ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 9, September 2015

SaaS: Privacy Data Auditor in Private Cloud Computing

Gadekar Anant[1], Dr. S. Lomate[2], Prof. Saad Siddiqui[3]

M.E. CSE Student, Everest College of Engineering & Technology, Aurangabad[1]

Professor, Everest College of Engineering & Technology, Aurangabad[2]

Assistant Professor, Everest College of Engineering & Technology, Aurangabad[3]

Abstract: Cloud Computing has scalable and effective architecture for service provide such as IaaS, PaaS, SaaS. For data storage in cloud it not depends on local hardware and software environments. The Management of hardware and software are depends on remote server and enjoying the service on demand. Service security risks are demanded correctness of the data in cloud. To overcome the problem the privacy data auditor, we propose in this paper a flexible distributed storage integrity third party data auditing mechanism, utilizing the randomized homomorphism token generation and distributed encrypted data with error recovery. The proposed design allows users to audit the cloud storage with data center with low cost of computation and lightweight communication. The Proposed model allows the user to inset, delete and update its data to data center.

Keywords: Cloud Service, correctness of data, error recovery, Token generation, Data Auditor, Cloud Computing.

I. INTRODUCTION

Cloud computing is the term used to share the resources globally with less cost .we can also called as ”IT ON DEMAND‟. It provides three types of services i.e., Infrastructure as a service (IaaS), Platform as a service (PaaS) and Software as a service (SaaS). 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. End users access the cloud based applications through the web browsers with internet connection. Moving data to clouds makes more convenient and reduce to manage hardware complexities. Data stored at clouds are maintained by Cloud service providers (CSP) with various incentives for different levels of services.

End users access the cloud based applications through the web browsers with internet connection. Moving data to clouds makes more convenient and reduce to manage hardware complexities. Data stored at clouds are maintained by Cloud service providers (CSP) with various incentives for different levels of services. However it eliminates the responsibility of local machines to maintain data, there is a chance to lose data or it effects from external or internal attacks. To maintain the data integrity and data availability many people proposed several algorithms and methods that enable on demand data correctness and verification. So Cloud servers are not only used to store data like a ware house , it also provides frequent updates on data by the users with different operations like insert, delete , update and append.

We also provide third-party data auditing, where users can the integrity checking tasks to third-party auditors and use the cloud storage services. Our contribution can be summarized as the following three aspects

1) The identification of misbehaving server(s).
2) Remote data integrity, the new scheme further supports secure and efficient dynamic operations on user data block, including: update, delete and append.

II. SYSTEM MODEL

The cloud storage system architecture consists of following network entities

User: An entity, which performs data storage and retrieval operations without knowing the internal issues.

Cloud Server (CS): An entity, which provides data storage space and resources, required for computations, cloud servers are managed by cloud service providers.

Third Party Auditor (TPA): An optional Entity, but here we use TPA as Trusted party and to perform some computations instead of users.

In cloud data storage system, user can upload or stores the data into cloud or use services from the cloud (Here we focused on file storage and retrieval operations). User stores data into set of cloud servers which are running in a distributed and cooperated manner. Data redundant techniques can be employed using erasure correcting code to protect from faults or server crashes.

Users can perform manipulations on stored data like insert update and append through blocks. Block level updating and deletions are allowed with token checking. If user has not having enough resources to compute tokens or required hardware support then he can easily delegate the work to a third party auditor called as TPA. He is responsible to generate homomorphic token and stores the token persistently and securely for further verification. In our scheme we assume that TPA is secure and he is responsible to protect from threats, users will pay some incentives to TPA for maintenance.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 9, September 2015

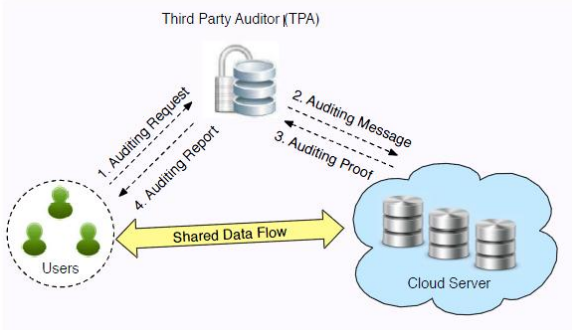

Fig. 1: Cloud data storage service auditor architecture

We assume the data integrity threats toward users' data can come from both internal and external attacks at CS. We assume the data auditor, who is auditing, is reliable and independent. However, it may harm the user if the data auditor could learn the outsourced design should achieve in some circumstances like privacy preventing, audit ability etc.

