IOSR Journal of Computer Engineering (IOSRJCE)
ISSN: 2278-0661, ISBN: 2278-8727 Volume 6, Issue 1 (Sep-Oct. 2012), PP 06-11
www.iosrjournals.org

Enabling Public Audit Ability and Data Dynamics for Storage Security in Cloud Computing Data Integrity Proofs In Cloud Storage

K.C Ravi Kumar[1], E.Krishnaveni Reddy [2], Priyanka Rakam[3]
[1, 2, 3]*(Cse, Sri Devi Women's Engineering College, Hyderabad, Andhra Pradesh, India)*

Abstract: *IT has moved into next generation with cloud computing being realized. The way application software and databases are stored has been changed. Now they are stored in cloud data centers in which security is a concern from client point of view. The new phenomenon which is used to store and manage data without capital investment has brought many security challenges which are not thoroughly understood. This paper focuses on the security and integrity of data stored in cloud data servers. The data integrity verification is done by using a third party auditor who is authorized to check integrity of data periodically on behalf of client. The client of the data gets notifications from third party auditor when data integrity is lost. Not only verification of data integrity, the proposed system also supports data dynamics. The work that has been done in this line lacks data dynamics and true public auditability. The auditing task monitors data modifications, insertions and deletions. The proposed system is capable of supporting both public auditability and data dynamics. The review of literature has revealed the problems with existing systems and that is the motivation behind taking up this work. Merkle Hash Tree is used to improve block level authentication. In order to handle auditing tasks simultaneously, bilinear aggregate signature is used. This enables TPA to perform auditing concurrently for multiple clients. The experiments reveal that the proposed system is very efficient and also secure.*

Index Terms: *Cloud computing, public audit ability, cloud storage, cloud service provider*

I. Introduction

Various trends are opening up the era of Cloud computing. Cloud computing is a development that is internet based and use of computer technology. The most cheaper as well as powerful processors, along with the "software as a service" (SaaS) architecture of computing , are able to transform data centers into computing service pools on a large scale. The increasing bandwidth of the network as well as reliable also flexible connections of the network makes it more possible that clients can be able to subscribe services of high-quality from data along with software that reside only on remote data centers. Even though visualized as a promising platform of service for the Internet, this new paradigm of data storage in "Cloud" possess many challenging design issues that have greater influence on the security as well as performance of the whole system. The verification of data integrity at entrusted servers is one of the main concerns in case of cloud data storage.

For solving the problem which involves data integrity verification, many schemes are proposed under various systems and security models [1], [2], [3], [4], [5], [6], [7], [8], [9], and [10]. In all these systems, great efforts are put for designing solutions that meet several requirements like high scheme efficiency, verification that is stateless, use of queries which is unbounded and data retrievability, etc. In the model, based on the verifier role, all the above schemes fall into two different categories. They are private auditability and public auditability .Though schemes that fall into private auditability category are able to achieve higher scheme efficiency, the schemes that belong to public auditability category permits anyone, not simply the client or data owner, to address the server of the cloud for accuracy of data storage while maintaining no personal information. Then, it is possible for clients are to delegate the service performance evaluation to an independent third party auditor (TPA), with no devotion of their resources of computation. The clients themselves are unreliable in the cloud or not able to address the overhead which involves carrying out integrity checks more frequently.

Another important concern among earlier designs is that of providing support to data operation dynamically for applications of cloud data storage. The proposed system deals with motivation of the public auditing system of data storage security in Cloud Computing, and develops a protocol that supports for completely dynamic data operations, particularly for supporting block insertion, which is a drawback in most of the existing schemes. Also the system is extended for supporting scalable as well as effective public auditing in Cloud Computing. To be in particular, the scheme is able to achieve batch auditing in which multiple numbers of delegated auditing tasks from various users can be carried out in a parallel manner by the TPA. The security related to proposed scheme is proved and the performance of the scheme is justified through concrete implementation as well as evaluation with the state of the art.

