Data storage security in cloud computing

Priyansh jauhari
[1]*Student, College of Computing Sciences And Information Technology,Teerthanker Mahaveer University, Moradabad*
[2] *Assistant Professor, College of Computing Sciences And Information Technology,Teerthanker Mahaveer University, Moradabad*
Priyanshjauhari13@gmail.com

***Abstract*— IN these days's cloud computing emerging field because of its performance high availability at low cost .cloud is kind of compact database where many organizations store their data. Data store is main cloud service provide to big organization to store huge amount of data but still many organization are not ready to implement cloud computing technology because is following reason that lack of safety, data redundancy misbehaviour of server. So main object of this paper is solve the above reason that are prevent not permitted access ,it can be done with help of distributed some homomorphism token provide security of data cloud .the cloud is support for data redundancy means client can insert, delete or can update data should be security mechanism which ensure integrity of data. This paper also secures the data while the misbehaving of server side arises.**
In this paper, we focus on ensure data storage security in cloud computing, which is an important aspect of Quality of Service

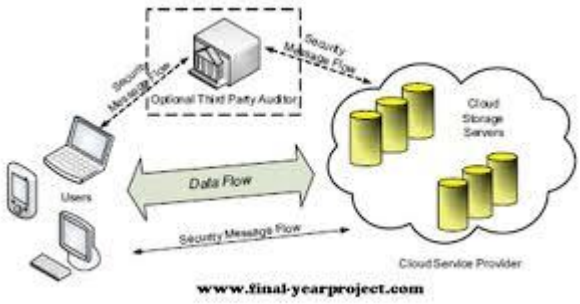
***Keywords*—** Homomorphism token, Distributed scheme, Data redundancy, cloud.

I. INTRODUCTION

Now a day Cloud Computing become so strong, because it is an Internet-based development and use of computer technology and also cheaper as well as more powerful processors, together with the software as a service (SaaS) computing architecture. Due to increase in network bandwidth it becomes faster to provide quality of services as compare to previous. Also support to moving the data between cloud and client without any complexity because of releasing the hardware difficulty. Because of online base computing it provide huge amount of data storage and resources to the local machine and eliminate the local machine to maintenance separate data.

As a result, users are at the thankful of their cloud service providers for the availability and integrity of their data

Data security is always been the important aspect of quality of services, Cloud computing every time invites the new challenges of security thread for number of reasons. Firstly, traditional cryptographic cannot be used directly data security purpose because users' loss control of data under Cloud Computing. Therefore, verification of correct data whether it store correctly or not in the cloud must be conducted without explicit knowledge of the whole data. Due to the continuously demanding of long term storage of data with correctness and security become more challenging. Secondly, Cloud Computing is not just a third party data warehouse.



Cloud computing, to put it simply, means internet computing. The internet is commonly visualized as clouds; hence the term "cloud computing" for computation done through the internet. With cloud computing users can access database resources via the internet from anywhere, for as long as they need, without worrying about any maintenance or management of actual resources. Besides, databases in cloud are very dynamic and scalable. Cloud Computing is unlike grid computing, utility computing, or autonomic computing. In fact, it is a very independent platform in terms of computing. The best example of cloud computing is Google apps where any application can be accessed using a browser and it can be deployed on thousands of computer through the internet. It also provides facilities for users to develop, deploy and manage their applications on the cloud, which entails virtualization of resources that maintains and

