SECURE PUBLIC AUDITING WITH NETWORK CODING BASED STORAGE IN MULTI STORAGE CLOUD ARCHITECTURE

S. Aishwarya* & R. Sentamil Selvi**
* PG Scholar, Department of Computer Science and Engineering, M.I.E.T Engineering College, Tiruchirappalli, Tamilnadu
** Associate Professor, Department of Computer Science and Engineering, M.I.E.T Engineering College, Tiruchirappalli, Tamilnadu

Abstract:

Cloud computing setting in which probabilistic querying of outsourced data is a service provider. The data is to be revealed only to trusted users, not to the service provider or anyone else. Outsourcing offers the data owner scalability and a low initial investment. The need for privacy may be due to the data being sensitive or otherwise confidential. Security challenges and Data Loses are still among the biggest problem when considering the adoption of cloud services. This triggered a lot of research activities, resulting in a quantity of proposals targeting the various cloud security threats and repair Storage node but doesn't provide efficient security mechanism and fault isolation process. Now this proposed system will overcome for all existing problems. New public auditing scheme for regenerating-code-based cloud storage, which is solve the problem of failed authenticators in during the absence of data owners; to introduce a proxy, which is give privileged to authenticator to regenerate the data, into the traditional public auditing system model. To public verifiable authenticator is generated by a couple of keys and can be regenerated using partial keys. Thus, the scheme can completely release data owners from online burden. In addition, to randomize the encode coefficients with a pseudorandom function to preserve data privacy. The scheme is highly efficient and can be feasibly integrated into the regenerating-code-based cloud storage.

Key Words: Multi Storage Cloud, Regeneratingcode & Proxyserver

1. Introduction:

Cloud storage means massive clusters of server in large data centres for processing and storing data. The cloud storage can also considered as the virtual storage area that combines many different physical storage devices. The data centres are that store physical equipment used by the cloud. The single cloud storage providers are become less popular among customers due to risk of failure of service availability, loss of data and also single point of failure. Distributed cloud storage improves the fault tolerant of cloud storage with sufficient redundancy. The striping of data in distributed cloud storage using the conventional erasure coding can still leads to two types of failures like permanent and transient failure. Permanent failure where the data stored in failed storage node is permanently lost or unavailable. In this paper focusing code regeneration in multi storage cloud architecture, where repair is to reconstruct the lost data in failed node from other serviving node. With convention erasure code repair operation reconstruct the original data from failed node and stores in new node. The proposed of this paper is to introduce third party auditor and semi trusted proxy server on behalf of data owner. Proxy removes the data owner from online burden. Corrupted block can be regenerated using two versions of repair strategy: Exact repair and functional repair. Exact repair strategy store on exact replica of corrupted blocks, while functional repair indicates the new generated blocks are different from corrupted. Functional repair blocks are non-systematic.

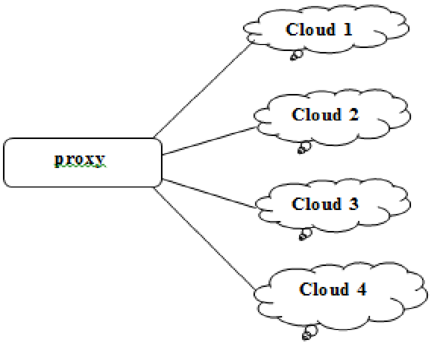

Figure 1: Normal Operation

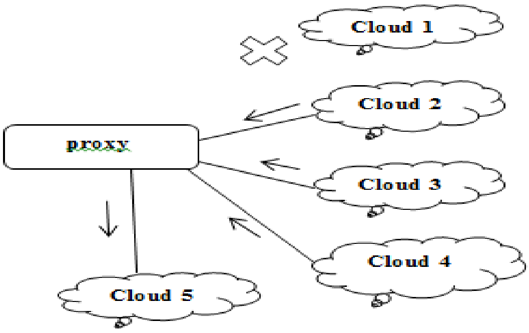

Figure 2: Repair operation when cloud 1 fails. Proxy regenerates the data for new cloud

A. Related Works:

Let us discuss some of the previous work done in this area. In single server seanario PDP-Provable Data Possession at untrusted storage and POR-Proves of Retrivability both to checking the integrity files on untrusted storage, but does not guarantee the retrivability of out sourced data. In PDP, to verify the server to possess the original data with out retrieving it.

It allows the server to access the small portion of file to generating the proof. In multi cloud architecture, HAIL-High availability and Integrity of Layer provide integrity and availability guarantees for stored data. In HAIL each file block is distributed across the 'n' server under dispersed code block. It maintain primary and secondary server, it maintain the redundancy of file block and also helps to recover the failed block from the storage. All above systems are built on erasure codes to provide fault tolerance, while NC-Cloud-network coding for cloud storage takes the regenerating codes, it consider both fault tolerance in storage repair. Optimal recovery for specific RAID-6 based on Reed Solomon. It reconstructs and stores the lost data in to new node. It reduces the amount of data read up to 25 percent. In FMSR-Functional Minimum Storage Repair code, can save 50 percent repair traffic while the node is increases. FMSR codes are non-systematic.

