International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 6, Issue 6, November- December 2017 ISSN 2278-6856

PAS-Cloud : Toward a Framework With privacy preserving public Auditing Strategies for Heterogeneous Data in cloud storage

[1]. Akheel. Mohammed, [2]. Dr.D.Vasumathi

[1].Research Scholar, Asst. Professor Department of CSE, Shadan Woman's College of Engineering and Technology.

[2]. Professor Department of CSE, JNTUH, Kukatpally, Hyderabad.

Abstract

As enterprises cannot maintain local copy of outsourced data due to local storage issues, it is essential to have secured and privacy preserving means of outsourcing. Otherwise the data owners do not trust cloud service providers. Efficient auditing techniques are important to ensure privacy, security and data integrity. Many public auditing schemes came into existence. Recently, proposed a novel public auditing scheme based on Dynamic Hash Table. Third Party Auditor (TPA) in their scheme cannot view the data while performing auditing. However, most of the existing public auditing schemes do not have mechanisms differently for different type of data outsourced to public cloud. Structure, unstructured and semi-structured data can exist in the outsourced cloud storage. One size does not fit all as no single audit method can satisfy all types of cloud data. Therefore it is the new research trend to design a comprehensive framework for public auditing strategies for structured and unstructured outsourced cloud data. In this paper, we propose such framework which supports multiple public auditing strategies to cater to heterogeneous types of data. It is named as PASCloud. It framework consists Dynamic Hash Table (DHT) based public auditing for unstructured data and a verifiable auditing scheme (VAS) for outsourced structured data or relational database. The former can detect data integrity issues caused by CSP either intentionally or unintentionally while the latter can withstand intentional misbehaviour of CSP when a query is made to outsourced relational database. Amazon AWS cloud platform is used for experiments. The experimental results revealed the utility of the framework in rendering auditing services for user's unstructured outsourced data while the implementation for structured data is deferred for our future work.

Index Terms – Cloud computing, structured data, unstructured data, Dynamic Hash Table (DHT), Verifiable Auditing Scheme (VAS), privacy preserving public auditing

1. INTRODUCTION

Cloud computing is the Internet based computing which provides inexhaustible computing resources in pay per use fashion. Its storage infrastructure is widely used [1]. Many cloud service providers such as Amazon, Google, Microsoft and IBM to mention few exist to render different services such as Infrastructure as a Service (IaaS), Software as a Service (SaaS) and Platform as a Service (PaaS). Though cloud storage is viable and preferred for storing huge amount of data or big data, it throws many security challenges. One of the important challenges is data security and privacy. It is the concern of data owners as they do not maintain local copy of data when data is outsourced to public cloud [10].

There are many reasons for loss of data or loss of privacy. Cloud Service Provider (CSP) may intentionally steal data. CSP may hide inherent security flaws in the infrastructure [9]. CSPs may even delete data of users which is not frequently accessed [32]. Many solutions came into existence for cloud storage security and data integrity and preserving privacy as explored in [3], [10], [32], [33], [34]. The computational cost of the schemes was found to be more in those schemes except that of [10] where Dynamic Hash Table (DHT) is used as the underlying data structure. Since DHT is based on linked lists it reduces computational cost. Apart from computational and communication cost concerns, the existing solutions do not support all types of data outsourced to cloud. For instance, they focused more on unstructured data. The solutions for structured data also exist but there is no comprehensive framework that can support privacy preserving public auditing of structure data, unstructured data and semi-structured data. Our contributions in this paper are as follows.

1. We proposed a framework that has TPA to serve both relational and non-relational data sources for privacy preserving public auditing.
2. DHT based scheme influenced by the work in [10] is proposed and implemented for privacy preserving public auditing of unstructured data. Verifiable Auditing Scheme (VAS) is proposed in the framework but its realization is left for our future work.
3. The proposed scheme is evaluated with Amazon EC2 where services like Amazon S3 re used for storing and retrieving of unstructured data. The empirical results revealed the utility of the proposed framework.

The remainder of the paper is structured as follows. Section 2 reviews literature on public auditing schemes and privacy preserving public auditing schemes used to secure outsourced cloud storage from privacy and other attacks. Section 3 presents the proposed framework named as PASCloud which is meant for serving both relational and non-relational data that has been outsourced for privacy preserving public auditing. Section 4 presents experimental setup. Section 5 shows results of experiments made. Section 6 concludes the paper and provides directions for future work.

