INTERNATIONAL JOURNAL OF SCIENTIFIC PROGRESS AND RESEARCH (IJSPR) ISSN: 2349-4689
Volume-19, Number - 03, 2016

Outsourcing Data Using Map-Based Provable Data Possession In Cloud Computing

S. J. Gayathri Pillai[1,] L. Sharmila[2]
[1]*P.G Student, M.E CSE, Alpha College of Engineering*
[2]*Assistant professor, Dept. of M.E CSE, Alpha College of Engineering*

Abstract – Outsourcing data to Cloud Service Provider (CSP) allows more and more organizations to store data on the CSP than on private computer systems. Outsourcing of data enables organization to many concentrate on the innovations than resolving the server updates and other computing issues. Customers can rent the CSPs storage infrastructure for storing and retrieving unlimited amount of data by paying fees based on gigabyte/month. The growth rate of internet usage is increasingly , so all the customers requires increased amount of availability, security, durability and some require multiple data to be replicated and stored in multiple sites. The CSP charges more fees if multiple data has to be stored. Therefore customer needs to have a strong guarantee that whether CSP is storing all the data that is been outsourced. For this purpose in the proposed scheme Map-based provable dynamic data possession is used that supports 1) Authentication 2)Identify the corrupted copies 3)Reduces the storage space for verification in both CSP and Verifier side 4)Dynamic operations such as insertion, deletion.

Keywords: Outsourcing data storage, Cloud computing, dynamic environment, data replication.

I. INTRODUCTION

Cloud computing allows the data owner to outsourced their data to remote CSP which may not be trustworthy; the data owner loses the direct control over their sensitive data. This lack of control leads the data owner to fear about the security issues including confidentiality and integrity. The confidentiality issues can be solved by encrypting the data before outsourcing it to the remote CSP, to avoid the sensitive data reaching to the unauthorized person. The integrity issues can be solved by incorporating the Provable Data Possession (PDP) scheme [1].

In a general PDP model, before storing the data on a server, the client must store, locally, some meta-data. At a later time, and without downloading data, the client is able to ask the server to check that the data had not been falsified. This approach is used for static data. The other mechanism of PDP is scalable PDP: This approach is premised upon a symmetric-key which is more efficient than public-key encryption. It supports some dynamic operations (modification, deletion and append) but it cannot be used for public verification [2].

A crucial demand of customers to have strong evidence that the cloud servers still possess their data and it is not being tampered with or partially deleted over time. Consequently, many researchers have focused on the problem of Provable Data Possession (PDP) and proposed different schemes to audit the data stored on remote servers.

PDP is a technique for validating data integrity over remote servers. In a typical PDP model, the data owner generates some metadata/information for a data file to be used later for verification purposes through a challenge-response protocol with the remote/cloud server.

The owner sends the file to be stored on a remote server which may be untrusted, and deletes the local copy of the file. As a proof that the server still possesses the data file in its original form, it needs to correctly compute a response to a challenge vector sent from a verifier who can be the original data owner or a trusted entity that shares some information with the owner.

The main principle of outsourcing data is to include dynamic behavior of data for various applications. This means that the outsourced data can be modified, inserted, deleted by the data owner.

The PDP schemes presented in [1] [2] mainly focus on the static data; it means the outsourced data cannot be changed over remote CSPs.The construction of PDP with dynamic data is presented in [4].

Data replication is the copying data from a database in one computer or server to other database of other. The result is Distributed database that is built where users can access the data in order to improve availability of data, thus reduces delay in accessing the data from Cloud. A weighted K–means clustering of user location used to determine replica site location for data backup.

Generating unique differentiable copies of data file is important concept in Provable multi-copy data possession. Identical copies enable the CSP to deceive the owner by copying only one file. Using a simple yet efficient way, the proposed scheme generates unique differentiable copies by using diffusion property ensures that output bits of cipher depend on input bits of plain text in very complex way.

II. SYSTEM MODEL

The cloud computing storage model consists of mainly three componenets 1) Data Owner– may be an oraganization or person who are the owner of their sensitive data. 2) CSP who provides the storage infrasturcture to the data owner to store their sensitive data and provides charges based on gigabyte /month. 3) Authorized users who have the right to access the remote data stored on CSP.

The data owner has the file F consisting of m blocks.The File F will be replicated into n copies if data owner wants to store multiple copies on different location based on users requirements for the critical data.The CSP pricing model is based on number of copies .

In order to include the confidentiality,the data owner encrypts the n copies of file before outsourcing it to the remote CSP.The data owner want to perform any block level operation then it must interact with the CSP.The block level operation includes insertion, deletion and modification of the outsourced n copies of data.

