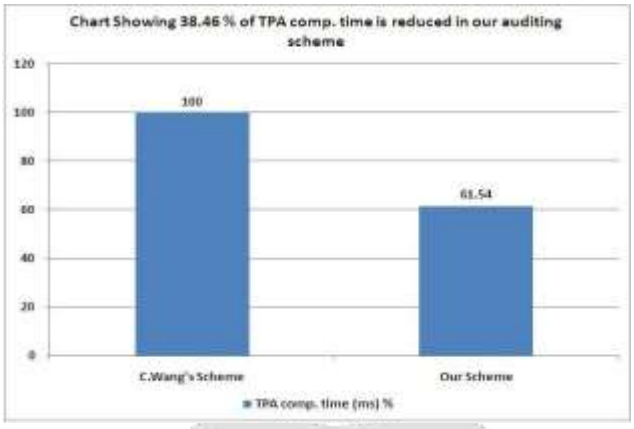

Fig 4 Chart Comparisons for TPA Computation Time In %

Fig (a)Add User details

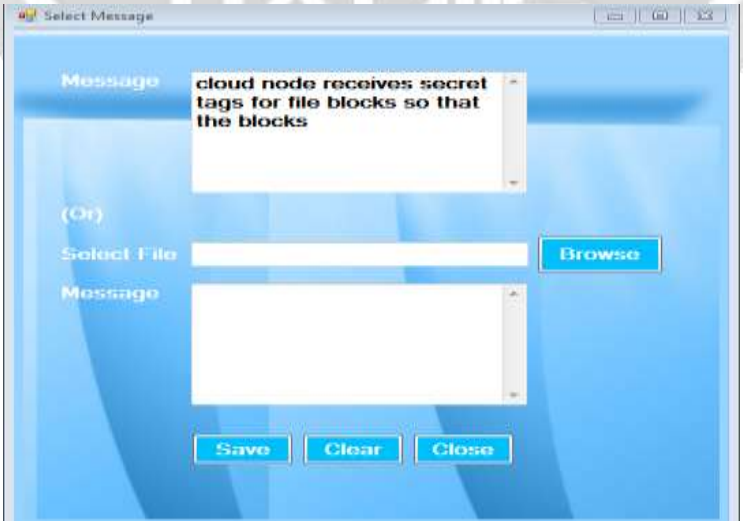

Fig (b)Select message from Sender

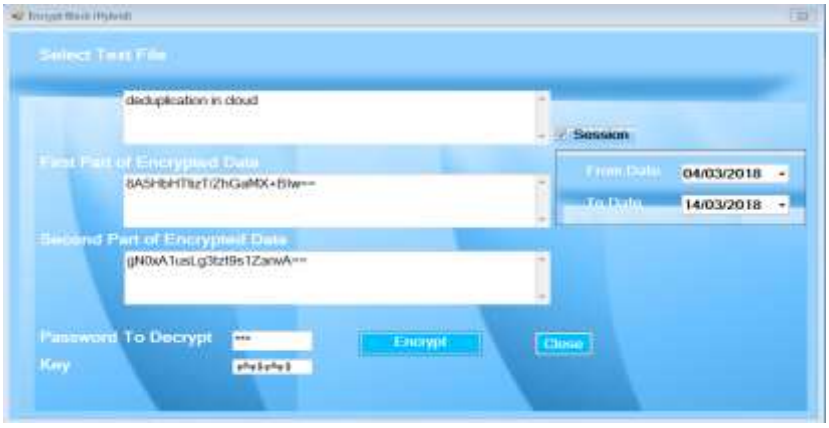

Fig (c) encryption

Fig (d) decryption

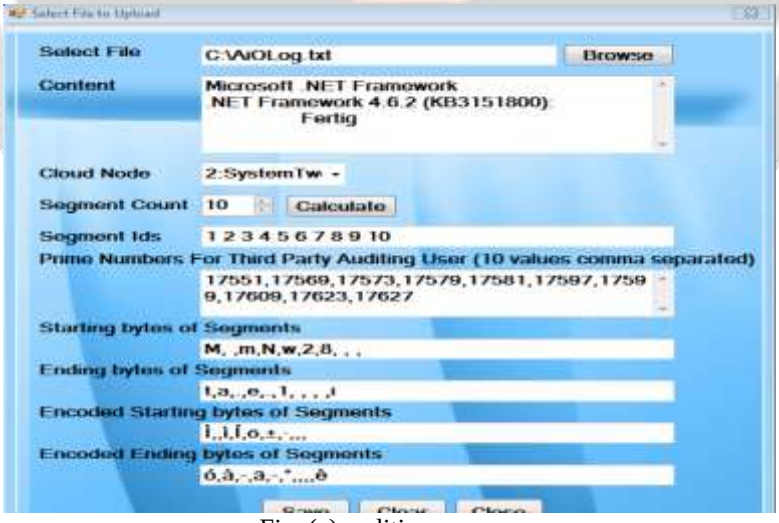

Fig (e) auditing process

6.CONCLUSION

This paper present secure information deduplication is a procedure for dispensing with copy duplicates of information, and has been broadly utilized in cloud capacity to decrease capacity space and transfer transmission capacity. This venture endeavors to formally address the issue of accomplishing effective and tried and true key administration in secure deduplication. A trial run of the framework has been made and is giving great comes about the methods for preparing is basic and customary arrange. The paper successfully stores and recover the records from the cloud space

database server. The records are scrambled and decoded at whatever point essential so that they are secure. This increments the security on the off chance that the outsourced information require to be securely gotten to on the given term. With the offer assistance of AES calculation, the encryption handle was effectively completed.

7.REFERENCES

[1] Tejashree Paigude, Prof. T. A. Chavan "A survey on Privacy Preserving Public Auditing fo Data Storage Security"International Journal ofComputer Trends and Technology- volume4Issue3- 2013

[2]"Privacy-Preserving Public Auditing for Regenerating-Code- Based Cloud Storage". Volume II, Issue X, October – 2015. International Journal On Engineering Technology and Science October – 2015

[3] Sultan Aldossary* William Allen" Data Security, Privacy, Availability and Integrity in Cloud Computing: Issues and Current Solutions" (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 4, 2016

[4]Jin Wang "Mutual Verifiable Provable Data Auditing in Public Cloud Storage "Journal of Internet Technology Volume 16 (2015) No.2

[5] Jiawei Yuan, Shucheng Yu "Public Integrity Auditing for Dynamic Data Sharing with Multi-User Modification". 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.