ISSN:0975-9646

A Comparative Study: Various Approaches for Cloud Data Security

Richa Singh[1] Amit Kumar Sharma[2]

M. Tech Research Scholor[1] ,Assistant Professor [2]
Department of Computer Sc. & Engineering
PSIT, Kanpur[1] PSIT-COE, Kanpur [2]

***Abstract*— Cloud computing has been envisioned as the next-generation technology of IT industries. The Cloud is a platform where data owner remotely store their data in the cloud to enjoy the high quality applications and services. Cloud is a model where user is provided services by CSP (Cloud Service Provider) on pay per use base. In spite of its vitality, it exhibits various security flaws including integrity of data, data dynamics and data privacy affects the performance of a number of approaches in cloud storage. The CSP will manage the data of client at data centre. If there is large number of clients is there who using the services of cloud then the management of data at data centre will be difficult and even some time for their mutual benefit of CSP (limited space available at Data Centre) it can discard some data of client which is not used by the client for a long time. This paper will give the concise review of various approaches for cloud data security and their limitations.**

Keywords— Cloud Computing, Third party Auditor, MAC.

I. INTRODUCTION

Cloud Computing [7] is a model which provides a large number of applications under different topologies. It is the technology of building a robust data security between CSP and User. This technology is literally called Cloud Data Security. In this paper, we present an introduction to the Cloud computing, TPA, security algorithm of different papers with their limitations.

A. Cloud computing

Cloud computing [7] is a model which enables convenient, efficient, on-demand network access to a shared pool of configurable Computing resources (e.g. servers, networks, storage, services and applications) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud Computing is divided into further part i.e. Service models, Cloud Component for more understanding about cloud.

B. Types of Service Models in Cloud

Cloud service providers offer their services according to three fundamental models [7]. They are software as a service (SaaS), Platform as a service (PaaS) and Infrastructure as a service (IaaS).

1) Software as a Service (SaaS):

It is also referred to as Application or a Service Clouds. SaaS is the model which hosts the application as a service to its various cloud users via internet. The applications are accessible from various client devices through web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings. One of the biggest benefits of SaaS is, it helps in costing less money than actually buying the application. It provides with cheaper and reliable applications to the organization

2) Platform as a Service (PaaS):

It supply computational resources via a platform upon which applications and services can be urbanized and hosted. In other way, it supplies all the needed resources to build an application and service via the internet, without downloading or installing it. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

3) Infrastructure as a Service (IaaS): It referred as Resource Clouds generally provide resources which are managed and can easily be scaled up, as services to a variety of users. The success rate of data access defines the quality of these cloud servers. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).

C. Third Party Auditor

Third Party Auditor [8] is kind of inspector. There are two categories: private auditability and public auditability. Although private auditability can achieve higher scheme efficiency, public auditability allows anyone, not just the client (data owner), to challenge the cloud server for the correctness of data storage while keeping no private information. To let off the burden of management of data of the data owner, TPA will audit the data of client. It eliminates the involvement of the client by auditing that whether his data stored in the cloud are indeed intact, which can be important in achieving economies of scale for Cloud Computing. The released audit report would help owners to evaluate the risk of their subscribed cloud data services, and it will also be beneficial to the cloud service

provider to improve their cloud based service platform [4]. Hence TPA will help data owner to make sure that his data are safe in the cloud and management of data will be easy and less burdening to data owner.

II. LITERATURE SURVEY

Different factors such as integrity of data, data dynamics and data privacy affects.The performance of a number of approaches in cloud data storage. Each and every approach has merits and demerits which make them suitable for different applications.

A. Review of existing approaches

1) MAC(Message Authentication Code): It can be used to protect the data integrity. Data owners will initially locally maintain a small amount of MACs [10] for the data files which are to be outsourced. The data owner can verify the integrity by recalculating the MAC of the received data file when he/she wants to retrieve data and will compare it to the local pre computed value but if the data file is large, MACs cannot be employed.