The authorized users access the data by sending the request to the CSP. The CSP use load balancing mechanism to reduce the congestion of the request been made by user.

The data owner or client, server and verifier interact with each other as shown in Fig:2.1.The proposed Scheme mainly runs seven polynomial time algorithms. In which data owner runs Key Gen, Copy Gen, Tag Gen and PrepareUpdate .The CSP runs ExecUpdate and Prove. The verifier runs Verify algorithms respectively.

The Data owner first runs KeyGen algorithm.The dataowner send request to verifier for public and private key.The verifier send the public and private key and it stores private key of the data owner.The CopyGen algorithm takes input as File and CNn ,n number of file copies and generates n copies of file.Inorder to distinguish n copies of file ,TagGen algorithm takes input as private key and the n copies of file and generates tag[6].

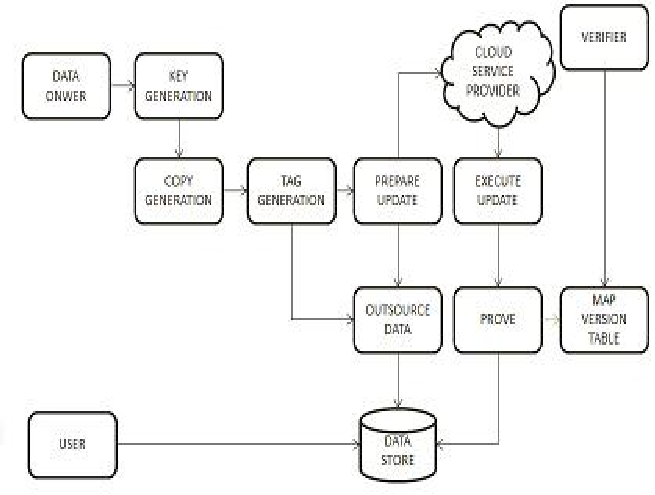

Fig:2.1 Architecture Diagram

The dataowner or client stores a meta data D’ in the form of map that is later used during verification process.Now the data is been outsourced to remote CSP.If data owner want to perform any block level operation then PrepareUpdate algorithm runs which take parameter as meta data D’ and update information such as insertion, deletion and modification.The data owner sends request to the CSP to store the updated data.The ExecUpdate algorithms takes input as request from data owner and store the updated data.

The verification process to check whether CSP still possess the data.the verifier runs verify algorithm which take parameter as private key ,meta data and generates a chalenge Chal that is sent to the CSP.The CSP runs Proof alogrithm by taking parameter as file copies F’, tag attached with the blocks of files and the Chal sent by the verifier.The CSP sends the Proof to the verifier.The verifier checks the proof by using meta data.the output is 1 if the integrity of F’ is correct otherwise it is 0.

III. PREVIOUS WORK

Provable data possession scheme supports integrity of outsourced data. The data owner stores Meta data about files before outsourcing, Meta data is generated by the information based on tag. The tag is mainly stored in the form of tree structure using MHTs (Merkle Hash Tree) .The concept of Merkle Hash Tree was explained by R.C. Merkle.

Merkle Hash Tree is a binary tree structure mainly used to verify the integrity of files. The MHT is a tree of hash values where all the leaves consist of hash values of data blocks. The Hash values h are created using SHA-2.If a file consists of 8 blocks then for which each blocks corresponding hash values will be generated namely h_1,h_2,h_3,...,h_8.Next the hash values will be concatenated to form higher level of hash values such as h_a= h($h_1||h_2$), h_b= h($h_3||h_4$), h_c= h($h_5||h_6$), h_d= h($h_7||h_8$).The concatenated hash values are again concatenated to form next higher level in the tree such as h_E= h($h_a||h_b$), h_F= h($h_c||h_d$), and then hash values h_E and h_F are concatenated to form the root hash value h_R. The CSP stores only the data blocks b1 to b8 of the file outsourced by CSP and verifier stores only the rooted hash value h_R.

During the verification process verifier needs to check integrity of file blocks namely b1 and b2, CSP will send theses two blocks b1 and b2 along with the authentication paths to the rooted hash value. The authentication paths are the edges from leaves node b1 and b2 to the root node h_R .

The verifier constructs the root of the corresponding blocks if the root hash values matches then the integrity of the files are correct.

The MHT can be used only to a single copy of data files, if MHT is used for multi-copy of data files. Then storage space, communication cost and computation time during verification process will be high.