2. RELATED WORK

Wang et al. [2] proposed privacy preserving public auditing scheme named Oruta. They focused on identity disclosure problem with public verifiers. However, it does not prove the data freshness. Wang et al. [4] proposed a public auditing scheme known as Panda for shared data in cloud with user revocation mechanism. Wang et al. [5] explored privacy preserving public auditing with Amazon EC2 cloud platform and found it to have faster performance. Liu et al. [6] made review of public auditing schemes that operate on big data. Le et al. [7] on the other hand focused on the auditing for distributed storage systems. They proposed a scheme known as NC-Audit with the combination of SpaceMac and a homomorphic message authentication scheme. Jin et al. [8] proposed a public auditing scheme with data dynamics. They used index switcher program to address limitation of index usage and used signature exchange idea for addressing fairness problem.

Yu et al. [11] proposed a scheme for cloud storage auditing with key updates and verifiable outsourcing. Here TPA needs only encrypted version of the secret key of client. The encrypted secret key from the TPA needs to be downloaded by client for uploading new files. The client also can verify the secret key provided by TPA. Liu et al. [12] proposed an auditing scheme for regenerating-code-based cloud storage. They proposed a public verifiable authenticator based on two keys and other partial keys. Thus data owners are relieved from online burden. This scheme was provably secure and feasible. Wang et al. [14] proposed a provably secure scheme known as ID-PUIC for integrity checking of cloud storage. Wang et al. [15] proposed a public auditing scheme that supports data dynamics and security is provided using Merkle Hash Tree construction. This tree is used for block tag authentication. Jiang et al. [17] proposed a public integrity auditing scheme with user revocation in the presence of sharing data across groups. Druschel and Rowstorn [18] proposed a method known as PAST for peer-to-peer storage in large scale. A secure and distributed storage is system is modelled in [19] while secure and scalable file sharing is made in [20].
Automated availability management [21], distributed storage with decentralized erasure codes [22], automatic proxy cryptography and divertible protocols [23], Proxy re-encryption (PRE) [24] and efficient and scalable provable data possession [25] are other researches on distributed storage systems. Compact proofs of retrievability [26], privacy-preserving public auditing [27], Attribute Based Encryption (ABE) [28], secure multi-owner data sharing scheme known as Mona [29], and privacy-preserving delegated access control [30] are other methods associated with cloud storage security. The schemes close to our scheme are [3],[10], [32], [33], [34]. The novelty of the proposed framework in this paper is to have mechanisms for supporting privacy preserving public auditing for structured and unstructured data.

3. PROPOSED FRAMEWROK

A) Problem Definition

Generally outsourcing data to public cloud is done when the local resources do not support storage of huge amount of data with availability and scalability. When data is outsourced to public cloud, the local copy of data is not maintained. This is the cause of concerns from data owner point of view. Data owners do not want to outsource their data unless they have guarantee that their data is safe, secure and privacy is preserved. There are many possibilities related to data leakage or data loss or loss of privacy. 1) Cloud Service Provider (CSP) may remove some unused data to make revenues by giving that space to others. 2) There might be malicious insider attacks to cause data theft. 3) The authorized people of public cloud might perform a less than ideal operation that may cause loss of data. 4) There might be latent and inherent security issues in infrastructure provided by CSP. 5) Data is subjected to attacks by external adversaries. 6) There might be privacy attacks that cause disclosure of sensitive data. 7) In case of outsourced relational database, CSP may intentionally return empty set when user makes a request. Therefore there is necessity to ensure correctness and completeness of query results. For any reason, if data is lost or privacy is lost, or data is not returned on demand, it is very serious risk to data owner as data became valuable asset to businesses. This is the main problem to be addressed.

In the existing work such as [3], [10], [32], [33], [34] public auditing and/or privacy preserving solutions are provided. However, there is no attempt to have a comprehensive framework that support public auditing services to all kinds of data stored in public cloud. The types of data include structured, unstructured and semi-structured. We focus on the research which is aimed towards a comprehensive framework for privacy preserving public auditing strategies that cater to all types of data outsourced to cloud. It includes both relational and non-relational storage. Third Party Auditor (TPA) in the proposed framework takes care of privacy preserving public auditing of both relational and non-relational outsourced storage. This paper focuses on the realization of TPA activities for non-relational data storage that is structured and semi-structured data while the same for relational data is deferred to our future work.

