IDENTIFYING INFORMATION RELOCATE WITH RELIABLE ESTIMATION AND SECURING DATA ON CLOUD COMPUTING

ISWARYA G M[1], KAAVYA SHREE M A[2], KEERTHANA K[3]

[1,2,3]*Computer science and Engineering, Kingston Engineering College, Vellore, India.*

---***---

Abstract— The cloud data services, it is commonplace for data to be not only stored in the cloud, but also shared across multiple users. Unfortunately, the integrity of cloud data is subject to skepticism due to the existence of hardware/software failures and human errors. Several mechanisms have been designed to allow both data owners and public verifiers to efficiently audit cloud data integrity without retrieving the entire data from the cloud server. Public auditing on the integrity of shared data with these existing mechanisms will inevitably reveal confidential information. Cloud Computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. In particular, we consider the task of allowing a threshold proxy re- encryption, on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud. While prior works on ensuring remote data integrity often lacks the support of either public Audit ability or dynamic data operations, this paper achieves both.

Keywords— Block chain technique, secure erasure code, AES, checksum algorithm.

1. INTRODUCTION

Distributed computing has been envisioned as the accompanying creation information development plan for endeavors, due to its broad summary of unparalleled inclinations in the IT history: on-ask for self-advantage, inescapable framework get to, zone self-choosing resource pooling, quick resource adaptability, use based assessing and transference of peril. As an aggravating development with huge consequences, distributed computing is changing the specific method for how associations use information advancement. One fundamental piece of this standpoint changing is that data are being united or outsourced to the. From customers' view, including together individuals and IT tries, securing data remotely to the in a versatile on-ask for strategy bring engaging focal points: landing of the weight for storage space organization, vast data access with put self-sufficiency, and avoidance of advantages costs on hardware, programming, and staff frameworks of help, etcetera. While distributed computing make these compensation more captivating than some other time in ongoing memory, it also passes on new and testing security risks to customers' outsourced data. As organization providers (CSP) are part administrative components, data outsourcing is truly surrendering customer's last control more than the fate of their data. As an issue of first significance, in spite of the way that the structures underneath the are altogether more powerful and trustworthy than individual enlisting devices, they are still before the broad assortment of both inside and outside risks for data respectability.

2. RELATED WORKS

REFERENCES	FEATURES	DRAWBACKS
Zhen Liu, Zhenfu Cao, and Duncan S. Wong	Traceable cipher text policy attribute based encryption (T-CP-ABE).	Decryption keys of multiple users are not linked to user's identification details.
Jinguang Han, Willy Susilo, Yi Mu , Jianying Zhou, and Man Ho Au	Privacy-preserving decentralized CP-ABE (PPDCPABE).	Attributes can reveal user identities.
Jianting Ning, Xiaolei Dong, Zhenfu Cao, Lifei Wei, and Xiaodong Lin	Ciphertext-policy attribute-based encryption (CP-ABE) enables fine-grained access control.	User must share data in expressive way.
Jiguo Li, Wei Yao, Yichen Zhang, Huiling Qian and Jinguang Han	ciphertext policy attribute based encryption (CP-ABE) scheme with efficient user revocation.	Revocation of a user affects another.

Jiguo Li, Xiaonan Lin, Yichen Zhang and Jinguang Han	KSF-OABE scheme is proved secure against chosen-plaintext attack (CPA).	Collusion attack.
YuJui Chang, JaLing Wu	Searchable encryption scheme to search the data stored on the Cloud server in ciphertext domain.	Security and privacy risks.
Cong Zuo, Jun Shao , Joseph K. Liu, Guiyi Wei and Yun Ling Jian Xu, Changyong Liang, Hemant K. Jain, and Dongxiao Gu	To protect the confidentiality of the shared sensitive data, the cryptographic techniques are used.	Data access without authorization.
Jian Xu, Changyong Liang, Hemant K. Jain, and Dongxiao Gu	Nash equilibrium.	Time factor and security.

3. ANALYSES OF PREVIOUS METHOD

Cloud Computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. In particular, we consider the task of allowing a threshold proxy re-encryption, on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud. While prior works on ensuring remote data integrity often lacks the support of either public Audit ability or dynamic data operations, this paper achieves both. The integrity of shared data with these existing Mechanisms will inevitably reveal confidential information—identity privacy—to public verifiers. Public auditing mechanisms can actually be extended to verify shared data integrity.

