ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, January 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

Analysis of Integrity Checking Schemes in Cloud Computing

S.Ramila[1], R.Maheswari[2]

II-M.E, Department of CSE, Nandha College of Technology, Erode, Tamilnadu, India[1]

Assistant Professor, Department of CSE, Nandha College of Technology, Erode, Tamilnadu, India [2]

Abstract-Cloud Computing is a collection of servers hosted on the internet. User can request the cloud provider to get the services. One of the advantage of cloud storage is that the originality(integrity) of the data is maintained in the cloud. It can be achieved by various cryptographic techniques. This new data storage paradigm introduces some security challenges, which requires an auditing service to check the integrity of the data. In this paper, a various integrity checking schemes namely Provable Data Possession(PDP), Compact Proofs of Retrievability(CPOR), Dynamic Provable Data Possession(DPDP), Auditing and CPDP are analyzed and also conclude the best strategy to store the data.

Index Terms-Data Sharing, Data storing, Public auditability, Batch auditing, Cloud Computing.

I. INTRODUCTION

Cloud computing is a promising computing model that enables convenient and on-demand network access to a shared pool of computing resources. Cloud Storage is an important technique to store the data from their local system to the cloud. The cloud storage service (CSS) plays an important role in data storage. Because it is mainly designed to relieves the burden of storage management. It is managed by the cloud service provider (CSP). Even though it is efficiently designed it follows some of the security risks: First, the cloud computing infrastructures are suspected to the internal threats; next, there may be a chance for CSP to behave unfaithfully toward cloud users. Traditionally, owners can check the data integrity by two-party storage auditing protocols. In that user only allowed to verify their data by sending some challenge to the server. After verification, server will send proof to the

client. It is an inefficient technique to conduct auditing by both server and user. Because both of them could produce the inappropriate result. To overcome this situation, third-party auditing is developed. It should provide the following three properties: 1) Confidentiality- The auditing scheme should keep the data as more secret. 2) Dynamic auditing- It should support periodic updates of the data in the cloud. 3) Batch Auditing- When the request arise from the multi-user or from the multi-cloud, it should perform the batch operation.

A. Cloud Storage Architecture

It is depicted in the following fig 1.It contains the three different network entities. They are as follows:

- *Client*: An entity which has large data to store on the cloud and relies on the cloud for data maintenance and computation. Client may be of individual or grouped as organizations.
- *Cloud Storage Server* (CSS): An entity which is maintained by the cloud service provider.
- *Third Party Auditor*: An entity which is used to assess and expose cloud storage services on behalf of user request.

This paper is organized as follows: Section 2, discussion about PDP Scheme. Section 3, some theoretical description on CPOR. In Section 4, an efficient scheme of PDP called Dynamic PDP is discussed. In Section 5, efficient and secure auditing scheme is defined. This auditing scheme is extended to support dynamic operation is mentioned in section 6. In Section 7, Conclusion about some integrity checking schemes which based on data privacy and dynamic for multi owner and cloud.

II. PDP (PROVABLE DATA POSSESSION)

In this scheme[11], before the data storage in the cloud user have to produce the meta data for the respected file. Metadata can be stored locally in the client's system. User can send the data to the server for storage. It uses the challenge and response for the data verification.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, January 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

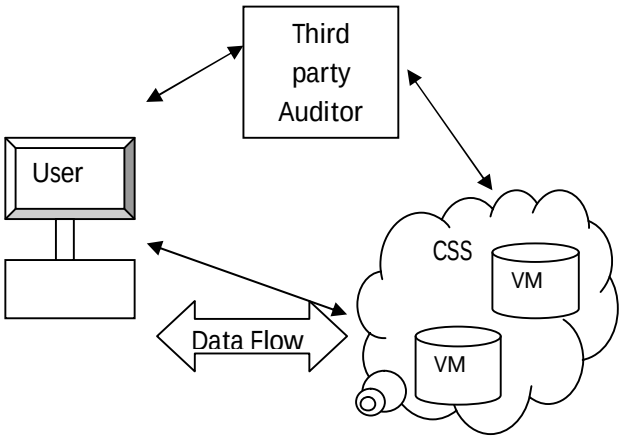

Fig 1: Cloud Storage Architecture

Main goal of PDP is to detect the misbehavior of server when the server has deleted a fraction of the file. It is also the efficient version of the Proof Of Retrievability (POR)[3]. Because it is suitable to the limited queries. By using homomorphic verifiable tags PDP can support to the large data bases. A PDP system can be constructed two phases, Setup and Challenge.