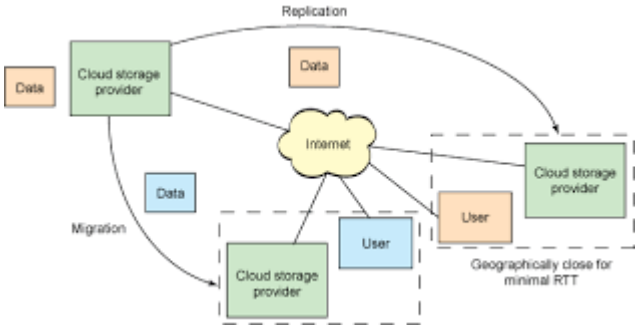
manages itself. Our proposed scheme enables the data owner to delegate tasks of data file re-encryption and user secret key update to cloud servers without disclosing data contents or user access privilege information. We achieve this goal by exploiting and uniquely combining techniques and algorithms (Correctness Verification and Error Localization, traditional replication-based file distribution, adding random perturbations). 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 erasure-correcting code 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 homomorphic token with distributed verification of erasure-coded data, Cloud computing, to put it simply, means internet computing. The internet is commonly visualized as clouds; hence the term "cloud computing" for computation done through the internet. With cloud computing users can access database resources via the internet from anywhere, for as long as they need, without worrying about any maintenance or management of actual resources. Besides, databases in cloud are very dynamic and scalable. Cloud Computing is unlike grid computing, utility computing, or autonomic computing. In fact, it is a very independent platform in terms of computing. The best example of cloud computing is Google apps where any application can be accessed using a browser and it can be deployed on thousands of computer through the internet. It also provides facilities for users to develop, deploy and manage their applications on the cloud, which entails virtualization of resources that maintains and manages itself. Our proposed scheme enables the data owner to delegate tasks of data file re-encryption and user secret key update to cloud servers without disclosing data contents or user access privilege information. We achieve this goal by exploiting and uniquely combining techniques and algorithms (Correctness Verification and Error Localization, traditional replication-based file distribution, adding random perturbations). 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 erasure-correcting code 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 homomorphic token with distributed verification of erasure-coded data, our scheme achieves the storage correctness insurance as well as data error localization: whenever data corruption has been detected during the storage correctness verification, our scheme can almost guarantee the simultaneous localization of data errors i.e. the identification of the misbehaving server(s).

II. ENSURING CLOUD DATA STORAGE

In cloud data storage system, users store their data and do not possess longer data locally. Due to which, the correctness and availability of the data files which being stored on the distributed cloud servers must be definite. The most important issue is to effectively detect any unauthorized data modification and corruption, which occur due to server compromise and random Byzantine failures. Whereas, in the distributed case such inconsistencies are successfully detected, and also to find on which server the data error lies become great significance, hence it can be the first step to fast recover the storage errors. So to address and solve all these kind of problems, our main scheme for ensuring cloud data storage is presented in this section. The first part of the section is to develop and review of basic tools from coding theory that is needed in our scheme for file distribution in cloud servers. Then, our main tool homomorphic token is introduced. The token computation function we are considering belongs to a family of universal hash function, chosen to preserve the homomorphism properties, which can be perfectly integrated with the verification of erasure-coded data. Side by side,

it is also shown verifying the storage correctness as well as identifying misbehaving server. The last but not the least the most important point of this paper is that, here trying to implement the cloud computing with the mobile computing. So that the user are not restricted to access the data in the cloud via personal computer or laptop. But he or she can be access the cloud account via mobile phone also. It will be so handy for the user to take the cell phone with them. So that they can access their cloud account everywhere.

1) **Structure of cloud**: Whatever the data store in the cloud that may be update frequently by the user like insertion, deletion, appending, recording, etc. So to ensure the data storage correctness under dynamic data update is hence so much of important. However, this dynamic feature also makes traditional integrity insurance techniques futile and entails new solutions. Last but not the least, the deployment of Cloud Computing is powered by data centres running in a simultaneous, cooperated and distributed manner. The different user data is stored in different physical locations to further reduce the data integrity threats. Therefore, it is most importance to achieving a robust and secure cloud data storage system in the real world. Recently, the importance of ensuring the remote data integrity has been highlighted by the following research works. Since they are all focusing on single server scenario so these techniques can be useful to ensure the storage correctness without having individual users possessing data, also cannot address all the security threats in cloud data storage, and most of them do not consider dynamic data operations. As a balancing approach, researchers have also proposed distributed protocols for ensuring storage correctness across multiple servers or peers.



