IJERST

International Journal of Engineering Research and Science & Technology

ISSN : 2319-5991
Vol. 4, No. 2
May 2015

www.ijerst.com

Email: editorijerst@gmail.com or editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 4, No. 2, May 2015
© 2015 IJERST. All Rights Reserved

Research Paper

THIRD PARTY AUDITING FOR CLOUD STORAGE

Kedar Jayesh Rasal[1]* and Sandip A Kahate[1]

*Corresponding Author: Kedar Jayesh Rasal ✉ kdr.rasal@rediffmail.com

With data storage and sharing services provided by the cloud, people can easily work together as a group by sharing data with each other. The Cloud provide platform where all users not only store their data but also used the software and services provided by Cloud Service Provider (CSP). In cloud storage system, the clients store their data in the server (often locations) without keeping a local copy. Clients store their data in the private clouds but when storage expansion is needed they move to public clouds. Security is the major concern in the public clouds. The security mechanisms for private and public clouds are different. It may be possible that an unauthorized user can access the data from the public clouds .It is very important that the client should be able to verify the integrity of the data stored in the remote un-trusted server. There may be security services offered by public clouds but they are not sufficient. In order to address the security issues, Trusted Third Party Auditing (TPA) is used as a service for private and public clouds, which offers various services to check for the integrity of the data. TPA mechanisms offer various auditing mechanisms such as read, write, update to verify the integrity of the data stored in the public clouds. This paper discusses an auditing model based on Merkle Hash Tree. This paper conducts a study on possible auditing mechanisms which can be offers as a service over hybrid/public clouds.

Keywords: Third Party Auditor (TPA), Data storage, Integrity, Public auditability, Cloud computing, Cloud service provider

INTRODUCTION

The present availability of high-capacity networks, low-cost computer and storage devices as well as the widespread adaption of hardware virtualization, service-oriented architecture and autonomic and utility computing have led to growth in cloud computing. By the use of virtualization and resource time-sharing, clouds address with a single set of physical resources a large user base with different needs. The service provided by the cloud is very economical. The user pay only for what he used. This is a platform where data owner remotely store their data in the cloud to enjoy the high quality applications and services. The user can access the data, use the data and store the data. In a Corporate world there are large

[1] Computer Engineering, Pune University, SPCOE, Otur, Pune, Maharashtra, India.

numbers of clients who access their data and modify their data.

The ever cheaper and more powerful processors, together with the "Software as a Service" (SaaS) computing architecture, are transforming data centers into pools of computing service on a huge scale. Meanwhile, the increasing network bandwidth and reliable and flexible network connections make it even possible that clients can now subscribe high-quality services from data and software that reside solely on remote data centers.

The integrity of data in cloud storage as data stored in an untrusted cloud can easily be lost or corrupted, due to hardware failures and human errors. For example, the storage service provider, which experiences Byzantine failures occasionally, may decide to hide the data errors from the clients for the benefit of their own. What is more serious is that for saving money and storage space the service provider might neglect to keep or deliberately delete rarely accessed data files which belong to an ordinary client. Consider the large size of the outsourced electronic data and the client's constrained resource capability, the core of the problem can be generalized as how can the client find an efficient way to per-form periodical integrity verifications without the local copy of data files (Sivachitralakshmi and Judgi, 2012; and Govinda *et al.*, 2012). TPA is the third party auditor who will audit the data of data owner or client so that it will let off the burden of man-augment of data of data owner.

New data storage paradigm in "Cloud" brings about many challenging design issues which have pro-found influence on the security and performance of the overall system. One of the biggest concerns in cloud data storage is data integrity verification at entrusted servers. TPA eliminates the involvement of the client through the auditing of 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 not only help owners to evaluate the risk of their subscribed cloud data services, but also be beneficial for the cloud service provider to improve their cloud based service platform (Nandeesh *et al.*, 2012; and Wang Shao-hui *et al.*, 2012).This public auditor will help the data owner that his data are safe in cloud.