The communication cost in Tree based PDP is more than Map based PDP as in Tree based the CSP as to send blocks and the authentication paths but in Map based only the tag value of particular blocks are sent.

Dynamic operation cannot be performed on MHT as it do not include a storage space for storing update information in the tree structure, but in Map based scheme it stores the update information on which blocks the modification has been done.

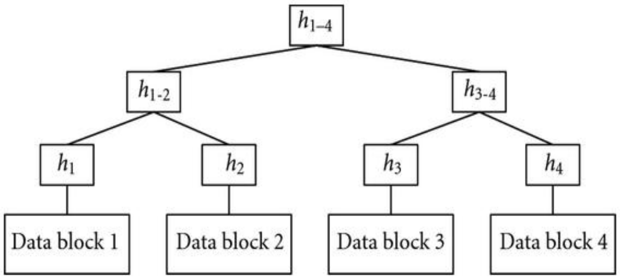

Fig: 3.1 Merkle Hash Tree for Table based PDP

IV. PROPOSED METHODOLOGY

The module consists of Cloud Infrastructure Formation, Data Replication, and Overview and Rationale, Algorithm Implementation, Map-Version Table. The cloud computing storage model considered in the proposed model consists of three main components, (i) a data owner that can be an organization originally possessing sensitive data to be stored in the cloud; (ii) a CSP who manages Cloud Servers (CSs) and provides paid storage space on its infrastructure to store the owner's files; and(iii) authorized users — a set of owner's clients who have the right to access the remote data.(iv) Verifier (original owner or any other trusted auditor) verifies the integrity of file copies.

Database replication is the frequent copying data from a database in one computer or server to a database in another so that all users share the same level of information. The result is a distributed database in which users can access data relevant to their tasks without interfering with the work of others. The implementation of database replication for the purpose of eliminating data ambiguity or inconsistency among users is known as normalization.

In data replication across datacentres with the objective of reducing access delay is proposed. The Optimal replication site is selected based on the access history of the data. A weighted k-means clustering of user locations is used to determine replica site location. The replica is deployed closer to the central part of each cluster.

The Map-Version Table (MVT) is a small dynamic data structure stored on the verifier side to validate the integrity and consistency of all file copies outsourced to the CSP. The MVT consists of three columns: Serial Number (SN), Blocks Number (BN), and Block Version (BV). The SN is an indexing to the file blocks. It indicates the physical position of a block in a data file. The BN is a counter used to make a logical numbering/indexing to the file blocks. Thus, the relation between BN and SN can be viewed as a mapping between the logical number BN and the physical position SN. The BV indicates the current version of file blocks.

It is possible to obtain a provable multi-copy dynamic data possession scheme by extending existing PDP models for single-copy dynamic data. Such PDP schemes selected for extension must meet the following conditions: (i) support of *full* dynamic operations (modify, insert, append, and delete), (ii) support of public verifiability, (iii) based on pairing cryptography in creating block tags (homomorphic authenticators); and (iv) block tags are outsourced along with data blocks to the CSP (*i.e.*, tags are not stored on the local storage of the data owner). Meeting these conditions allows us to construct a PDP reference model that has similar features to the proposed MB-PMDDP scheme. Therefore, establishing a fair comparison between the two schemes and evaluate the performance of the proposed approach.

V. SIMULATION/EXPERIMENTAL RESULTS

The comparison between Map based scheme and table based scheme from the point of storage, communication cost, and computation time.It has been reported that if the remote server is missing a fraction of the data, then the number of blocks that needs to be checked in order to detect server misbehaviour with high probability is constant independent of the total number of file blocks. For example, if the server deletes 1% of the data file, the verifier only needs to check for c = 460-randomly chosen blocks of the file so as to detect this misbehaviour with probability larger than 99%.

1) Proof Computation Time: For different number of copies the proof computation times (in seconds) provides evidence that the file copies are actually stored on the cloud servers in an updated, uncorrupted, and consistent state. The timing curve of the MB-PMDDP scheme is much less than that of the TB-PMDDP. For 20 copies, the proof computation times for the MB-PMDDP and the TB-PMDDP schemes are 1.51 and 5.58 seconds, respectively (≈ 73% reduction in the computation time).The verification timing curve of the MB-PMDDP scheme is *almost* constant. There is a very small increase in the verification time with increasing number of copies.This is due to the fact that although the term $s(\text{n} - 1)$ AZp in the verification cost of the MB-PMDDP scheme is linear in n. This feature makes the MB-PMDDP scheme computationally cost-effective and more efficient when verifying a large number of file copies.