II. Related Work

Nowadays, a large amount of growing interest is pursued in the area of verification of remotely stored data in [1], [2], [3], [4], [5], [6], [7], [8], [9], [11], [12], [13], and [14]. Ateniese et al. [2] are the first for considering public auditability in their respective model of "provable data possession" to ensure possession of files on storages that are not trusted. In their model, they used homomorphic tags that are RSA-based for auditing data which is out sourced; hence public auditability has been achieved. Moreover, Ateniese et al. have not considered the context that involves storage of dynamic data, and extending their scheme from static data storage to dynamic storage in a direct manner can result in design as well as security problems. In [12], Wang et al. have taken into account dynamic data storage in a scenario which is distributed, and their challenge-response protocol is able to determine both the data correctness as well as location of possible errors.

Juels and Kaliski [2] proposed a model of "proof of retrievability", where spot-checking as well as error-correcting codes have been used for ensuring both "possession" as well as data files "retrievability" on archive service systems. Though the existing systems strive for providing verification of integrity for various systems of data storage, the issue that involves supporting public auditability as well as data dynamics is not fully addressed.

III. Problem Description

The proposed system architecture is as shown in fig. 1. The entities in the proposed network are client, cloud storage server and third party auditor. Client is an individual or organization who depends on cloud service provider for storing data files and maintaining them. The cloud storage server is having lot of storage space and computational resources. It is maintained by cloud service provider. Third party auditor is trusted and has capabilities of auditing the client's data on demand.

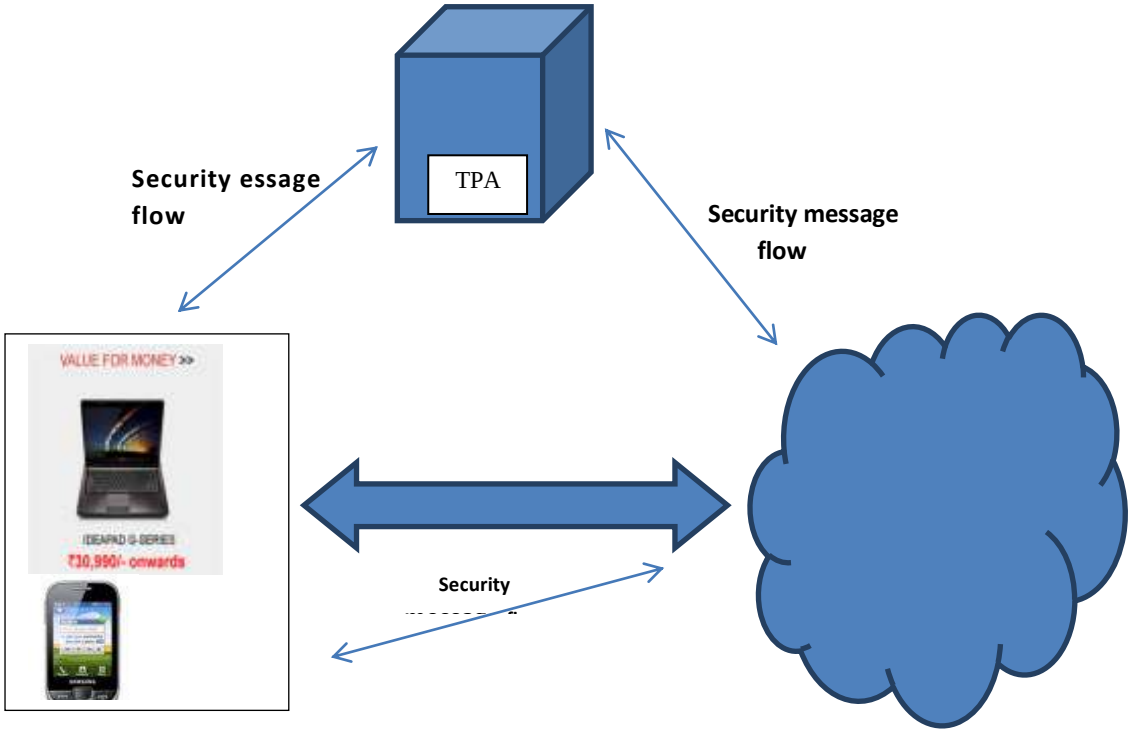

Fig. 1 shows proposed architecture

As can be seen in fig. 1, it is evident that clients store their data into cloud storage servers provided by cloud service provider. This model assumes two things. They are a) the cloud data provider may delete files of client. B) Cloud data provider may hide potential problems in the data center. Keeping these assumptions in mind, the mechanisms in the proposed system are designed.