B) Framework Description

We proposed a framework named PASCloud for privacy preserving public auditing. The novelty of this framework is that unlike existing frameworks that focused on one type of data such as unstructured data [10], PASCloud is meant for different kinds of data such as structured data, unstructured data and semi-structured data. Structured data is with a high degree of organization and it is stored in relational databases. Unstructured data generally has no pre-defined data model. It is stored in the form of files in storage media [31]. Semi-structured data is a kind of structured data but it does not conform to any formal data model. However, it contains markers or tags that provide some sort of structure [35]. The privacy preserving public auditing framework proposed by us is shown in Figure 1. It supports privacy preserving public auditing of structured data (relational), unstructured and semi-structured data (non-relational).

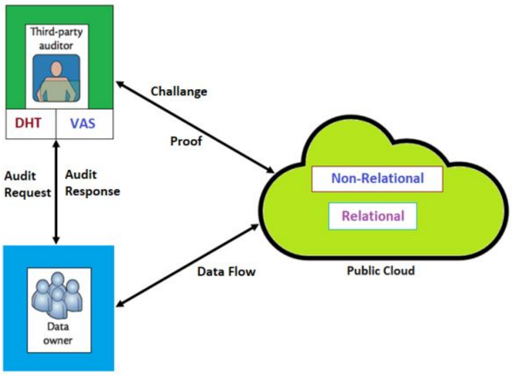

Figure 1: Overview of the proposed framework *PASCloud*

As shown in the proposed framework there are three parties involved. They are known as data owner, TPA and CSP. Data owner is an individual or organization who is willing to outsource data to public cloud. TPA is an entity which is responsible to perform auditing based on request. It is independent of data owners. It does mean that it is responsible for performing auditing tasks of all users. TPA verifies reliability of cloud storage services based on request of data owners. CSP is responsible for storage. Storage contains both relational and non-relational data [31]. Data owners to not have local copy of data as the data are very huge. This is the reason that they are concerned about security of their data. Therefore, they like to verify integrity f their data periodically by requesting TPA.

C) Design Goals of the PASCloud

The following are the design goals of the framework with respect to both relational and non-relational outsourced cloud storage.

1. TPA should perform public auditing for all users up on request. The users' data stored in public cloud needs to verified for correctness and integrity.
2. During verification process, the TPA does not need actual data blocks. It is known as blockless verification. It is required as TPA is trusted but curious to know about data. It does mean that TPA should not be able to see actual data though it is able to perform auditing.
3. Data dynamics are supported to have insertion and update of existing data besides achieving auditing tasks.
4. Privacy of data is preserved as data outsourced cannot be misused by either CSP or TPA. Especially TPA is not able to derive data from auditing information.
5. TPA is able to perform auditing of data of multiple users. It is known as batch auditing.
6. The operational of TPA such as verification of data integrity needs to be done with less computation and communication overhead.
7.

D) Dynamic Hash Table (DHT)

As shown in Figure 1, the PASCloud has provision for TPA who can provide privacy preserving auditing services to users of cloud. The novelty of the framework is that TPA supports auditing of all kinds of cloud data such as structured, unstructured and semi-structured data. Structured data is known as relational data while the other two categories are known as non-relational. For effective auditing of non-relational data Dynamic Hash Table (DHT) based public auditing is implemented. The DHT based solution is influenced by the work of [10]. The rationale behind this is that DHT reduces computational and communication overheads. It is flexible for data dynamics such as block insertion, block deletion and block modification. Since it based on linked lists, it outperforms other data structures such as MHT [32], skip list [34], DAP [33] and IHT [3]. However, the search operation involves in DHT based solution takes more time than IHT but the difference is negligible.

E) Verifiable Auditing Scheme

This scheme is employed by TPA for privacy preserving public auditing of outsourced databases or structured data. In relational models, Structured Query Language (SQL) is used to interact with the database. Especially SELECT query is used to retrieve data that satisfies query criteria. This scheme is responsible, therefore, to ensure verification of correctness and completeness of query results. Its implementation is deferred to our future work.

F) Cloud Service Provider

In our implementation Amazon AWS cloud platform is used for outsourcing relational and non-relational data. Amazon EC2 [36] is the cloud infrastructure used for experiments. This platform has Amazon Simple Storage Service (S3) [37] for holding unstructured and semi-structured data while Amazon Relational Database Service (RDS) [38] is meant for rendering scalable relational database services. The relationship among EC2, S3 and RDS is shown in Figure 2.