To protect the integrity of cloud data, it is best to perform public auditing by introducing a Trusted Third Party Auditing (TPA), which offers its auditing service with more powerful computation and communication abilities. With the use of TPA, management of data will be easy and less burdening to data owner but without encryption of data, how data owner will ensure that his data are in a safe hand. The users may resort to TPA for ensuring the storage security of their outsourced data, while hoping to keep their data private from TPA. Security aspects like data integrity, confidentiality, and non-repudiation are seen in public cloud not much in private cloud, because the private cloud does not allow the unauthorized users to access the data but the public cloud can be accessible and seen by everyone. When n numbers of user are using the data than consistency of data is quite important because anyone can use the data, modify the data or delete the data. If situation arise where one is writing a data while one is reading than it may be wrong read by second user .So to resolve the data dynamics is be-come an important task of the data owner. This paper added the information of insertion, updation and deletion in the message (Shacham and Waters, 2008; and Qian Wang *et al.*, 2009).

RELATED WORK

In Cloud, application software and services are move to the centralized large data center and management of this data and services may not be trustworthy. For example, clouds may become for scientists an alternative to clusters, grids, and parallel production environments (Mortaza Mokhtari Nazarlou and Javad Badali, 2012).

Cloud Software as a Service (SaaS): The capability pro-vided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email). 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 set-tings (Irfan Gul *et al.*, 2011; Ateniese *et al.*, 2008; and Qian Wang *et al.*, 2009).

In the context of remotely stored data verification the growing interest has been pursued. Considering the role of the verifier in the model, the schemes presented earlier fall into two categories: private auditability and public auditability (Mortaza Mokhtari Nazarlou and Javad Badali, 2012). Schemes with private auditability can achieve higher scheme efficiency, public auditability allows anyone, not just the client (data owner), to challenge the cloud server for correctness of data storage while keeping no private information (Mortaza Mokhtari Nazarlou and Javad Badali, 2012; and Boyang Wang *et al.*, 2014).

The TPA mechanism that utilizes public key based homomorphic authenticator with random masking. In Qian Wang *et al.* (2011) privacy preserving TPA is proposed. In this paper (Cong Wang *et al.*, 2010) encryption is done at user level and then data is sent to TPA, so that data is kept secured against TPA. TPA mechanism proposed in Sivachitralakshmi and Judgi (2012) uses digital signature method for auditing the data on the cloud.

The research work conducted in Irfan Gul *et al.* (2011) proposes auditing mechanism that uses homomorphism token and distributed erasure coded data. The most recent work conducted in Wang *et al.* (2009) and Gayatri (2012) proposes Trust Enhanced Third Party Auditor (TETPA) in which cloud service providers' accountability is enabled and protects cloud users' benefits. The existing schemes aim at providing integrity verification for different data storage systems, the problem of supporting both public auditability and data dynamics has not been fully addressed (Sivachitralakshmi and Judgi, 2012; and Govinda *et al.*, 2012).

CLOUD ARCHITECTURE WITH TPA

In the figure shows a model for TPA which is used for auditing between public cloud and private cloud with Client, CSP and TPA. The client asks the CSP to provide service where CSP authenticate the client and provide a virtual machine by means of Software as a service. In this Virtual Machine (VM), RSA algorithm are used where client encrypt and de-crypt the file. In this VM, SHA-512 algorithms also there which make the message digest and check the integrity of data. The figure has following entities:

Client: An entity which has large data files to be stored in the cloud and relies on the cloud for data maintenance and computation can be either individual customers or organization clients are

Classified into 2 groups such as:

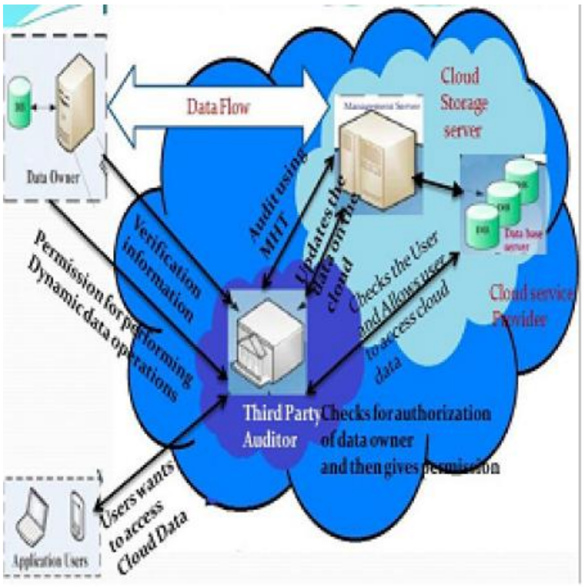
1. Data Owner: Have a large amount of data to be stored in the cloud.
2. Application Users: Users who can access the applications with proper access permission