B. Contributions:

Introduce an algorithm for regenerating code. For using algorithm is pseudorandom function PRF is a collection efficient computable function which emulate a random oracle. PRF is not a pseudorandom generator. The guarantee of PRG is that single output appears the random if the input was chosen at random .PRF function is used during the file upload module.The coeffients are masked pseudorandom function to avoid the leakage of original data.

BLS (Boneh-lynn Shacham Signature) it allows user to verify the signer is authentic. The scheme uses the bilinear paring function for verification and signature. The signature scheme is provably secure that is the scheme is existentially unforgeable under adaptive chosen message attack

2. System Model:

System model contains of four entity data owner, cloud, TPA-Third party Auditor and Proxy agent. Data owner who owns the large amount of data to be stored on the cloud storage. Cloud, this is interring cloud storage, it maintains several storage servers, during the file uploaded process the uploaded files is spitted then the splitted files is stored at each storage server. TPA performs the auditing process it providing auditing results for both data owners and cloud servers. A proxy agent acts as a semi-trusted server behalf of data owner. Proxy who would always be online, to regenerate the authenticators and recover the failure blocks.

A. Setup Creation Module:

The Setup creation module fully worked on data owner side and also maintains the auditing scheme. 1. Data owner generating the couple of key that is public key and private key and it also generating the secret key of X. 2. Delegation: This algorithm is interaction between data owner and proxy. The data owner delivers the partial secret key X to the proxy through a secure approach. 3. Signature and block generation: This algorithm is by data owner and takes the secret parameter SK and the original file F is input and then output coded block set an authenticator set and a file tag T.

B. File Upload Module:

The file F split in to 'm' blocks, and the original m blocks is uploaded to the cloud storage then this file is managed by cloud file system. Then the original 'm' S-dimensional vectors each original block Wi is appended with the vector of length m containing a single 1 in the I' th position. The augmented vectors are encoded into coded blocks. They are linearly combined and generate coded blocks with randomly chosen coefficients vector. Original encoded files upload into several storage node and coefficient vector has been stored into proxy server.

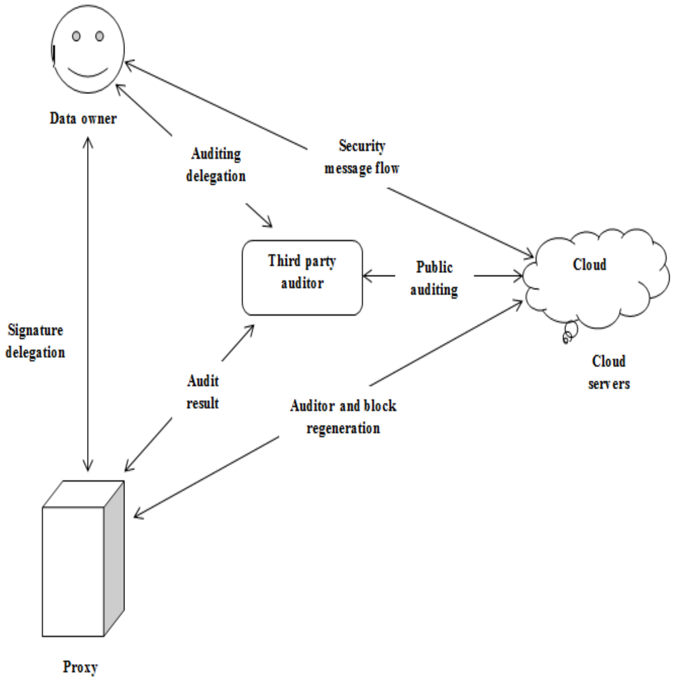

Figure 2: System Architecture

The data server doesn't response that time will check vector and regenerate the missing data into new storage.

C. Public Auditing Module:

The cloud servers and TPA interact with one another. 1. Challenge(C): This algorithm is performed by the TPA with the information of the file F as input and a challenge C as output. 2. Proof gen (P): This algorithm is run by each cloud server with input challenge C, coded block set and authenticator set, and then it output a proof P. 3. Verify (0,1): This algorithm is run by TPA immediately after a proof is received. Taking the proof P, public parameter pk and the corresponding challenge C as inputs 1 if the verification passed and 0 otherrwise.

D. Storage Node Re-Generation Module:

In the absence of data owner, proxy interacts with the cloud servers during the repair process. 1. Claim for Rep (Cr): This algorithm is similar with the challenge algorithm in the audit phase, but the outputs a clime for repair Cr. 2. Gen for Rep (BA): The cloud servers run this algorithm upon receiving the Cr and finally output he block and authenticators set BA with another two points. 3. Block and Sig Regeneration (Cr, BA): The proxy implements this algorithm with the clime Cr and the responses BA from each serves as input and outputs anew coded block set and authenticator set if successful outputting.