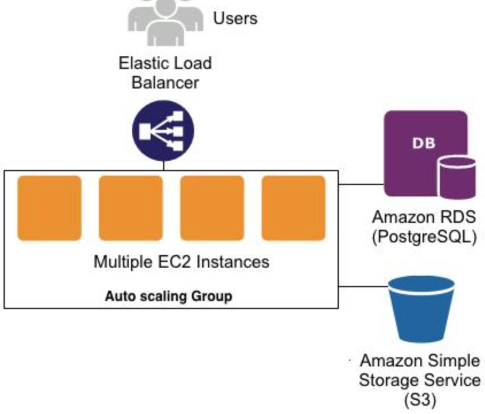

Figure 2: Amazon EC2 instances supporting RDS and S3 [39]

As presented in Figure 2, AWS cloud platform is encapsulated in the form of EC2 instances that make use of commodity computers and other computing resources. EC2 instance enables users to create both relational and non relational storage. RDS supports multiple databases such as PostgreSQL, MY SQL, and Oracle to mention few while S3 is for storing files of any kind. Thus it is suitable for the research in this paper. Both RDS and S3 are accessible from programs written in any language. SQL is used to interact with RDS while file I/O API is used to work with S3.

Table 1: Notations used in the scheme

Notation	Description
SK	Secret key
PK	Public key
A	Random number
sk	User's random secret key
G	Random element
Y	g^a
U	Random element
pk	User's random public key
ID	File identifier
SIG	Signature for given ID and private key
C	Set of all blocks
V	Version information
T	Timestamp information
F	File
Θ	File tag
Σ	Signature of a block
$\mathit{\Theta}$	Tag for a block
C	Total number of blocks
S	Set of random numbers
S	Random number
R	Random mask
chal	Challenge
IDX	Index set of the blocks to be verified
Θ	Tag proof
Λ	Data proof

Table 1 shows notations used in the proposed scheme. Before discussing setup and verification phases of the proposed scheme, the following information helps in understanding the underlying equations.

1.SK {a, sk} and PK={g, y, u, pk}
SK is set of random numbers (a) and Secret keys(sk)
PK is set of random elements (g,u) and public keys.
2. $(ID, \emptyset = \{(\vartheta_i t_i) | 1 \leq i \leq n\})$
$\emptyset$ is set of n number of versions and time stamps
3. $\vartheta = \mathrm{ID}||\mathrm{SIG}(\mathrm{sk}, \mathrm{ID})$
File tag ϑ is ExOR operation of ID and signature.
4. $\sigma = \{\sigma | 1 \leq i \leq n\}$
σ is the set of all blocks' signatures
5. $(F, \vartheta, \sigma = \{q_i | 1 \leq \mathrm{i} \leq \mathrm{n}\})$
σ is set of n number of q.
6. $Cha1 = (IDX = \{idx_i | 1 \leq i \leq c, c \leq n\}, S = \{s_i | i \in IDX\}, R)$
IDX is Set of idx(Index set of blocks) values and S is set of s(random numbers) values that belong to IDX.
Challenge is values of both IDX and S

G) Steps in Privacy Preserving Public Auditing

There is interaction among three parties such as user, TPA and CSP. The steps in the interaction for data integrity verification are as follows. The whole procedure involves two phases. They are known as setup and verification.

Setup Phase

1. User initiates a key pair known as SK {a, sk} and PK={g, y, u, pk}
2. User initiates data information $(ID, \emptyset = \{(\vartheta_i t_i) | 1 \leq i \leq n\})$
3. User sends data information to TPA
4. User generates the file tag $\vartheta = \mathrm{ID}||\mathrm{SIG}(\mathrm{sk}, \mathrm{ID})$
5. User generates signatures for each block $\sigma = \{\sigma | 1 \leq i \leq n\}$
6. User sends file and verification metadata $(F, \vartheta, \sigma = \{q_i | 1 \leq \mathrm{i} \leq \mathrm{n}\})$ to CSP.
7. CSP creates tags for each signature received.

Verification Phase

1. TPA sends file identifier (ID) to CSP.
2. CSP returns file tag.
3. TPA verifies SIG(sk, ID) using pk and quit in case of failure in verification.
4. TPA generates a challenge $Cha1 = (IDX = \{idx_i | 1 \leq i \leq c, c \leq n\}, S = \{s_i | i \in IDX\}, R$
5. TPA sends challenge to CSP.
6. CSP computes proof for data and tag.
7. TPA verifies the proof.

The verification process is between the TPA and CSP based on the auditing request made by the user. The setup process involves all three parties to be involved.