However, a new significant privacy issue introduced in the case of shared data with the use of existing mechanisms is the leakage of identity privacy to public verifiers. Once a block in this shared file is modified by a user, this user needs to sign the new block using his/her private key. Eventually, different blocks are signed by different users due to the modification introduced by these two different users. The main problem with this approach is that it requires all the users using designed hardware, and needs the cloud provider to move all the existing cloud services to the trusted computing the verifier does not need to download all the blocks to check the integrity of data. Non-malleability indicates that an adversary cannot generate valid signatures on arbitrary blocks by linearly combining existing signatures.

4. ANALYSES OF PROPOSED METHOD

In this paper, we propose an effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud. We rely on Block chain Genesis technique in the file distribution preparation to provide redundancies and guarantee the data dependability. This construction drastically reduces the communication and storage overhead as compared to the traditional replication-based file distribution techniques. By utilizing the homomorphism token with distributed verification of block chained data, our scheme achieves the storage correctness insurance as well as data error localization. To allow not only data owner itself but also a public verifier to efficiently perform integrity checking without downloading the entire data from the cloud, which is referred to as public auditing With ring signatures, a verifier is convinced that a signature is computed using one of group members' private keys, but the verifier is not able to determine which one. The public verifier knows each block in shared data is either signed by Alice or Bob, because it needs both users' public keys to Verify the correctness of the entire shared data.

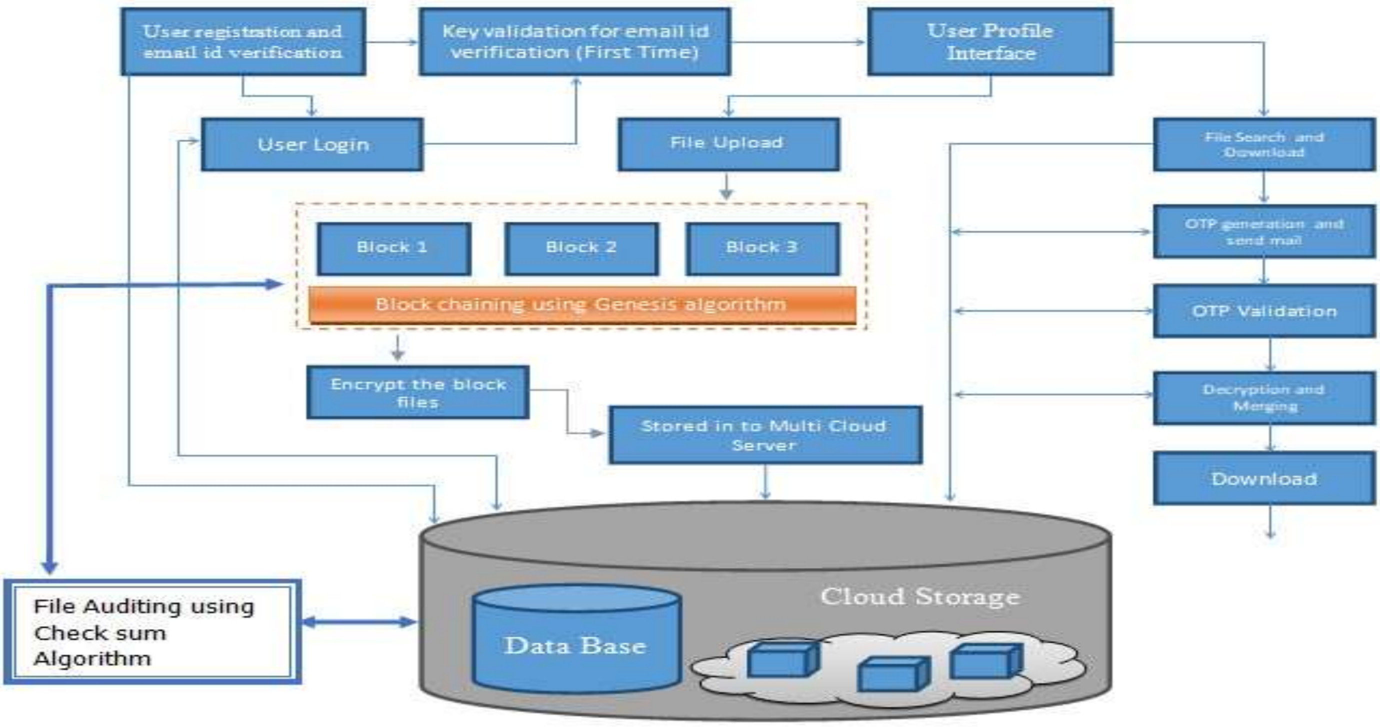
User registration and email id verification
Key validation for email id verification (First Time)
User Profile Interface
User Login
File Upload
File Search and Download
Block 1
Block 2
Block 3
Block chaining using Genesis algorithm
OTP generation and send mail
OTP Validation
Encrypt the block files
Stored in to Multi Cloud Server
Decryption and Merging
Download
File Auditing using Check sum Algorithm
Cloud Storage
Data Base