Security Analysis and Performance Evaluation: Our security analysis focuses on the model as defined. We also provide an efficiency of our scheme via implementation of both file distribution preparation and verification token pre-computation. In our scheme, servers are required to operate on specified rows to check correctness and verification for the calculation of requested token. We will show that this "sampling" strategy on selected rows instead of all document or data which can greatly reduce the computational overhead on the server, also maintaining the detection of the data corruption with high probability. Suppose any servers are misbehaving due to the possible compromise or Byzantine failure. In the following analysis, we do not limit the value of any server, i.e., servers $\leqslant$ n. Assume the adversary modifies the data blocks in z rows out of the l rows in the encoded file matrix. Let r be the number of different rows for which the user asks for check in a challenge. Let X be a discrete random variable that is defined to be the number of rows chosen by the user that matches the rows modified by the adversary.

2) ***Secure Data Storage In Cloud***: In cloud data storage system, users store their data in the cloud and no longer possess the data locally. Thus, the correctness and availability of the data files being stored on the distributed cloud servers must be guaranteed. One of the key issues is to effectively detect any unauthorized data modification and corruption, possibly due to server compromise and/or random Byzantine failures. Besides, in the distributed case when such inconsistencies are successfully detected, to find which server the data error lies in is also of great significance, since it can be the first step to fast recover the storage errors. To address these problems, our main scheme for ensuring cloud data storage is presented in this section. The first part of the section is devoted to a review of basic tools from coding theory that are needed in our scheme for file distribution across cloud servers. Then, the homomorphic token is introduced. The token computation function we are considering belongs to a family of universal hash function, chosen to preserve the homomorphic properties, which can be perfectly integrated with the verification of erasure- coded data.

3. Security Analysis and Performance Evaluation

Our security analysis focuses on the model as defined. We also provide an efficiency of our scheme via implementation of both file distribution preparation and verification token pre-computation. In our scheme, servers are required to operate on specified rows to check correctness and verification for the calculation of requested token. We will show that this "sampling" strategy on selected rows instead of all document or data which can greatly reduce the computational overhead on the server, also maintain the detection of the data corruption with high probability. Suppose any servers are misbehaving due to the possible compromise or Byzantine failure. In the following analysis, we do not limit the value of any server, i.e., servers $\leqslant$ n. Assume the adversary modifies the data blocks in z rows out of the l rows in the encoded file matrix. Let r be the number of different rows for which the user asks for check in a challenge. Let X be a discrete random variable that is defined to be the number of rows chosen by the user that matches the rows modified by the adversary.

.

III. RELATED TO WORK:

An effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud was proposed by C. Wang, Q. Wang, K. Ren, and W. Lou in July 2009. C. Wang, Q. Wang, K. Ren, and W. Lou rely on erasure correcting code in the file distribution preparation to provide redundancies and guarantee the data dependability. This construction might drastically reduce the communication and storage overhead as compared to the traditional replication-based file distribution techniques. Their scheme achieves the storage correctness insurance as well as data error localization, that is, whenever data corruption has been detected during the storage correctness verification, their scheme can almost guarantee the simultaneous localization of data errors. Later in May 2011, Cong Wang, Qian Wang, Kui Ren, Wenjing Lou extended their work to allow user to audit the cloud storage with very lightweight communication and computation cost, proposed scheme that is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

A formal "Proof of Retrievability" (POR) model for ensuring the remote data integrity was described by A. Juels and J. Burton S. Kaliski in October 2007. Their scheme combines two methods spot-checking and error- correcting code to ensure both possession and retrievability of files on archive or backup service systems. H. Shacham and B. Waters in 2008 built on this model and constructed a random linear function based homomorphism authenticator which enables unlimited number of queries and requires less communication overhead. An improved framework for POR protocols that generalizes both Juels and Shacham's work was illustrated [17]. All these schemes are focusing on static data. The effectiveness of their schemes rests primarily on the pre-processing steps that the user conducts before outsourcing the data file F. Any change to the contents of F, even few bits, must propagate through the error-correcting code, thus introducing significant computation and communication complexity was proposed by Bowers in 2009.