III. THE PROPOSED SCHEMES

Proposed model was introduced to explore some of threats associated in this model. As we know that the data is not present at users place because data is stored at cloud servers. It may lead to some security threats mainly two, internal attacks and external attacks. Internal attacks comes from the cloud servers itself, these servers may be malicious and lead to byzantine failures and hide some data loss issues. Secondly external attacks are from outsiders who are compromised the data from cloud service providers without its permission. Outsider attacks may lead to modification of data or deleting the users and so on which is completely masked from cloud service providers. All though TPA can also possibly hack the data for itself interested and it is also a case for inside attacks, but we ensure that TPA"s are trusted party servers. Therefore, we consider the adversary in our model to capture all types of attacks both internal and external threats. Once the server is compromised, the data is polluted with fraudulent data and users cannot get the original data from the clouds

IV. PERFORMANCE ANALYSIS & RESULT

Analysis of System:-
The proposed scheme and also such works done in the literature search are presented in table. The original PDP scheme [1] is extended by [13] in order to support data dynamics with due authentication. Thus the proposed scheme is also known as DPPP scheme. AES-based security algorithm and also BLS were implemented. The test bed used is IntelCore2 processor with 2.4 GHz HDD and 768 MB RAM. Various data integrity checking tools that monitor data remotely are gathered and tabulated. They are then compared with the proposed algorithm performance of this paper. Table 4.1 shows the comparison details.
As can be seen in table 1, the comparison results show that our scheme is supporting data dynamics and also public audit ability while other tools are supporting either data dynamic or public audit ability but not both. This shows that our proposed system is better than existing ones

Metric/Scheme	[2]	[4]	[12]*	[14]	Our Scheme
Data Dynamics	No	No	Yes	Yes	Yes
Public Auditability	Yes	Yes	No	No	Yes
Server comp. complexity	0(1)	0(1)	0(1)	0(log n)	0(log n)
Verifier comp. complexity	0(1)	0(1)	0(1)	0(log n)	0(log n)
Comm. Complexity	0(1)	0(1)	0(1)	0(log n)	0(log n)
Verifier storage complexity	0(1)	0(1)	0(1)	0(1)	0(1)

Table No 1 Shows results of various tools

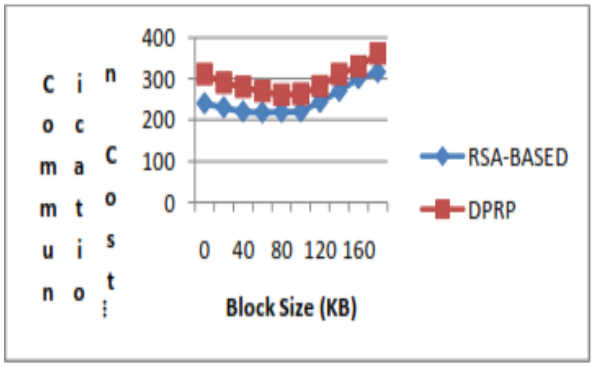

Fig 2 Communication cost over block size

As can be seen in Figure 2, it is evident that the proposed AES based scheme communication cost of DPRP scheme is more when compared with the proposed AES based scheme. AES based approach is yielding more performance.

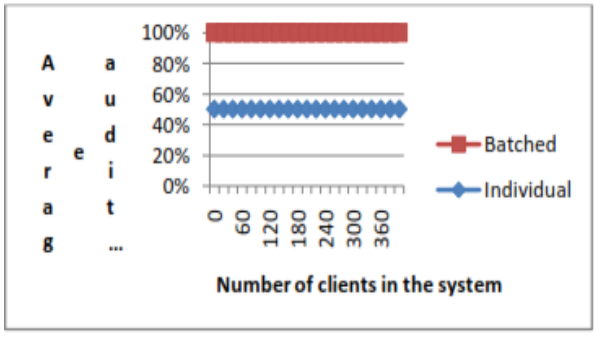

Fig 3 Average auditing time individual client

As can be seen in Figure 3, number of clients in the system and average auditing time per client are plotted in x and y axes respectively. The individual approach is showing better performance when compared with batch approach.

As can be seen in figure 4, number of clients in the system and average auditing time per client are plotted in x and y axes respectively. The individual approach is showing better performance when compared with batch approach[5].