2) Hash Tree: For large data file a hash tree [9] can be employed, where the leaves are hashes of data blocks and internal nodes are hashes of their children of the tree. The data owner only needs to store the root hash of the tree to authenticate his received data. But it does not give any assurance about the correctness of other outsourced data.

3) TPA (Third Party Auditor): It relieves the burden of data owner of local data storage and maintenance; it also eliminates their physical control of storage dependability and security, which traditionally has been expected by both enterprises and individuals with high service-level requirements. An auditing service helps to save data owner's computation resources and provides a transparent yet cost-effective method for data owners to gain trust in the cloud. It eliminates the involvement of the client through the auditing of whether his data stored in the cloud.
The author Abhishek Mohta, R. Sahu and L. Awasthi [8] have given algorithm which ensures data integrity and dynamic data operations. They have designed algorithm for data manipulation, insertion of record and record deletion. Insertion and manipulation algorithms inserts and manipulate data efficiently but in data deletion we can't identify the person who have deleted record, how and when means if any one deletes record then this algorithm can no longer work.

4) Indexing Scheme: If we trace every record by index we can easily identify which user is accessing the record and deleting the record as we have traced him by index [11].

5) PDP Method: The author Ateniese et al. [4] are the first who have considered the public adaptability in their defined \provable data possession. (PDP) method which ensures possession of data files on untrusted storages. For auditing outsourced data their technique utilizes the RSA-based homomorphic authenticators and suggests to randomly sample a few blocks of the file. However, in their scheme the public auditability demands the linear combination of sampled blocks which exposed to the external auditor.

The goal of a PDP scheme that achieves probabilistic proof of data possession is to detect server misbehavior when the server has deleted a fraction of the file.

Requirements and Parameters: The important performance parameters of a PDP scheme include

1) Computation complexity: The computational cost to pre-process a file (at C), to generate a proof of possession (at S) and to verify such a proof (at C);

2) Block access complexity: The number of file blocks ac-cessed to generate a proof of possession (at S);

3) Communication complexity: The amount of data transferred (between C and S).

Homomorphic Verifiable Tags (HVTs):
Given a message m (corresponding to a file block), we de-note by Tm its homomorphic verifiable tag. The tags will be stored on the server together with the file F. Homomorphic

Verifiable tags act as verification metadata for the file blocks and, besides being unforgeable, they also have the following properties:

1) Blockless verification: Using HVTs the server can construct a proof that allows the client to verify if the server possesses certain file blocks, even when the client does not have access to the actual file blocks.
2) Homomorphic tags: Given two values Tm_i and Tm_j , anyone can combine them into a value Tm_i+m_j corresponding to the sum of the messages $m_i + m_j$.

6) Random Mask Technique: The author Cong Wang et al. [5] used the public key based homomorphic authenticator and to achieve a privacy-preserving public auditing system for cloud data storage security while keeping all above requirements in mind, it uniquely integrates it with random mask technique. For efficiently handling multiple auditing tasks, the technique of bilinear aggregate signature can be explored to extend the main result into a multi-user setting, where TPA can perform multiple auditing tasks simultaneously.

7) Proof of retrievability: A keyed hash function hk(F) is used in Proof of retrievability (POR) [3]scheme. The verifier, pre-computes the cryptographic hash of F using hk(F) before archiving the data file F in the cloud storage, and stores this hash as well as the secret

key K. The verifier releases the secret key K to the cloud archive to check the integrity of the file F and asks it to compute and return the value of hk(F). The verifier can check for the integrity of the file F for multiple times by storing multiple hash values for different keys, each one being an independent proof.

Although this scheme is very simple and easily implementable the main drawback of this scheme is that it requires higher resource costs for the implementation.