3. Performance Evaluation:

To evaluate the efficiency of code regeneration method is perfectly light weight for the data owner to execute. Because the auditing process is performed only once during whole life of users file. Public auditing improve the performance audit process. During the reparation process the delegation is send to the proxy server to repair the faulty blocks.

In auditing process performs the batch auditing; multiple users auditing request is handled simultaneously. Compared to separately audit the batch auditing improve the performance and speed.

4. Conclusion:

The public auditing scheme for the regenerating code based cloud storage system, where the data owners are privileged to delegate TPA, It randomize the coefficient s in the beginning rather than applying the blind technique during the auditing process. Considering that the data owner cannot always stay online, in order to keep the storage available and verifiable after a malicious corruption, It introduce a semi-trusted proxy into the system model and provide a privilege for the proxy to handle the reparation of coded blocks and authenticators . To better appropriate for the regenerating code scenario, for design based on the BLS signature.

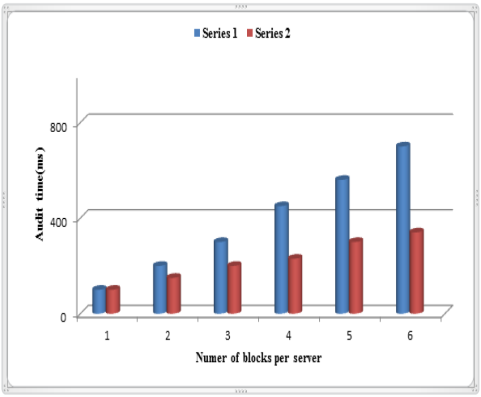

Figure 3: Time for Audit

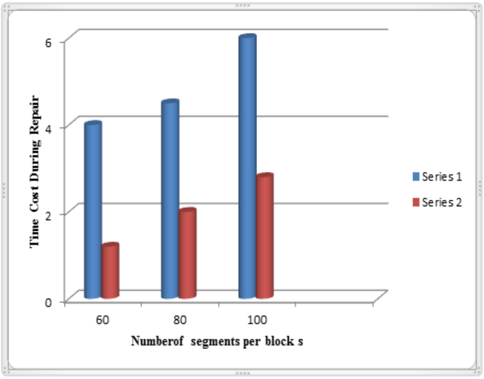

Figure 4: Number of Segmentation per Block

5. Future Enhancement:

In future, extend this work with different modelling techniques to build a more accurate model. This project aimed to collect information from any remote location in the absence of network connectivity and to recover the files in case of the file deletion or if the cloud gets destroyed due to any reason.

Client creates the file in cloud first time, it is stored at the main cloud. When it is stored in main server, the main file of client is being EX-OR with the Seed Block of the particular client. And that EX-OR file is stored at the remote server in the form of file'. If either unfortunately file in main cloud crashed / damaged or file is been deleted mistakenly, then the user will get the original file by EX-OR file with the seed block of the corresponding client to produce the original file and return the resulted file i.e. original file back to the requested client.

6. References:

1. Rmburst M., (2009), 'Above the cloud: Barkley view of cloud computing', in dept of elec engineering
2. Ateniese G., (2007), 'Provable data possession at untrusted stores', in proc 14th ACM conf, pp. 29-43.
3. Jules A., Kaliski B.S., (2007),'POR: Proofs of retrievability ACM Trans. Inf. Syst. Secur., vol. 9, no. 1, pp. 584-597.
4. Curtmola R. and Khan O. (2008), 'MR-PDP-Multiple replica in provable data possession', in Proc conf, pp. 411-420.
5. Bowers K.D., Jules A. and Oprea A. (2009), 'HAIL:High availability and integrity layer in cloud computing', in Communication security. pp. 187-198.
6. He J, Zhang Y, Shi Y and Cao.J. (2012), 'Distributed data possession checking for secure replicas in geographical dispersed clouds', vol. 78. pp. 1345-1358.
7. Chen B., Curtmola R, Ateniese, and Bruns R. (2010) 'Remote data checking for network coding based distributed storage systems', in Parallel distribution, pp. 31-42
8. Chen H.C.H. and Lee P.P.C. (2014), 'Enabling data integrity protection in regenerating code based cloud storage', in Parallel distribution, vol. 25. pp. 407-406.
9. Yang K., and Jia X. (2013), 'An efficient and secure dynamic auditing protocol for data storage in cloud computing', vol. 99. pp. 1717-1726.
10. Zhu Y., Hu H. and Yu M. (2012), 'Coperative provable data possession for integrity verification in multi cloud storage', in Parallel distribution vol. 23. pp. 2231-2244.