4. EXPERIMENTAL SETUP

Amazon AWS is the cloud platform used for experiments. Datasets are collected from UCI machine learning repository [40]. An account is taken from AWS for the research. An EC2 instance is used with a new cluster containing a single node. Amazon S3 is used to store and retrieve data. File I/O provided by java.io package is used to interact with S3. A prototype application is built using Java language. It is used as the interface for performing data owner and TPA activities and demonstrating proof of the concept. The application runs in the local machine and provides required interface to upload data to cloud storage, retrieve data and perform auditing operations.

5. EXPERIMENTAL RESULTS

Experiments are made to observe the performance of the proposed system in terms of two phases involved in the privacy preserving public auditing. They are known as setup and verification. The results of the proposed system are compared with that of [3], [32], [33] and [34].

Table 2: Processing time in setup phase

Block Number	Processing Time (sec)			
	IHT-PA	DAP	DHT-PA	DHT-PAScloud
10	0.2	0.2	0.2	0.2
100	3.5	3	2.5	2
200	6	5	4.5	4
300	8	7	6.5	6

As shown in Table 2, the processing time of setup phase is presented. The results revealed that as the blocks are increased, the processing time is increased. The proposed system has take less processing time when compared with other schemes.

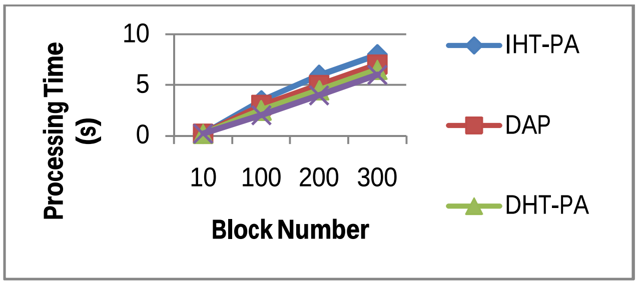

Figure 3: Processing time comparison in setup phase.

As presented in Figure 3, it is evident that the time taken for processing in setup phase is more when more number of blocks is processed. The performance of the proposed scheme DHT-PASCloud is more when compared with other schemes.

Table 3: Time taken for file appending operation .

File Size (GB)	Time Taken for File Appending (sec)			
	IHT-PA	DAP	DHT-PA	DHT-PAScloud
5	1.6	1.6	1.1	1
10	1.7	1.7	1.4	1.2
15	1.8	1.8	1.6	1.4
20	1.9	1.9	1.8	1.6

As shown in Table 3, the file size has its influence on the file appending process. The time taken for file appending is less in the proposed scheme when compared with other schemes.

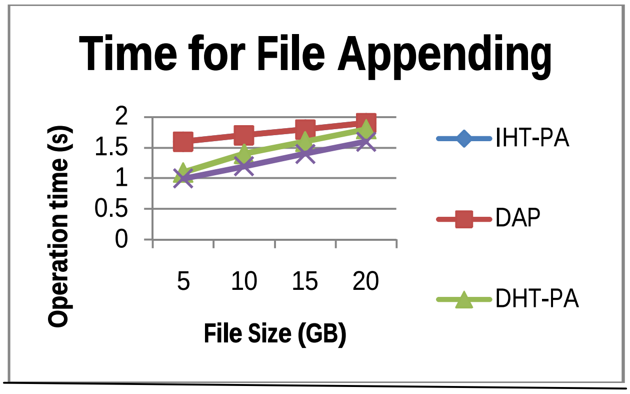

Figure 4: Comparison of file appending operation

As shown in Figure 4, it is evident that the file size has its impact on the time taken for file appending. The proposed system reflected less operation time in the verification phase.

Table 4: Time taken for file deletion.

File Size (GB)	Time Taken for File Deletion (sec)			
	IHT-PA	DAP	DHT-PA	DHT-PAScloud
5	0.7	0.7	0.25	0.22
10	0.8	0.8	0.6	0.5
15	0.9	0.9	1	0.9
20	0.95	0.95	1.5	1.3

The time taken by the schemes for file deletion is shown in Table 4. The results revealed that the time taken by the proposed system for file deletion is less than other schemes. Interestingly when the file size is 20 GB, the DAP has better performance than other schemes.