Setup: In this phase, The client C having the file F and runs KeyGen algorithm, followed by TagBlock for all blocks of the file. C stores the pair(sk, pk). sk-secret key, pk-public key. Client C then sends pk, F to the server S for storage.

Challenge: Client C generates a challenge(chal) algorithm for the data which needed the proof. C then sends chal to server S. Then server runs GenProof algorithm and sends proof to C. Finally, client can check the proof validity by generate CheckProof algorithm.

In the Setup phase, C computes tags for each block of the file and collectively stored at the server. In the Challenge phase, client can request the proof for a subset of the blocks in F .Finally client can check the validity of the proof. The following algorithms are used in setup and challenge phases of PDP.

A.Algorithm

KeyGen → (pk, sk): It is run by the client to set up the process. It takes some security parameter K that may identity of user and produce the output as secret(sk) and public(pk) keys.

TagBlock(pk, sk,m) → Tm: It is also run by the client. It takes input as public key, secret key and file block and returns the verification meta-data.

GenProof(pk, F, chal,) → V: It is run by the server in order to generate a proof. It takes a inputs of public key (pk), an ordered collection of file blocks, a challenge chal and meta-data. It returns a proof of possession V for the blocks in F that are determined by the challenge chal.

CheckProof(pk, sk, chal, V) → {"success", "failure"} is run by the client in order to validate a proof of possession. Input of the process are public key , a secret key, a challenge and a proof . It returns whether V is a correct proof of possession for the blocks determined by challenge(chal).

PDP having two schemes namely Secure PDP(S-PDP) and Efficient PDP(E-PDP).Difference in S-PDP is that, in setup phase random values are generated to maintain the data storage. In TagBlock an information about the block is stored additionally. E-PDP scheme reduce the computation overhead with compared to S-PDP. It produce the proof to sum of the blocks. Some of the drawbacks of PDP are, it support only the static data. It failed to support the Dynamic operations like insertion, deletion, modification. And also it doesn't support the batch operation.

II.CPOR (**C**OMPACTS **P**ROOFS **O**F **R**ETRIEVABILITY)

In a Proof of Retrievability(POR)[3], data storage center only responsible for give proof to the verifier. Compact POR is the advanced version of the POR. It describes about two methods, first method based on the Boneh–Lynn–Shacham(BLS). It uses short response and short request for both server and client. In this scheme, anyone can act as verifier. In second scheme, it fully based on the concept of pseudorandom functions (PRFs). It uses shorter server response than the first scheme but client response is larger. Both of the schemes use the homomorphic property. Storage auditing is needed to verify the user data. Cryptographic systems allow users to verify the data and store securely. For a security model system should have the following properties: 1) System should be efficient as possible in terms of communication and computation complexity. 2) It should provide the unbounded usage of the system.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, January 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

Table 1: Comparison of various Integrity Checking Schemes

Schemes	Data privacy	Dynamic Data	Multi-Owner	Multi-Cloud
PDP[11]	Yes	No	No	No
CPOR[12]	No	No	No	No
DPDP[13]	No	Yes	No	No
Audit[14]	Yes	Yes	Yes	No
IPDP[15],[16]	Yes	Yes	No	Yes

3) System should be stateless, the periodic data updates can be supported. Last two properties supports the public verifiability. In POR [3] author described MAC scheme to authenticate the data. It defined as follows:

A.MAC-Based construction

In that each stored files are encoded separately and each of them authenticated using the MAC. It is sufficient for the client to retrieve the data by using user's secret key. This method is depicted in [3]. In DPDP scheme, signatures are used instead of MAC.

B. Homomorphic Authenticator

Homomorphic encryption plays an important role in data storage. It is developed to improve the response of the MAC scheme. It aggregate the block as a whole to reduce the response length of the server. Because of these, time consumption will be reduced.

III. DPDP (**D**YNAMIC **P**ROVABLE **D**ATA **P**OSSESSION)

To overcome the drawbacks of the PDP this scheme[13] was developed. The main goal of DPDP is to support the dynamic updates of the user data. Let us consider a file F consists of n blocks, If modification is needed , user can insert a new block or modify the existing block. This solution is based on authenticated dictionaries. It can be ordered by using the rank information. This scheme will explained in the following section.

A.DPDP Scheme

It follows the same PDP steps. In First step it generate the key as public and secret keys. After the key generation user have to prepare the data for the updation, if any of the user want to update. In next step, user can perform the update operation (insert, delete). After that user can request the server to verify their data**.** To respond that request, server should send proof to the user. Algorithms used for DPDP are explained as follows:

A. Algorithm

Key generation, challenge and verification algorithms follows the same as in PDP.