Verifier has to store as many keys as the number of checks it wants to perform as well as the hash value of the data file F with each hash key. Computation of the hash value for even a moderately large data files can be computationally burdensome for some clients (PDAs, mobile phones, etc.). Each invocation of the protocol at archive requires the archive to process the entire file F. This processing can be computationally burdensome for the archive even for a lightweight operation like Hashing. Furthermore, it requires the prover to read the entire file F - a significant overhead for an archive whose intended load is only an occasional read per file, where every file to be tested frequently [1]. The author Ari Juels and Burton S. Kaliski Jr proposed a scheme "Proof of retrievability" for large files using "sentinels"[2]. In this scheme, only a single key can be used irrespective of the size of the file or the number of files unlike in the key-hash approach scheme in which many number of keys are used.

The archive needs to access only a small portion of the file F unlike in the key-hash scheme which required the archive to process the entire file F for each protocol verification. This small portion of the file F is in fact independent of the length of F. The schematic view of this approach is shown in Fig 1.

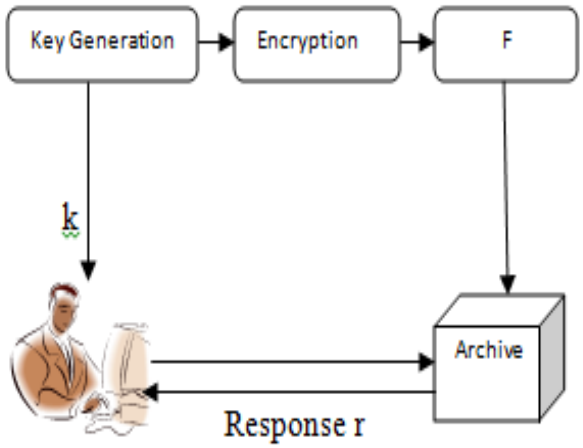

Fig. 1 Schematic view of a Proof Of Retriveability based on inserting random sentinels in the data file F.

In their scheme, Ari Juels and Burton S. Kaliski used special sentinels blocks, which are hidden among other blocks in the data file F. In initial phase, the verifier randomly embeds these sentinels among the data blocks. To check the integrity of the data file F, the verifier challenges the prover (cloud archive) during the verification phase by specifying the positions of a collection of sentinels and asks the prover to return the associated sentinel values. If the prover has modified or deleted a substantial portion of F, then with high probability it will also have suppressed a number of sentinels. Therefore it is unlikely to respond correctly to the verifier. To in distinguish the sentinels from the data blocks, the whole modified file is encrypted and stored in the archive. Here the use of encryption renders the sentinels indistinguishable from other file blocks. This scheme is best suited for storing encrypted files. It becomes computationally cumbersome to encrypt data file especially when the data to be encrypted is large as this scheme involves encrypting data file. Hence, this scheme has disadvantage that small users are left with limited computational power (PDAs, mobile phones etc.). This method also has storage overhead on the server, partly due to the newly inserted sentinels and partly due to the error correcting codes that are inserted. And the clients need to store all the sentinels with them, what may be storage overhead to thin clients (PDAs, low power devices etc.). It is not a practical solution to simply download the file for its integrity verification as it requires high cost of input/output and transmission cost across the network. Also it is not easy to check the data thoroughly and compare with our data. If we consider the large size of the outsourced data and the owners constrained resource capability, the task of auditing the data correctness in a cloud environment can be formidable and expensive for data owners. To fully ensure data security and save data owners, computation resources, we propose to enable publicly auditable cloud storage services, where to verify the outsourced data, the data owners can resort to an external TPA when needed. The TPA provides a transparent and cost-effective approach for establishing trust between client and cloud service provider. Based on the audit report of TPA, the released audit result would help the data owner to evaluate the risk of their subscribed cloud data services, and also beneficial for the CSP [6] to improve their cloud based service platform.