Cloud Storage Server: An entity managed by CSP to provide data Storage Service.CSS is divided into two components. Management Server, manages the server and Data Server, Stores the clients data

Cloud Service Provider: Is an Entity Which has significant storage apace and computation recourse to maintain clients data

Third Party Auditor: Has capabilities to manage or monitor – outsourced data under the delegation of data owner. Hash values of the files are stored at TPA.

Figure 1: TPA Which is Used for Auditing Between Public Cloud and Private Cloud


Data Owner
DB
Data Flow
Management Server
Cloud Storage server
Data base server
Cloud service Provider
Audit using MHT
Updates the data on the cloud
Verification information
Permission for performing Dynamic data operations
Checks the User and Allows user to access cloud data
Third Party Auditor
Checks for authorization of data owner and then gives permission
Users wants to access Cloud Data
Application Users

USER LEVEL CRYPTOGRAPHY

After performing file operation it will send the data to CSP and TPA. This CSP and TPA will keep our data not only safe but also provide integrity but how it does not ensure that one will full trust on TPA. He can send data's of data owner to unauthorized user. If we remove the TPA even it will not solve the problem because CSP can also send the data to unauthorized user and also data owner does not get an advantage of TPA. So cryptography is required at user level. In this scheme encryption and decryption is done with the help of RSA algorithm. For supporting data dynamics when data owner got services from CSP than at that time it will generate a two large prime number as a key, i.e., *Puk* and *Prk*. *Puk* is the public key of Data owner where all clients will use this key as encryption and *Prk* is the private key of Data Owner or Client. *Prk* will be used to decrypt the file. *Puk* will be same for all users but *Prk* is different for the entire user. Data owner first generate his public key and private key. His public key will be same for entire user. After generation of keys by data owner or client he will encrypt the file F to F'. This F' is an encrypted file. This encrypted file will reduce the understanding of message for not only unauthorized user but also for TPA. Decryption will also be done at client side with the help of his private key *Prk* he will decrypt the file. Integrity of data check mechanism as data owners no longer physically possess the storage of their data, cryptographic primitives for the purpose of data security protection cannot be directly adopted. In particular, simply downloading the file for its integrity verification is not a practical solution due to the high cost of input/output (I/O) and transmission cost across the network. Also it is not easy to check the data thoroughly and compare with our data. Even the loss of data and recovery of data is also not easy. Considering the large size of the outsourced data and the owner's constrained resource capability, the tasks of auditing the data correctness in a cloud environment can be formidable and expensive for data owners.

Hence, to fully ensure data security and save data owners' computation resources, we propose to enable publicly auditable cloud storage services, where data owners can resort to an external Third Party Auditor (TPA) to verify the outsourced data when needed. In fact, based on the audit result from a TPA, the released audit report would not only help owners to evaluate the risk of their subscribed cloud data services, but also be beneficial for the cloud service provider to improve their cloud based service platform.