DOI 10.17148/IJARCCE.2015.4980

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 9, September 2015

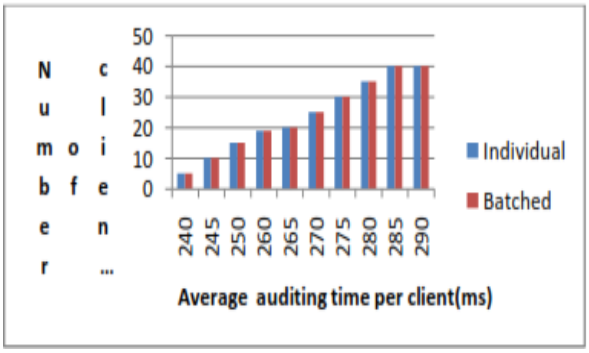

Fig 4 Average auditing time per client

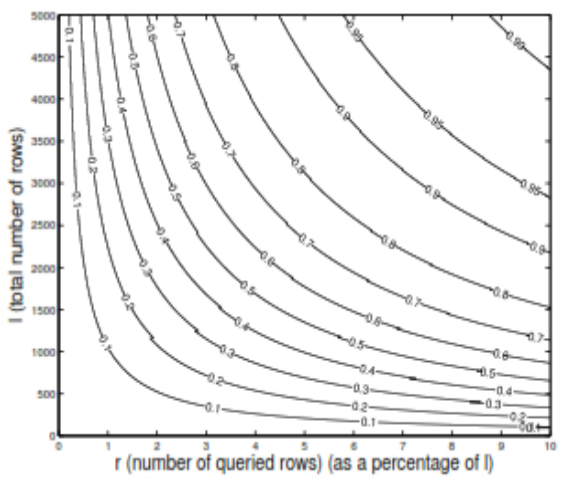

Figure 5: Detection of Probability with 1%

The detection probability Pd against data modification. Show P as a function of l (the number of blocks on each cloud storage server) and r (the number of rows queried by the user, shown as a percentage of l) for two values of z (the number of rows modified by the adversary). Both graphs are plotted under p = 16, nd= 10 and k = 5, but with different scale.

Recall that in the file distribution preparation, the redundancy parity vectors are calculated via the file matrix F by P, where P is the secret parity generation matrix we later relies on for storage correctness assurance.

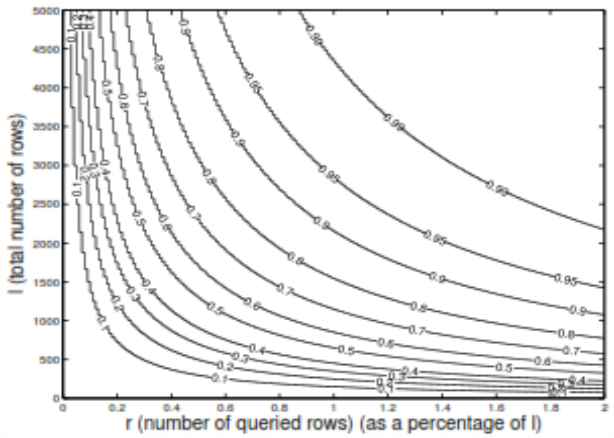

Fig 6: Detection of Probability with 10%

If disperse all the generated vectors directly after token pre-computation, i.e., without blinding, malicious servers that collaborate can reconstruct the secret P matrix easily: they can pick blocks from the same rows among the data and parity vectors to establish a set of m · k linear equations and solve for the m · k entries of the parity generation matrix P.

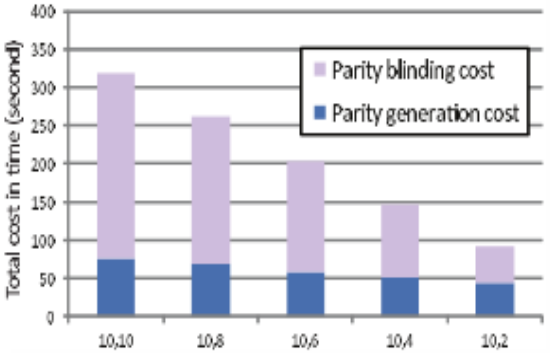

Fig 7: Blinding cost.

Performance comparison between two different parameter settings for 1 GB file distribution preparation. The (m, k) denotes the chosen parameters for the underlying Reed-Solomon coding. For example, (10, 2) means divide file into 10 data vectors and then generate 2 redundant parity vectors.

IV. CONCLUSION

In this paper, investigate the problem of data security in cloud data storage, which is essentially a distributed storage system. To achieve the assurances of cloud data integrity and availability and enforce the quality of dependable cloud storage service for users, system propose an effective and flexible distributed scheme with explicit dynamic data support, including block update, delete, and append. Rely on erasure-correcting code in the file distribution preparation to provide redundancy parity vectors and guarantee the data dependability. By utilizing the homomorphism token with distributed verification of erasure-coded data, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., whenever data corruption has been detected during the storage correctness verification across the distributed servers, can almost guarantee the simultaneous identification of the misbehaving server(s). Considering the time, computation resources, and even the related online burden of users, also provide the extension of the proposed main scheme to support third-party auditing, where users can safely delegate the integrity checking tasks to third-party auditors and be worry-free to use the cloud storage services. Through detailed security and extensive experiment results, show that our scheme is highly efficient and resilient to Byzantine failure, malicious data modification attack, and even server colluding attacks.