TABLE III : OWNER COMPUTATION TIMES (SEC) DUE TO DYNAMIC OPERATIONS ON A SINGLE BLOCK

No of copies	1	2	3	4	5
Table based scheme	0.261	1.304	2.608	3.913	5.17
Map based scheme	0.261	1.305	2.61	3.916	5.21

2) Dynamic Operations Cost: For different number of copies. The owner computation times for both schemes are approximately equal. The slight increase of the TB-PMDDP Scheme is due to some additional hash operations required to regenerate a new directory root that constructs a new M (Meta Data).

As noted, the computation overhead on the owner side is practical. It takes about 5 seconds to modify/insert/append a block of size 4KB on 20 copies (< 1 minute for 200 copies).

In the experiments, we use only *one* desktop computer to accomplish the organization (data owner) work. In practice during updating the outsourced copies, the owner may choose to split the work among a few devices inside the organization or use a single device with a multi-core processor which is becoming prevalent these days, and thus the computation time on the owner side is significantly reduced in many applications.

VI. CONCLUSION

In this paper, data outsourced to remote CSPs has reduced the burden of data storage and maintenance for organizations. A map-based provable dynamic data possession helps in outsourcing of multi-copy where data owner can access, update and scale these copies on CSPs.

The proposed scheme is the first to address multiple copies dealing with dynamic data. The interaction between the authorized users and CSP is also considered in this scheme where the authorized users can access the data from CSP by using a single secret key shared with the data owner.

VII. FUTURE SCOPES

An insignificant modification can be done on the proposed scheme to support the feature of identifying the indices of corrupted copies. The corrupted data copy can be reconstructed even from a complete damage using duplicated copies on other servers. Through security analysis, Future scheme is provably secure with cryptography methods.

REFERENCES

[1] G.Ateniese et al.,"Provable data possession at untrusted stores," inproc. 14th ACM Conf. Comput. Commun.secur.(CCS),NewYork,NY,USA,2007,pp,598-609.

[2] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proc. 4th Int. Conf. Secur. Privacy Commun. Netw. (SecureComm), New York, NY, USA, 2008, Art. ID 9.

[3] K. D. Bowers, A. Juels, and A. Opera, "Proofs of retrievability: Theory and implementation," in Proc. ACM Workshop Cloud Comput. Secur. (CCSW), 2009, pp. 43-54.

[4] C. Elway, A. Kupcii, C. papamanthou, and R. Tamassia, "Dynamic provable data possession," in Proc. 16th ACM Conf. Comput. Commun.Secur. (CCS), New York, NY, USA,2009,pp. 213-222.

[5] P. Golle, S. Jarecki, and I. Mironov, "Cryptographic primitives enforcing communication and storage complexity," in Proc. 6th Int. Conf. Financial Cryptograph. (FC), Berlin, Germany, 2003, pp. 120–135.

[6] F. Sebé, J. Domingo-Ferrer, A. Martinez-Balleste, Y. Deswarte, and J.-J. Quisquater, "Efficient remote data possession checking in critical information infrastructures," IEEE Trans. Knowl. Data Eng., vol. 20, no. 8, pp. 1034–1038, Aug. 2008.

[7] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan, "Auditing to keep online storage services honest," in Proc. 11th USENIX Workshop Hot Topics Oper. Syst. (HOTOS), Berkeley, CA, USA, 2007, pp. 1–6.

[8] M. A. Shah, R. Swaminathan, and M. Baker, "Privacy-preserving audit and extraction of digital contents," IACR Cryptology ePrint Archive, Tech. Rep. 2008/186, 2008.

[9] E. Mykletun, M. Narasimha, and G. Tsudik, "Authentication and integrity in outsourced databases," ACM Trans. Storage, vol. 2, no. 2, pp. 107–138, 2006.

[10] H.Shancham and B.waters," Compact proofs of retrievability ," in Proc, 14th Int.Conf.Theory Appl. Cryptol. Inf.Secur. 2008, pp. 90-107.

[11] C.Wang, Q. Wang, K. Ren, and W.Lou.(2009). "Ensuring data storage security in cloud computing ,"IACR Cryptography PrintArchive ,Tech.Rep. 2009 /081.[Online].Available:http://ePrint.iacr.org/

[12] Y.Zhu, H.wang, Z. Hu, G, J.Ahn, H .Hu and S. S.Yau,"Efficient provable data possession for hybrid clouds,"in Proc, 17th ACM Conf. Comput. Commun. Secur. (CSS), 2010, pp. 756-758.