III. CONCLUSION

In this paper we explained different approaches for cloud data security. This paper includes the techniques/algorithms applied in various research papers with their merits and demerits. Here we describe the method of data security and privacy etc. In all those papers some papers haven't described data security mechanisms, some were lack in supporting dynamic data operations, some were lack in ensuring data integrity, while some were lacking by high resource and computation cost .Hence this paper gives overall description of all existing techniques for cloud data security and methods proposed for ensuring data authentication using TPA. The summary table includes all the algorithms/techniques their description with limitations.

TABLE 1. SUMMARY TABLE OF VARIOUS APPROACHES FOR CLOUD DATA SECURITY

Research Paper	Algorithms/Techniques	Description	Limitations
Privacy Preserved secure and dependable cloud data storage	MAC	Protect data integrity	Not applicable for large data files.
Secure hash Standard	Hash Algorithm	Store the root hash of the tree to authenticate his received data	Not provide assurance about the correctness of other outsourced data.
Robust data security for cloud while using third party auditor	RSA, SHA- 512	Design algorithm for data manipulation, insertion of record and record deletion.	If anyone deletes record then this algorithm can no longer work.
A secure index management scheme for providing data sharing in cloud storage.	Proxy- Re-encryption	Identify the user which accessing the record	
Provable Data Possession at untrusted store	RSA-based homomorphic authenticators	Ensure possession of data files on untrusted storage	May leak user data information to the auditor, when used directly.
Privacy preserving public auditing for secure cloud storage	Random Mask Technique	Bilinear aggregate signature explored to extend the main result, where TPA perform multiple auditing task simultaneously	
Towards Publicly Auditable secure cloud data storage services	Proof of retrievability	A single key used irrespective of the size of the file	Require high resource cost for implementation

REFERENCES:

[1] Qian Wang, Cong Wang, Kui Ren, Wenjing Lou, Jin Li,‖ Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing‖ in IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, Vol. No. 22, Issue 5, MAY 2011.

[2] Cong Wang and Kui Ren, Wenjing Lou, Jin Li,‖Toward Publicly Auditable Secure Cloud Data Storage Services‖ in IEEE Network July/August 2010.

[3] A. Juels, J. Burton, and S. Kaliski, - Proofs of Retrievability for Large Files, Proc. ACM CCS =07, Oct. 2007, pp. 584–97.

[4] G.Ateniese et al., ―Provable Data Possession at Untrusted Stores Proc. ACM CCS .07, Oct. 2007, pp. 598.609.

[5] Cong Wang, Qian Wang, Kui Ren, and Wenjing Lou,‖Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing‖ in IEEE INFOCOM 2010, San Diego, CA, March 2010.

[6] Federal Information Processing Standards (FIPS) 140-2. (2001, May 25). Security Requirements for Cryptographic Modules. Gaithersburg, MD: National Institute of Standards and Technology (NIST).

[7] Voorsluys, William; Broberg, James; Buyya, Rajkumar (February 2011) "Introduction to Cloud Computing". Cloud Computing : Principles and Paradigms. New York, USA: Wiley Press.pp.1–44.ISBN978-0-470-88799-8.

[8] Abhishek Mohta,Ravi Kant Sahu and LK Awasthi, "Robust Data Security for Cloud while using Third Party Auditor" in International Journal of Advanced Research in Computer Science and Software Engineering. Volume 2, Issue 2, February 2012.

[9] Bhavana Makhija, VinitKumar Gupta, Indrajit Rajput, "Enhanced Data Security in Cloud Computing with Third Party Auditor" in International Journal of Advanced Research in Computer Science and Software Engineering. Volume 3, Issue 2, February 2013.

[10] B.Dhivya, L.M.Nithya "Privacy Preserved Secure and Dependable Cloud Data Storage" in International Journal of Computer Science and Management Research,NCNICS 2013 Issue.

[11] Sun-Ho-Lee, Im-Yeong Lee "A Secure Index Management Scheme for Providing Data Sharing in Cloud Storage" in J Inf Process System,Vol.9, No.2, June 2013