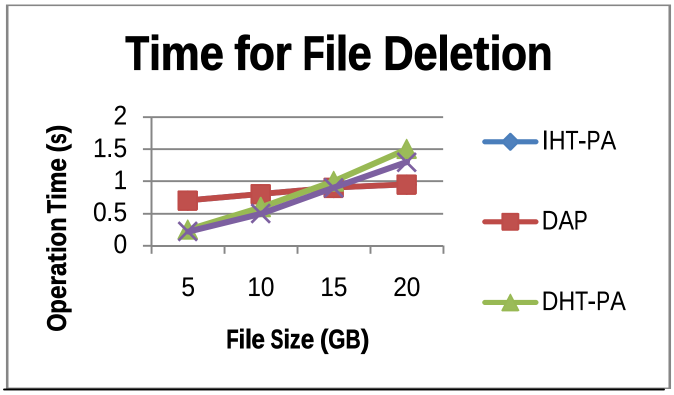

Figure 5: Comparison of file deletion performance.

As shown in Figure 5, the file size has its impact on the option time for file deletion. The result shows that the proposed scheme is better than all other schemes when file size is 5, 10 and 15 GB. When it is 20 GB, the DAP showed better performance than the proposed scheme.

Table 5: Block insertion time

File Size (GB)	Block Insertion Time (sec)			
	IHT-PA	DAP	DHT-PA	DHT-PAScloud
5	0.1	0.1	0.1	0.1
10	0.4	0.4	0.2	0.18
15	0.8	0.75	0.3	0.28
20	1.15	1.13	0.4	0.38

Table 5 shows block insertion time. The time taken for block insertion in the verification phase reveals the fact that file size has its influence on the time taken for block insertion. The propose scheme shows better performance than other schemes as the file size is increased.

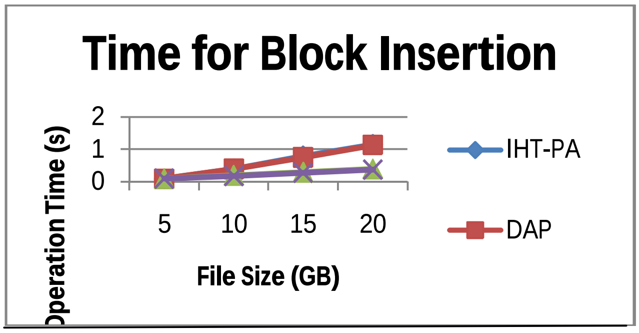

Figure 6: Block insertion time comparison

The proposed scheme showed better performance than other schemes. The size of file has its impact on the time taken for block insertion.

Table 6: Block deletion time

File Size (GB)	Block Deletion Time (sec)			
	IHT-PA	DAP	DHT-PA	DHT-PAScloud
5	0.1	0.1	0.1	0.1
10	0.4	0.4	0.2	0.18
15	0.8	0.78	0.3	0.28
20	1.2	1.2	0.4	0.38

As shown in Table 6, the block deletion time of the schemes is compared. The results revealed that the file size has its influence on the time taken for file deletion.

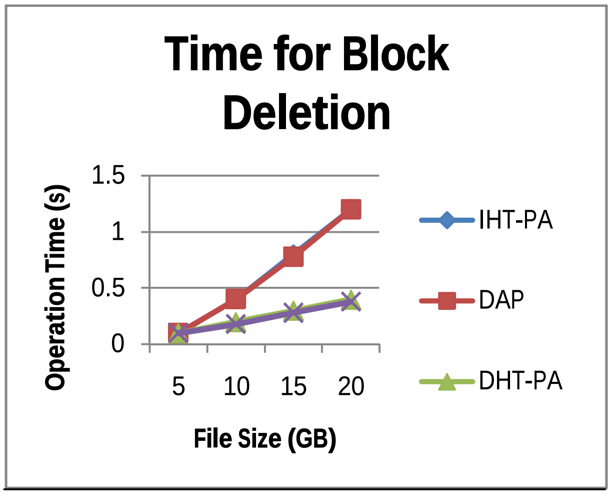

Figure 7: Comparison of block deletion time.

As presented in Figure 7, it is evident that the IHT-PA took more time than other schemes. The proposed scheme is slightly better than DHT-PA and outperforms DAP and IHT-PA schemes.

6. CONCLUSIONS AN FUTURE WORK

The public auditing schemes found in the literature focused on privacy preserving public auditing services with respect to outsourced unstructured data. There was no scheme found for all kinds of data such as structured, unstructured and semi-structured. In this paper we proposed a comprehensive framework that caters to the needs of privacy preserving public auditing of all kinds of outsourced cloud data. The framework has provision for activities of data owner, TPA and CSP. We proposed and implemented DHT based privacy preserving public auditing scheme for unstructured data. For relational data we proposed VAS but its realization is deferred to our future work. The proposed scheme is evaluated with Amazon AWS cloud platform. EC2 and S3 are used for creating cluster and storing unstructured data respectively. The proposed DHT scheme has setup and verification phases. Experiments are made to observe time taken for user processing time in setup phase and file appending, file deleting, block insertion and block deletion time in the verification phase. The empirical results are compared with state of the art schemes such as IHT-PA, DAP and DHT-PA. The results revealed that the proposed scheme has better performance in privacy preserving public auditing of unstructured data.