Fig: Proposed diagram

DATA INTEGRITY

Data integrity refers to maintaining and assuring the accuracy and consistency of data over its entire life-cycle, Data integrity is the opposite of data corruption, which is a form of data loss. The overall intent of any data integrity technique is the same: ensure data is recorded exactly as intended (such as a database correctly rejecting mutually exclusive possibilities,) and upon later retrieval, ensure the data is the same as it was when it was originally recorded. In short, data integrity aims to prevent unintentional changes to information.

Data integrity is not to be confused with data security, the discipline of protecting data from unauthorized parties.

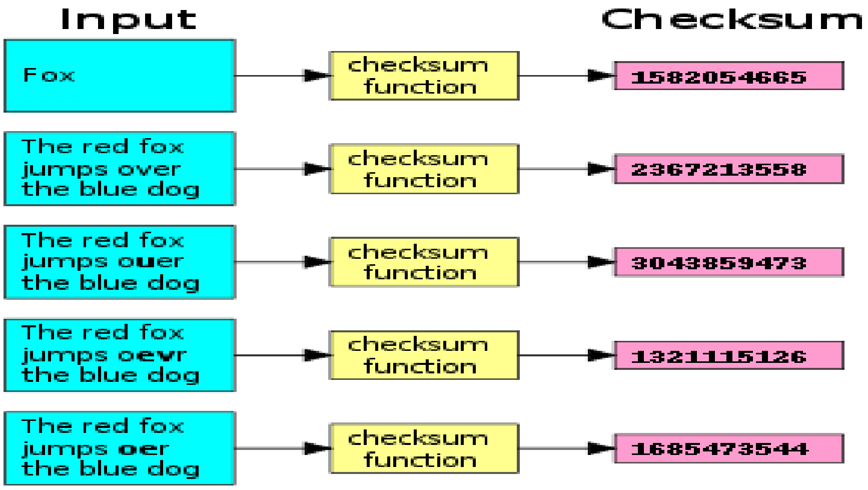

Fig: Data Integrity

ALGORITHM:

BLOCK CHAIN ALGORITHM:

Step 1: Given a signal of m blocks, recode to n

- Blocks where n > m
- Optimal: reconstruct signal given any m
- Unique blocks

Step 2: Suboptimal: Reconstruct signal using (1+e)

m unique blocks

- Rate r=m/n, and storage overhead is 1/r

Optimal erasure codes have the property that any k out of the n code word symbols are sufficient to recover the original message (i.e., they have optimal reception efficiency). Optimal erasure codes are maximum distance separable codes (MDS codes)

Step 3: A cloud storage system, consisting of a collection of storage servers, provides long-term storage services over the Internet. Storing data in a third party's cloud system causes serious concern over data confidentiality.

Step 4: General encryption schemes protect data confidentiality, but also limit the functionality of the storage system because a few operations are supported over encrypted data.

Step 5: Parity check

Parity check is the special case where $n = k + 1$. From a set of k values $\{v_i\}_{1 \leq i \leq k}$, a checksum is computed and appended to the k source values:

$$v_{k+1} = -\sum_{i=1}^{k} v_i \,.$$

The set of $k + 1$ values $\{v_i\}_{1 \leq i \leq k+1}$is now consistent with regard to the checksum. If one of these values, v_e, is erased, it can be easily recovered by summing the remaining variables:

$$v_e = -\sum_{i=1, i \neq e}^{k+1} v_i \,.$$

AES or DES

It is a web tool to encrypt and decrypt text using AES encryption algorithm. You can chose 128, 192 or 256-bit long key size for encryption and decryption. The result of the process is downloadable in a text file.

AES ENCRYPTION

AES (acronym of Advanced Encryption Standard) is a symmetric encryption algorithm.The algorithm was developed by two Belgian cryptographer Joan Daemen and Vincent Rijmen.

AES was designed to be efficient in both hardware and software, and supports a block length of 128 bits and key lengths of 128, 192, and 256 bits.