The "provable data possession" (PDP) model for ensuring possession of file on un trusted storages was defined by Ateniese et al . Their scheme utilized public key based homomorphic tags for auditing the data file, thus providing public verifiability. However, their scheme requires sufficient computation overhead that can be expensive for an entire file. Later in their subsequent work during 2008, described a PDP scheme that uses only symmetric key cryptography. This method has lower-overhead than their previous scheme and allows for block updates, deletions and appends to the stored file, which has also been supported in our work. However, their scheme focuses on single server scenario and does not address small data corruptions, leaving both the distributed scenario and data error recovery issue unexplored. A new efficient means of polynomial in the size of the input (i.e. key or data) was proposed by M. A. Shah, R. Swaminathan, and M. Baker during the year 2008 in "Privacy Preserving audit and extraction of digital contents". The main threat from the auditor is that it may glean important information from the auditing process that could compromise the privacy guarantees provided by the service. For example, even a few bits from a file containing medical history could reveal whether a customer has a disease. To ensure privacy, there exist different standards for the encrypted data and the encryption key. For the data, the system relies on the strength of the encryption scheme and the zero-knowledge property of the protocol for encryption-key audits.

To ensure file integrity across multiple distributed servers, using erasure-coding and block-level file integrity checks was proposed by T. S. J. Schwarz and E. L. Miller in 2009. However, their scheme only considers static data files. To verify data integrity using RSA-based hash for data possession in peer-to-peer file sharing networks was defined by D. L. G. Filho and P. S. L. M. Barreto in 2006. However, their proposal requires exponentiation over the entire data file, which is clearly impractical for the server whenever the file is large

IV. CONCLUSION AND FUTURE WORK

Cloud Computing is gaining remarkable popularity in the recent years for its benefits in terms of flexibility, scalability, reliability and cost effectiveness. Despite all the promises however, Cloud Computing has one problem: Security. In this paper, we studied the problems of data security in cloud data storage, which is essentially a distributed storage system. An effective and flexible distributed scheme is proposed to ensure the correctness of users' data in the cloud servers. If this correctness verification is too much resource consuming on the user's side, the task can be delegated to the third party auditor and the pre-computed tokens could be either in the user's local device or cloud server in encrypted format. By detailed security and performance analysis, we show that our scheme is highly efficient in recovering the singleton losses almost immediately and recovers from bursty data losses. We envisage several possible directions for future research on this area. As our future work we focus on reducing the impact in maintaining the challenge key in user's local space. For this we can split the challenge key into several parts- partial keys and maintain those keys in different cloud server and yet ensure security and data transparency. This might reduce the space overhead and possible cross verification of the verification process of a TPA by other TPAs.

V. ACKNOWLEDGEMENT

This work was supported in part by the US National Science Foundation under grant CNS-0831963, CNS-0626601, CNS-0716306, and CNS-0831628.

VI. REFERENCES

, [1] S.Sajithabanu and Dr.E.George Prakash Raj, "Data Storage Security in Cloud" IJCST Vol. 2, Issue 4, Oct. - Dec. 2011

[2] A. Jules and J. Burton S. Kaliski, "Pors: Proofs of retrievability for large files," in Proc. of CCS'07, Alexandria, VA, October 2007, pp. 584–597.

[3] H. Shacham and B. Waters, "Compact proofs of retrievability," in Proc. of Asiacrypt'08, volume 5350 of LNCS, 2008, pp. 90–107.

[4] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proc. of Secure Comm'08, 2008, pp. 1–10.

[5] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in Proc. of ESORICS'09, volume 5789 of LNCS. Springer-Verlag, Sep. 2009, pp. 355–370

. [6] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in Proc. of CCS'09, 2009, pp. 213–222.

[7] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring data storage security in cloud computing," in Proc. of IWQoS'09, July 2009, pp. 1–9.

[8] C. Wang, K. Ren, W. Lou, and J. Li, "Towards publicly auditable secure cloud data storage services," IEEE Network Magazine, vol. 24, no. 4, pp. 19–24, 2010.

[9] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "Mr-pdp: Multiple-replica provable data possession," in Proc. of ICDCS'08. IEEE Computer Society, 2008, pp. 411–420.

[10] M. Bellare, O. Goldreich, and S. Goldwasser, "Incremental cryptography: The case of hashing and signing," in Proc. Of CRYPTO'94, volume 839 of LNCS. Springer-Verlag, 1994, pp. 216–233. [11] Amazon.com, "Amazon Web Services (AWS)," Online at http://aws.amazon.com, 2008.