REFERENCES

[1] Boyang Wang, Baochun Li and Hui Li. (2010). Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud. IEEE. 10 , P1-15.

[2] Boyang Wang, Baochun Li and Hui Li. (2012). Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud. IEEE. 10 , p1-14.

[3] Yan Zhu, Gail-Joon Ahn, Hongxin Hu, Stephen S. Yau, Ho G. An, and Chang-Jun Hu. (2013). Dynamic Audit Services for Outsourced Storages in Clouds. IEEE. 6 , P227-238.

[4] Boyang Wang, Baochun Li and Hui Li. (2013). Panda: Public Auditing for Shared Data with Efficient User Revocation in the Cloud. IEEE. 10 , P1-14

[5] Cong Wang, Sherman S.M. Chow, Qian Wang, Kui Ren and Wenjing Lou. (2013). Privacy-Preserving Public Auditing for Secure Cloud Storage. IEEE. 62 , p362-375.

[6] Chang Liu, Rajiv Ranjan, Xuyun Zhang, Chi Yang, Dimitrios Georgakopoulos and Jinjun Chen. (2013). Public Auditing for Big Data Storage in Cloud Computing -- A Survey. IEEE, p1128-1135.

[7] Anh Le, Athina Markopoulou, and Alexandros G. Dimakis, . (2014). Auditing for Distributed Storage Systems. IEEE, p1-36.

[8] Hao Jin, Hong Jiang and Ke Zhou. (2014). Dynamic and Public Auditing with Fair Arbitration for Cloud Data. IEEE. 13 , p1-14.

[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, 2010.

[10] Hui Tian, Yuxiang Chen, Chin-Chen Chang,Hong Jiang, Yongfeng Huang, Yonghong Chen and Jin Liu. (2015). Dynamic-Hash-Table Based Public Auditing for Secure Cloud Storage. IEEE, p1-14.

[11] Jia Yu, Kui Ren and Cong Wang. (2015). Enabling Cloud Storage Auditing with Verifiable Outsourcing of Key Updates. IEEE, p1-13

[12] Jian Liu, Kun Huang, Hong Rong, Huimei Wang and Ming Xian. (2015). Privacy-Preserving Public Auditing for Regenerating-Code-Based Cloud Storage. IEEE, p1-13.

[13] Hui Tian, Yuxiang Chen, Chin-Chen Chang, Hong Jiang, Yongfeng Huang, Yonghong Chen and Jin Liu,. (2016). Dynamic-Hash-Table Based Public Auditing for Secure Cloud Storage. IEEE, p1-14

[14] Huaqun Wang, Debiao He and Shaohua Tang. (2016). Identity-Based Proxy-Oriented Data Uploading and Remote Data Integrity Checking in Public Cloud. IEEE, p1-11.

[15] Qian Wang, Student Member, IEEE, Cong Wang, Student Member, IEEE, Kui Ren, Member, IEEE, Wenjing Lou, Senior Member, IEEE, and Jin Li. (2011). Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing. IEEE, p1-13.

[16] Cong Wang, Sherman S.-M. Chow, Qian Wang, Kui Ren and Wenjing Lou. (2013). Privacy-Preserving Public Auditing for Secure Cloud Storage. IEEE, p1-12.

[17] Tao Jiang, Xiaofeng Chen, and Jianfeng Ma. (2015). Public Integrity Auditing for Shared Dynamic Cloud Data with Group User Revocation. IEEE, p1-12.

[18] Peter Druschel and Antony Rowstron. (2001). PAST: A large-scale, persistent peer-to-peer storage utility. IEEE, P1-6.

[19] Zooko Wilcox-O'Hearn and Brian Warner. (2008). Tahoe – The Least-Authority Filesystem. IEEEp1-6.

[20] Mahesh, Kallahalla, Erik Riedel,Ram Swaminathan,Qian Wang and Kevin Fu. (2003). Plutus: Scalable secure file sharing on untrusted storage, p1-14

[21] Ranjita Bhagwan, Kiran Tati, Yu-Chung Cheng, Stefan Savage, and Geoffrey M. Voelker. Total Recall: System Support for Automated Availability Management, p1-14.