PrepareUpdate(sk, pk, F, info,Mc): It is an algorithm run by the client to prepare the data for updation. It takes the input of secret key, public key, and information to be updated along with its previous metadata. It will bring the output as encoded file with its new metadata which will send to the server.

PerformUpdate(pk, Fi−1,Mi−1, e(F), e(info), e(M)): It is an algorithm run by the server after the arrival of the update request from the user. Input of this algorithm is old version of the file(Fi-1), the metadata and the encoded information of the file, info and metadata. The output is new version of the file(updated copy of old file), new metadata along with the metadata send to the client.

VerifyUpdate(sk, pk, F, info,Mc,M'c,PM'c): It is run by the client to verify the server update. It takes all the input of the PrepareUpdate algorithm along with the metadata. It produce the acceptance or rejection signals as a output.

IV. **A**UDITING

In all the previous schemes, user only responsible to verify their data. Clients are unreliable and may not be able to check the integrity frequently. To improve the verification process, another server named Third Party Auditor(TPA) to perform the verification of data

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, January 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

instead of client. All security models follow the public and private auditability. In private auditability user only allowed to verify their data. But in public auditability anyone (server or client) can perform the verification. In this scheme[14] it achieves both public auditability and data dynamics. It uses the Merkle Hash Tree construction to authenticate the blocks of the file. It also designed to support the multi user by using the bilinear aggregate signature. By the above, it can support multi tasks.

A. Third Party Auditor

It is mainly designed to reduce the burden of the user to store their data locally. TPA is also an server to check the data integrity periodically. Instead of user, TPA can challenge the server and get the It can verify the user data periodically on behalf of user.

B. Audit Scheme

In Existing PDP and POR models support only the static data. It uses index information I, due to these if there any change in the blocks, following to that block also will get modified. Due to this, it failed to support the dynamic updates of data. To overcome this, auditing scheme eliminate the index information. It uses the PKC based homomorphic authenticator. In that method, it uses BLS or RSA signature. It follows Four phases:

Setup: It uses the same PDP key generation algorithm. Identity of a user is considered as a input. After the processing step, client can get the secret and public keys.

Integrity Verification: The client or TPA can challenge the server to verify the data. Chal message can generated by random selection of a data. For the TPA challenge, server can send the proof by generate proof algorithm. Finally verifier can send 'true', if it is valid data, otherwise it will emit 'false' signals.

Dynamic operation: Dynamic operations like modification, insertion, and deletion. In the case of modification, user can send the modification request to server along with the block and its data. After receiving this, server will modify the respective block with the modified value. As a result, server will send proof to the user. In insertion, a new block is inserted after the some specified position in the file. An opposite operation is deletion, in that user requested blocks will be deleted.

Batch Auditing: By using the bilinear aggregate scheme, multiple user request can process simultaneously. Multiple request can be aggregated to form a single request. It can reduce the processing time.

VI. CPDP(**C**o-**O**perative **P**rovable **D**ata **P**ossession)

This scheme[15] is developed to support distributed environment. It is an extended version of PDP scheme. Multiple cloud service providers co-operatively store and maintain the client's data. It based on the homomorphic verifiable tags and hash index hierarchy. Clouds can be combined by using the Multi cloud tools(overt, VMware sphere).

DPDP scheme[13] is based on the merkle hash tree. But it is failed to produce the response for the multi-cloud infrastructure. To overcome this, homomorphic tags are used to aggregate the response as a single. In distributed storage environment criteria: 1)Usability Aspect-user can use the integrity check at any time. 2)Security Aspect-It should restrict some security attacks. 3)Performance Aspect-Communication and computation overhead should be low. It uses two techniques, Hash Index Hierarchy(HIH) which is used to combine the multiple CSP challenge to produce single response. Next, Hash Verifiable Response(HVR) it is used to support distributed cloud storage. This architecture also follow the same entities as Client, CSP and the Trusted third Party.TTP is used to store the data owner public key and parameters used for the verification process.

A . Dynamic Audit Services

It is designed for untrusted and outsourced storages. It support dynamic data operations and timely abnormal detection with the help of the fragmentation, random sampling and index hash table. It follow the traditional audit structure, user can host their data in the cloud. In the case of data preprocess, user have to generate the tag for the file and both of that should be placed in the CSP. Index hash Table only stored in the TPA.

In periodic sampling audit TPA can challenge server periodically. If user want to modify their data , they are allowed to change their data and update their copy in the cloud. After that, modified index hash table should be send to the TPA by the client. TPA can challenge the server for the respective modification.