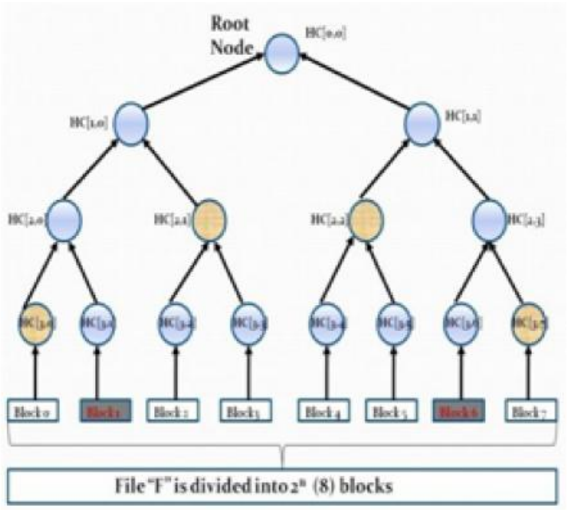
If data send by the client or data owner are not correct or transmission error or any error then how will found the accountability of data owner or client. To ensure that data reach to a CSP is in correct form and also send by the authenticate user we proposed a new scheme. This message digest will be made with the help of SHA-512 algorithm. Digital signature will be used as a client's or data owner identity. In case of any failure at client or data owner side digital signature will resolve the problem of accountability. Message Digest will helps in ensuring integrity of data. After a certain period of time TPA can check the data for integrity and reliability.

MERKLE'S HASH TREE

Merkle's hash tree construction built on the technique of Lamport's one-time signatures. Such a signature requires a setup stage in which many secret values are selected and the results of applying a hash function to each of them are published. The storage associated with the original one-time signature scheme grows too large to be practical for general use. Merkle proposed a method to sign multiple messages without the enormous cost of storing two secret values per bit to be signed. His construction makes use of a binary tree, where each node is associated with a bit-string. The bit strings associated to each leaf are the hash of a secret value associated to that leaf, and each internal node of the tree is assigned HASH (L||R), where L||R represents the concatenation of the values assigned to the left and right child nodes. Rather than randomly generating and storing the secret values for each leaf, the i'th secret value can be determined from a pseudo-random generator as PRNG(secret, i). The root of the tree is made public. This key generation process is time consuming, but is a one-time cost.

For a tree of height H, Merkle's scheduling algorithm required only O(H) HASH evaluations per round, and space to store O(H^2) intermediate hash values. For medium-size trees this original algorithm already embodied a reasonable degree of storage and computation efficiency.

Figure 2: Merkle's Hash Tree



DATA INTEGRITY VERIFICATION THROUGH TPA

The Client or TPA can verify the integrity of outsourced data by challenging the server. Before challenging, the TPA first uses spk to verify signature on t. If verification fails, reject by emitting FALSE else recover u. To generate the message “chal” the TPA (Verifier) picks a random c-element subset I = {s1, s2, ..., sc} of set [1, n] where scheme assume s1d”......d”sc. For each i “ I the TPA chooses a random element νi ← B⊆ Zp. The message chal specifies the positions of the blocks to be checked in the Merkle hash tree. The verifier sends the chal to the {(i, νi)} s1 ≤ i ≤ sc prover (server).

ALGORITHMS

Algorithm for Data Integrity Verification

- Start
- TPA generates a random set
- CSS computes root hash code based on the filename/blocks input
- CSS computes the originally stored value
- TPA decrypts the given content and compares with generated root hash
- After verification, the TPA can determine whether the integrity is breached
- Stop

Algorithm for Updating and Deleting Data Present in CSS

- Start
- Client generates new Hash for tree then sends it to CSS
- CSS updates F and computes new R’
- Client computes R
- Client verifies signature. If it fails output is FALSE
- Compute new R and verify the update and
- Stop

CONCLUSION

As market grows the threat on data also grows. To protect the data from unauthorized access and to ensure that this paper provides solution to the problem of integrity, unauthorized access, privacy and consistency. Although cloud providers promise a more secure and reliable environment to the users, the integrity of data in the cloud may still be compromised, due to the existence of hardware/ software failures and human errors. To protect the integrity of data in the cloud, a number of mechanisms have been proposed. In these mechanisms, a signature is attached to each block in data, and the integrity of data relies on the correctness of all the signatures. One of the most significant and common features of these mechanisms is to allow a public verifier to efficiently check data integrity in the cloud without downloading the entire data, referred to as public auditing (or denoted as Provable Data Possession would like to utilize cloud data for particular purposes (e.g., search, computation, data mining, etc.), or a Third Party Auditor (TPA) who is able to provide verification services on data integrity to users. This paper discusses the cloud architecture with TPA. The security in cloud compu-ting is very much needed as data in the cloud storage are not secure and require lots of attention of user.