[22] Alexandros G. Dimakis, Vinod Prabhakaran, and Kannan Ramchandran. (2006). Decentralized Erasure Codes for Distributed Networked Storage, p1-8.

[23] Matt Blaze ,Gerrit Bleumer and Martin Strauss . Divertible Protocols and Atomic Proxy Cryptography, p1-18.

[24] Giuseppe Ateniese, Karyn Benson and Susan Hohenberger. (2009). Key-Private Proxy Re-Encryption, p1-16.

[25] Giuseppe Ateniese, Roberto Di Pietro, Luigi V. Mancini and Gene Tsudik. (2008). Scalable and Efficient Provable Data Possession, p1-11.

[26] Hovav Shacham and Brent Waters. (2006). Compact Proofs of Retrievability, p1-36.

[27] Cong Wang, Qian Wang, Kui Ren, and Wenjing Lou. (2010). Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing. IEEE, p1-15.

[28] Junzuo Lai, Robert H. Deng, Chaowen Guan, and Jian Weng. (2013). Attribute-Based Encryption With Verifiable Outsourced Decryption. IEEE. 8, p1343-1354.

[29] Xuefeng Liu, Yuqing Zhang, Boyang Wang, and Jingbo Yan. (2013). Mona: Secure Multi-Owner Data Sharing for Dynamic Groups in the Cloud. IEEE. 24 , p1182-1191.

[30] Mohamed Nabeel and Elisa Bertino. (2013). Privacy Preserving Delegated Access Control in Public Clouds. IEEE, p1-14.

[31] Samundiswary.S and Nilma M Dongre . (2017). Object Storage Architecture in Cloud for Unstructured Data . IEEE, P1-6

[32]
Wang, C. Wang, K. Ren, W. Lou and J. Li. ''Enabling Public Auditability and Data Dynamics for Storage Se curity in Cloud Computing,'' IEEE Trans. on Parallel a nd Distributed Systems, vol. 22, no. 5, pp. 847-859, 20 11.

[33]
Yang and X. Jia, "An Efficient and Secure Dynamic A uditing Protocol for Data Storage in Cloud Computing, " IEEE Trans. on Parallel and Distributed Systems, vol . 24, no. 9, pp.171-1726, 201

[34] C. Erway, A. Küpçü, C. Papamanthou and R. Tamassia."Dynamic Provable Data Possession," Pro c. 16th ACM Conf. Computer and Com. Security, pp. 2 13-222, 2009.

[35] Buneman, P. Semestructured data. Available at: http://homepages.inf.ed.ac.uk/opb/papers/PODS1997a.pdf [Accessed: 02 Dec 2017]

[36] Amazon EC2 (2017). Amazon Elastic Compute Cloud. Available online at: https://aws.amazon.com/ec2/ [Accessed: 10 July 2017]

[37] Aamazon S3 (2017). Amazon Simple Storage Service. Available online at: https://aws.amazon.com/s3/ [Accessed: 10 July 2017]

[38] Amazon RDS (2017). Amazon Relational Database Service. Available online at: https://aws.amazon.com/rds/ [Accessed: 10 July 2017]

[39] REAL PYTHON. (2015). Deploying a Django App to AWS Elastic Beanstalk. Available: https://realpython.com/blog/python/deploying-a-django-app-to-aws-elastic-beanstalk/. Last accessed 10 June 2017.

[40] UCI (2017). UCI machine learning repository. Available online at: http://archive.ics.uci.edu/ml/index.php [Accessed: 20 Sep 2017]

Abstract

Dr.D.Vasumathi, currently working as Professor in department of computer science and engineering from JNTU College of engineering, Hyderabad. Her Research completed on Data Mining from JNTUH. She has vast experience in teaching. She has published several papers in international and national journals and conferences. She's the member of CSI, IEEE. Areas of Specialization are :

1. Data Mining &Data Warehousing 2. Information Retrieval System 3. Big Data Analytics 4. Cloud

Computing 5. Network Security 6. Database Management Systems.

Mr. Akheel. Mohammed. Completed M.Tech from Al - Habeeb College Of Engineering & Technology Affiliated to JNTUH in a Specialization Computer Science and Engineering. And PhD pursuing in JNTU Hyd .Presently working as Asst. Proff in SWCET, Hyderabad. Area of Specialization are Cloud computing Networks, cryptography and network security.