AES ENCRYPTION ALGORITHM:

AES encryption is used by U.S. for securing sensitive but unclassified material, so we can say it is enough secure. String key=”sgarfdh73286”

```
                int mode=Cipher.ENCRYPT_MODE; DESKeySpec dks = new DESKeySpec(key.getBytes());

                  SecretKeyFactory skf = SecretKeyFactory.getInstance("DES");

                  SecretKey desKey = skf.generateSecret(dks); Cipher cipher = Cipher.getInstance("DES"); //

DES/ECB/PKCS5Padding for SunJCE

      if (mode == Cipher.ENCRYPT_MODE) {

                           cipher.init(Cipher.ENCRYPT_MODE, desKey);

                           CipherInputStream cis = new

CipherInputStream(fise, cipher);

                           //doCopy(cis, os);

               byte[] bytes = new byte[64];

                  int numBytes;

                  while ((numBytes = cis.read(bytes)) != -1) {

                           fos.write(bytes, 0, numBytes);

                  }

                  }
```

CHECKSUM ALGORITHM WORKING:

Owner file size = 10MB

Number of lines in owner’s file = 360 lines

1st split file size and number of lines = 3.3MB / 120 lines

2nd split file size and number of lines = 3.3MB / 120 lines

3rd split file size and number of lines = 3.4MB / 120 lines

User edited size = 10.2MB

Number of lines in edited file = 370 lines

Edited 1st split file size and number of lines = 3.3MB / 120 lines Edited 2nd split file size and number of lines = 3.32MB / 130 lines Edited 3rd split file size and number of lines = 3.4MB / 120 lines

While comparing the Owner file and the edited file the changes can be found exactly in the particular area and it will be considered as an authorization and notification is send to the owner

CONCLUSIONS

We proposed a privacy-preserving public auditing system for data storage security in cloud computing. We utilize the homomorphic linear authenticator and random masking to guarantee that the TPA would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process, which not only eliminates the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage. Considering TPA may concurrently handle multiple audit sessions from different users for their outsourced data files, we further extend our privacy- preserving public auditing protocol into a multiuser setting, where the TPA can perform multiple auditing tasks in a batch manner for better efficiency.

REFERENCES:

[1] Z. Liu, Z. Cao and Duncan S. Wong, "White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures," IEEE Transactions on Information Forensics and Security, vol. 8, no. 1, pp. 76-88, 2013.

[2] J. Han, W. Susilo, Y. Mu, J. Zhou, M. Au, "Improving privacy and security in decentralized ciphertext-policy attribute-based encryption," IEEE Transactions on Information Forensics and Security, 2015, 10(3): 665-678

[3] J. Ning, X. Dong, Z. Cao, L. Wei and X. Lin, "White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes," IEEE Transactions on Information Forensics and Security, vol.10, no 6, pp. 1274-1288, 2015.

[4] J. Li, W. Yao, Y. Zhang, H. Qian and J. Han, "Flexible and fine grained attribute-based data storage in cloud computing," IEEE Trans. Service Computing, 2017, 10(5): 785-796.

[5] J. Li, X. Lin, Y. Zhang, and J. Han, "KSF-OABE: outsourced attribute-based encryption with keyword search function for cloud storage," IEEE Trans. Service Computing, vol. 10, no. 5, pp. 715-725, Sept.-Oct. 2017

[6] J. Li, Y. Shi, and Y. Zhang, "Searchable ciphertext-policy attribute-based encryption with revocation in cloud storage," International Journal of Communication Systems, vol. 30, no. 1, Art. no. e2942, Jan. 2017.

[7] C. Zuo, J. Shao, J. K. Liu, G. Wei and Y. Ling, "Fine- grained two-factor protection mechanism for data sharing in cloud storage," IEEE Trans. Information Forensics and Security, vol. 13, no 1, pp. 186-196, 2018

[8] Jian Xu, Changyong Liang, Hemant K. Jain, and Dongxiao Gu, "Openness and Security in Cloud Computing Services: Assessment Methods and Investment Strategies Analysis", IEEE Access, vol 7, Mar 2019.

[9] H. Li, H. Zhu, S. Du, X. Liang, X. Shen, "Privacy leakage of location sharing in mobile social networks: attacks and defense," IEEE Trans. on Dependable and Secure Computing, 2016, DOI: 10.1109/TDSC.2016.2604383

[10] J. Li, Y. Wang, Y. Zhang and J. Han. "Full verifiability for outsourced decryption in attribute based encryption," IEEE Transactions on Services Computing, 2017, DOI:

10.1109/TSC.2017.2710190