IV. Security Mechanisms

In the proposed security mode, the TPA periodically challenges the cloud storage server in order to find whether the data stored in server is secure. This model is based on the model proposed in [3]. Other existing solutions are [1], [2], and [3]. However, the solution proposed in this paper differs from them in the verification process. Moreover those schemes do not support block insertion and dynamic data operations. The main reason is that signature construction needs information on file index. Re-computing with new indexes once a block is

inserted has to be done which causes mover overhead. To overcome this problem, the index information is removed in the computation of signatures. The tag used H(mi) for block "mi" instead of using H(name||i) as used in [3] and h(v||i) [2]. Therefore it doesn't affect other up on individual data operation on any file. In the proposed scheme the client is not given any privilege to calculate H(mi) without data. This is known as blockless verification. In order to achieve, the server takes responsibility of computing H(mi) and give it back to prover. The security risk in this approach is that the adversaries are given more opportunities to cheat the prover by generating "mi" or H(mi). The design goals of the system are storage correctness assurance through public auditability, support for dynamic data operations, and blockless verification.

V. Proposed Solution

Keeping the research goals specified in the above section, this paper introduces a new scheme for ensuring security of client data stored in cloud storage server. The protocol developed supports public auditability with dynamic data operations.

Security Analysis

The proposed system enables public auditability without need for retrieving data blocks of a file. Towards this "homomorphic authenticator technique [1] [3] is used. There is the unforgeable metadata generator computed from individual data blocks. In the proposed work two authenticators such as BLS signature [3] and RSA signature based authenticator. The security mechanism is further described here. The procedure of protocol is divided into setup, default integration verification and dynamic data operation with integrity assurance. In the last step, data modification, data insertion, and data deletion are a part. Later on batch processing with multiclient data is also discussed here.

Setup

In this phase KeyGen() method is invoked to generate public key and private key. SigGen() is meant for pre-processing and homomorphic authenticators and along with meta data. The SigGen() method takes two arguments namely secret key and file. The file content is divided into blocks. Then signature is computed for each block. Each block's hash code is taken and two nodes' hash is merged into one in order to generate the next node. This process continues for all leaf nodes until tree node is found. The root element is then taken by client and signs it and send to cloud storage server.

Data Integrity Verification

The content of outsourced data can be verified by either client or TPA. This is done by challenging server by giving some file and block randomly. Up on the challenge, the cloud storage server computes the root hash code for the given file and blocks and then returns the computed root hash code and originally stored hash code along with signature. Then the TPA or client uses public key and private key in order to decrypt the content and compare the root hash code with the root hash code returned by client. This procedure is specified in the following algorithm.

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

Table 1 shows the algorithm for data integrity verification

Data Modification and Data Insertion

Data modifications are the frequent operations on cloud storage. It is a process of replacing specified blocks with new ones. The data modification operation can't affect the logic structure of client's data. Another operation is known as data insertion. Data Insertion is a process of inserting new record in to existing data. The new blocks are inserted into specified locations or blocks in the data file F. The procedure for modification and insertion is given in Table 2.

1. Start
2. Client generates new Hash for tree then sends it to CSS
3. CSS updates F and computes new R'
4. Client computes R
5. Client verifies signature. If it fails output is FALSE
6. Compute new R and verify the update and

Table 2 shows algorithm for updating and deleting data present in CSS

Batch Auditing for Multi-client Data

Cloud servers support simultaneous access. It does mean that in server it is possible to have different verification sessions running paralelly. Therefore it is essential to have auditing functionality that works concurrently for many user sessions. The proposed scheme is extended to achieve this for provable data updates and verification of multi-client system. Here an important decision made is to make use of "Bilearaggregate Signature Scheme" [15].