VII. **C**ONCLUSION **A**ND **F**UTURE **W**ORK

In this paper, Different integrity schemes are analyzed with respect to the data privacy, data

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, January 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

dynamics, multi cloud and the multi owner. In table 1 all the schemes are compared, from that CPDP scheme is better to fulfill the user requirements. In future work, data loss during data storage should be addressed and also support the dynamic owner.

REFERENCES

[1] Y. Deswarte, J. Quisquater and A. saidane, "Remote Integrity Checking" Proc.Sixth working Conf. Integrity and Internal Control in Information Systems (IICIS), Nov. 2004.

[2] M. Naor and G.N. Rothblum, "The Complexity of Online Memory Checking," J. ACM, vol. 56, no. 1,article 2, 2009.

[3] A. Juels and B.S. Kaliski Jr., "Pors: Proofs of Retrievability for Large Files," Proc. ACM Conf. Computer and Comm. Security, P. Ning,S.D.C. di Vimercati, and P.F. Syverson, eds., pp. 584-597, 2007.

[4] T.J.E. Schwarz and E.L. Miller, "Store, Forget, and Check:Using Algebraic Signatures to Check Remotely Administered storage," Proc. 26th IEEE Int'l Conf. Distributed Computing Systems,p.12 ,2006.

[5] D.L.G. Filho and P.S.L.M. Barreto, "Demonstrating Data Possession and Uncheatable Data Transfer," IACR Cryptology ePrint Archive, vol. 2006, p. 150, 2006.

[6] F. Sebe´, J. Domingo-Ferrer, A. Martı´nez-Balleste´, Y. Deswarte, And J.-J. Quisquater, "Efficient Remote Data Possession Checking
In Critical Information Infrastructures," IEEE Trans. Knowledge Data Eng., vol. 20, no. 8, pp. 1034-1038, Aug. 2008.

[7] G. Yamamoto, S. Oda, and K. Aoki, "Fast Integrity for Large Data," Proc. ECRYPT Workshop Software Performance Enhancement for Encryption and Decryption, pp. 21-32, June 2007.

[8] M.A. Shah, M. Baker, J.C. Mogul, and R. Swaminathan, "Auditing to Keep Online Storage Services Honest," Proc. 11th USENIX Workshop Hot Topics in Operating Systems (HOTOS), G.C. Hunt, ed.,2007.

[9] C. Wang, K. Ren, W. Lou, and J. Li, "Toward Publicly Auditable Secure Cloud Data Storage Services," IEEE Network, vol. 24, no. 4,pp. 19-24, July/Aug. 2010.

[10] K. Yang and X. Jia, "Data Storage Auditing Service in Cloud Computing: Challenges, Methods and Opportunities," World Wide Web, vol. 15, no. 4, pp. 409-428, 2012.

[11] G. Ateniese, R.C. Burns, R. Curtmola, J. Herring, L. Kissner, Z.N.J.Peterson, and D.X. Song, "Provable Data Possession at Untrusted Stores," Proc. ACM Conf. Computer and Comm. Security, P. Ning, S.D.C. di Vimercati, and P.F. Syverson, eds., pp. 598-609, 2007.

[12] H. Shacham and B. Waters, "Compact Proofs of Retrievability," Proc. 14th Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology, J. Pieprzyk, ed., pp. 90-107, 2008.

[13] C.C. Erway, A. Ku¨ pc¸u¨ , C. Papamanthou, and R. Tamassia, "Dynamic Provable Data Possession," Proc. ACM Conf. Computer and Comm. Security, E. Al-Shaer, S. Jha, and A.D. Keromytis, eds.,pp. 213-222, 2009.

[14] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," IEEE Trans. Parallel Distributed Systems, vol. 22, no. 5,pp. 847-859, May 2011.

[15] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," Proc. IEEE INFOCOM, pp. 525-533, 2010.

[16] Y. Zhu, H. Hu, G. Ahn, and M. Yu, "Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231-2244, Dec. 2012.

[17] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S.S. Yau, "Dynamic Audit Services for Integrity Verification of Outsourced Storages in Clouds," Proc. ACM Symp. Applied Computing, W.C. Chu, W.E. Wong, M.J. Palakal, and C.-C. Hung, eds., pp. 1550-1557, 2011.

[18] K. Zeng, "Publicly Verifiable Remote Data Integrity," Proc. 10th Int'l Conf. Information and Comm. Security, L. Chen, M.D. Ryan, and G. Wang, eds., pp. 419-434, 2008.