REFERENCES

1. Abhishek Mohta and Lalit Kumar Awasthi (2012), “Cloud Data Security while Using

Third Party Auditor", *International Journal of Scientific & Engineering Research*, Vol. 3, No. 6, ISSN: 2229-5518.

2. Ateniese G *et al.* (2008), "Scalable and Efficient Provable Data Possession", Proc. Secure Comm 08, September.
3. Balakrishnan S, Saranya G, Shobana S and Karthikeyan S (2011), "Introducing Effective Third Party Auditing (TPA) for Data Storage Security in Cloud", *IJCST*, Vol. 2, No. 2.
4. Boyang Wang, Baochun Li and Hui Li (2014), "Public Auditing for Shared Data with Efficient User Revocation in the Cloud".
5. Cong Wang, Kui Ren, Wenjing Lou and Jin Li (2010), "Toward Publicly Auditable Secure Cloud Data Storage Services", in *IEEE Network*, July/August.
6. Gayatri R (2012), "Privacy Preserving Third Party Auditing for Dynamic Data", *International Journal of Communications and Engineering*, Vol. 01, No. 1, Issue. 03.
7. Govinda K, Gurunathaprasad V and Sathishkumar H (2012), "Third Party Auditing for Secure Data Storage in Cloud Through Digital Signature Using RSA", *International Journal of Advanced Scientific and Technical Research*, Vol. 4, Issue 2, ISSN: 2249-9954.
8. Irfan Gul, Atiq ur Rehman and Hasan Islam M (2011), "Cloud Computing Security Auditing".
9. Mortaza Mokhtari Nazarlou and Javad Badali (2012), "A New Case for Trusted Third PartyAuditor in Cloud Computing", *European Journal of Scientific Research*, Vol. 95, No. 1, pp. 152-157, ISSN: 1450-216X/1450-202X, Euro Journals Publishing, Inc.
10. Nandeesh B B, Ganesh Kumar R and Jitendranath Mungara (2012), "Secure and Dependable Cloud Services for TPA in Cloud Computing", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Vol. 1, No. 3, ISSN: 2278-3075.
11. Qian Wang, Cong Wang, Kui Ren, Wenjing Lou and Jin Li (2009), "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", in Proc. ESORICS 09, September, pp. 355-370.
12. Qian Wang, Cong Wang, Kui Ren, Wenjing Lou and Jin Li (2011), "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", in *IEEE Transactions on Parallel and Distributed Systems*, Vol. 22, No. 5.
13. Shacham H and Waters B (2008), "Compact Proofs of Retrievability", Proc. Asia-Crypt 08, LNCS, Vol. 5350, December, pp. 90-107.
14. Shah M A *et al.* (2007), "Auditing to Keep Online Storage Services Honest", Proc. USENIX HotOS 07, May.
15. Sivachitralakshmi S and Judgi T (2012), "A Flexible Distributed Storage Integrity Auditing Mechanism in Cloud Computing", International Conference on Computing and Control Engineering (ICCCE 2012), April 12 & 13.
16. Ushadevi R and Rajamani V (2012), "A Modified Trusted Cloud Computing Architecture Based on Third Party Auditor (TPA) Private Key Mechanism", *International Journal of Computer Applications (0975–8887)*, Vol. 58, No. 22.

17. Wang Shao-hui, Chang Su-qin, Chen Danwei and Wang Zhi-we (2012), “Public Auditing for Ensuring Cloud Data Storage Security with Zero Knowledge Privacy”.

18. Wang C *et al.* (2009), “Ensuring Data Storage Security in Cloud Computing”, Proc. IWQoS 09, July, pp. 1-9.

19. Xinmiao Zhang and Keshab K Parhi (2004), “High-Speed VLSI Architectures for the AES Algorithm”, in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 12, No. 9.

International Journal of Engineering Research and Science & Technology
Hyderabad, INDIA. Ph: +91-09441351700, 09059645577
E-mail: editorijlerst@gmail.com or editor@ijerst.com
Website: www.ijerst.com

ISSN 2319-5991
02
9 772319 599001