` `

VI. Design Considerations

The main design consideration is to achieve auditability and data dynamics. The solution is BLS based and it can also be done with RSA based signatures. BLS solution is 160 bits where as RSA is of 1024 bits. Shortest query and response is possible with BLS. RSA also supports variable sized blocks. MHT (Merkle Hash Tree) has to be used to achieve the solution. The other design consideration is data dynamics. To achieve data dynamics PDP and PoR schemes can be extended. However, they have security problems. As discussed earlier an adversary can intrude and perform operations with ease unless, H(name||i) is changed for each update operation. Modifications are done in the existing blocks while insertion can be done at any point in F denoting a file which has been saved to cloud storage server. In basic PDP constructions the system stores static files without error correction capabilities. The proposed scheme aims at designing a blockless and stateless verification of data. This is important as the TPA does not need actual data. The actual data is not shown to anyone. Only hash values and secure keys are used for verification instead of actual data. Yet another design consideration is to support distributed storage security. When data of clients are stored in multiple cloud servers, it needs a mechanism to retrieve such data and manage data. The data is duplicated at many places to withstand faults. The given file F is stored in multiple cloud storage servers.

VII. Performance Analysis

The proposed scheme and also such works done in the literature search are presented in table 3. 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. RSA-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 3 shows the comparison details.

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 3 shows results of various tools

As can be seen in table 3, the comparison results show that our scheme is supporting data dynamics and also public auditability while other tools are supporting either data dynamic or public auditability but not both. This shows that our proposed system is better than existing ones

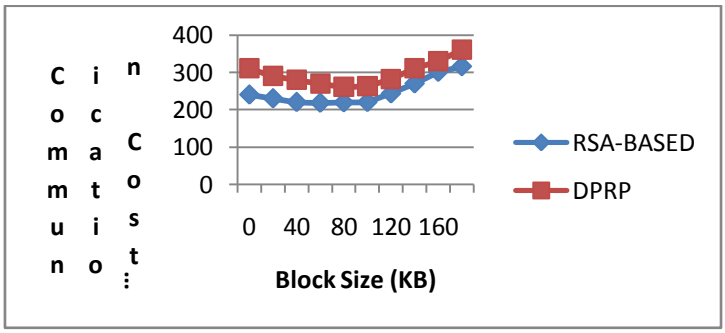

Fig. 2 shows communication cost over block size

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

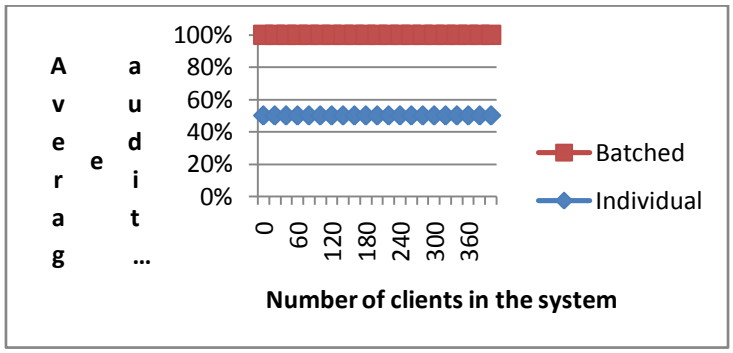

Fig. 3 shows average auditing time per client

As can be seen in fig. 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.

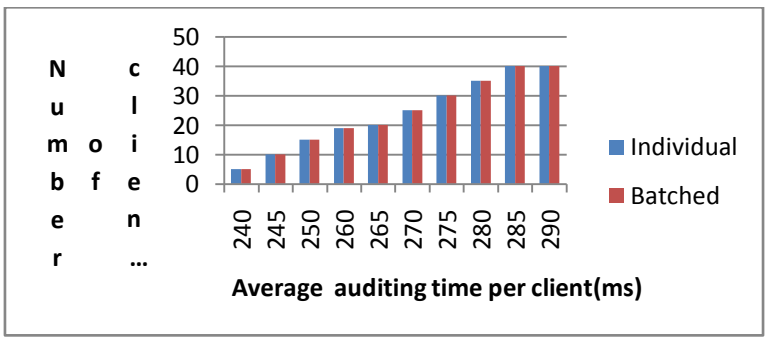

Fig. 4 shows average auditing time per client

As can be seen in fig. 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.

VIII. Conclusion

For ensuring security of cloud data storage, it is difficult for enabling a TPA for evaluating the quality of service from an objective and independent point of view. Public auditability is able to allow clients for delegating the tasks of integrity verification to TPA while they are independently not reliable or cannot commit

required resources of computation performing verifications in a continuous manner. One more important concern is the procedure for construction of verification protocols which can be able to accommodate data files that are dynamic. In this paper, the problem of employing simultaneous public auditability and data dynamics for remote data integrity check in Cloud Computing is explored.