REFERANCES

[1]. Amazon.com, — Amazon web services (aws), ‖ Online at http ://aws.amazon.com/, 2009.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 9, September 2015

[2]. M. Arrington, ―Gmail disaster: Reports of mass email deletions,‖28/gmail-disasterreports-of-mass-email-deletions/, December 2006.
[3]. Amazon.com, ―Amazon s3 availability event: July 20, 2008,‖Onlineat http://status.aws.amazon.com/s3-20080720.html, July 2008.
[4]. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z.Peterson,and D. Song, ―Provable data possession at untrustedstores,‖in Proc. of CCS'07, Alexandria, VA, October 2007, pp. 598–609.
[5]. G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, ―Scalableand efficient provable data possession,‖ in Proc. of SecureComm'08,2008.
[6]. K.D. Bowers, A. Juels, and A. Oprea, ―Proofs of retrievability: Theory and implementation,‖ in Proc. of ACM workshop on Cloud Computing security (CCSW'09), 2009.
[7]. K. D. Bowers, A. Juels, and A. Oprea, ―Hail: A high-availability andintegrity layer for cloud storage,‖ in Proc. of CCS'09, 2009.
[8]. R. Curtmola, O. Khan, R. Burns, and G. Ateniese, ―Mr-pdp:Multiple- replica provable data possession,‖ in Proc. of ICDCS'08.IEEE Computer Society, 2008.
[9]. M. Castro and B. Liskov, ―Practical byzantine fault tolerance andproactive recovery,‖ ACM Transaction on Computer Systems, vol.20,no. 4, pp. 398–461, 2002.
[10]. Y. Dodis, S. Vadhan, and D. Wichs, ―Proofs of retrievability via hardness amplification,‖ in Proc. of the 6th Theory of Cryptography Conference (TCC'09), San Francisco, CA, USA, March 2009.
[11]. C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, ―Dynamic provable data possession,‖ in Proc. of CCS'09, 2009.
[12]. Juels and J. Burton S. Kaliski, ―Pors: Proofs of retrievability forlarge files,‖ in Proc. of CCS'07, Alexandria, VA, October 2007,
[13]. Krebs, ―Payment Processor Breach May Be Largest Ever,‖ Online at http://voices.washingtonpost.com/securityfix/2009/01/payment processor breach may b.html, Jan. 2009.
[14]. J. Kincaid, ― MediaMax/TheLinkup Closes Its Doors ,‖Online at http://www.techcrunch.com/2008/07/10/mediamaxthelinkup-closes- its-doors/, July 2008.
[15]. .M. Lillibridge, S. Elnikety, A. Birrell, M. Burrows, and M. Isard, ―A cooperative internet backup scheme,‖ in Proc. of the 2003 USENIX Annual Technical Conference (General Track), 2003, pp. 29–41.
[16]. Sun Microsystems, Inc., ―Building customer trust in cloud computing security,‖ /offers/details/sun transparency.xml, November 2009. with transparent.
[17]. M.A. Shah, M. Baker, J.C. Mogul, and R. Swaminathan, ―Auditing to keep online storage services honest,‖ in Proc. of HotOS'07. Berkeley, CA, USA: USENIX Association, 2007.
[18]. M.A. Shah, R. Swaminathan, and M. Baker, ―Privacy-preserving audit and extraction of digital contents,‖ Cryptology ePrint Archive, Report 2008/186, 2008.
[19]. H. Shacham and B. Waters, ―Compact proofs of retrievability,‖ in Proc. of Asiacrypt'08, volume 5350 of LNCS, 2008, pp. 90–107.
[20]. T. Schwarz and E.L. Miller, ―Store, forget, and check: Using algebraic signatures to check remotely administered storage,‖ in Proc. of ICDCS'06, 2006, pp. 12–12.
[21]. C. Wang, Q. Wang, K. Ren, and W.Lou, ―Ensuring data storage security in cloud computing,‖ in Proc. of IWQoS'09, July 2009.
[22]. Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, ―Enabling public verifiability and data dynamics for storage security in cloud computing,‖ in Proc. of ESORICS'09, volume 5789 of LNCS. Springer- Verlag, Sep. 2009, pp. 355–370.