The construction is designed for meeting these two main goals but efficiency is set as the main goal. For achieving data dynamics that are effective, the existing proof of storage models is enhanced through manipulation of the construction of classic Merkle Hash Tree for authentication of block tag. For supporting good handling of multiple numbers of auditing tasks, the method of bilinear aggregate signature is further explored for extending the main result into a multiuser setting, where TPA is able to perform multiple auditing tasks in a simultaneous manner. Huge security as well as performance analysis proves that the proposed scheme is efficient and secure to a greater extent.

References

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z.Peterson, and D. Song, "Provable Data Possession at UntrustedStores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS'07), pp. 598-609, 2007.

[2] A. Juels and B.S. Kaliski Jr., "Pors: Proofs of Retrievability forLarge Files," Proc. 14th ACM Conf. Computer and Comm. Security(CCS '07), pp. 584-597, 2007.

[3] H. Shacham and B. Waters, "Compact Proofs of Retrievability,"Proc. 14th Int'l Conf. Theory and Application of Cryptology andInformation Security: Advances in Cryptology (ASIACRYPT '08),pp. 90-107, 2008.

[4] K.D. Bowers, A. Juels, and A. Oprea, "Proofs of Retrievability:Theory and Implementation," Report 2008/175, Cryptology ePrintArchive, 2008.

[5] M. Naor and G.N. Rothblum, "The Complexity of Online MemoryChecking," Proc. 46th Ann. IEEE Symp. Foundations of ComputerScience (FOCS '05), pp. 573-584, 2005.

[6] E.-C. Chang and J. Xu, "Remote Integrity Check with DishonestStorage Server," Proc. 13th European Symp. Research in ComputerSecurity (ESORICS '08), pp. 223-237, 2008.

[7] M.A. Shah, R. Swaminathan, and M. Baker, "Privacy-PreservingAudit and Extraction of Digital Contents," Report 2008/186,Cryptology ePrint Archive, 2008.

[8] A. Oprea, M.K. Reiter, and K. Yang, "Space-Efficient Block Storage Integrity," Proc. 12th Ann. Network and Distributed System Security Symp. (NDSS '05), 2005.

[9] T. Schwarz and E.L. Miller, "Store, Forget, and Check: UsingAlgebraic Signatures to Check Remotely Administered Storage,"Proc. 26th IEEE Int'l Conf. Distributed Computing Systems (ICDCS'06), p. 12, 2006.

[10] Q. Wang, K. Ren, W. Lou, and Y. Zhang, "Dependable and Secure Sensor Data Storage with Dynamic Integrity Assurance," Proc.IEEE INFOCOM, pp. 954-962, Apr. 2009.

[11] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, "Scalableand Efficient Provable Data Possession," Proc. Fourth Int'l Conf.Security and Privacy in Comm. Networks (SecureComm '08), pp. 1-10,2008.

[12] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring Data StorageSecurity in Cloud Computing," Proc. 17th Int'l Workshop Quality of Service (IWQoS '09), 2009.

[13] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia,"Dynamic Provable Data Possession," Proc. 16th ACM Conf.Computer and Comm. Security (CCS '09), 2009.

[14] K.D. Bowers, A. Juels, and A. Oprea, "Hail: A High-Availabilityand Integrity Layer for Cloud Storage," Proc. 16th ACM Conf.Computer and Comm. Security (CCS '09), pp. 187-198, 2009.

[15] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate andVerifiably Encrypted Signatures from Bilinear Maps," Proc. 22nd Int'l Conf. Theory and Applications of Cryptographic techniques (Eurocrypt '03), pp. 416-432, 2003.

Author Details:

K.C Ravi Kumar M.Tech CSE from JNTU Hderabad currently he is the head of department for M.Tech CSE programme in SriDevi Women's Engineering College having 17 years of Academic Experience. He is life member of IEEE & IST areas of research include Data Mining & Data Warehousing Information Retrival Systems Information Security.

Mrs.E.Krishnaveni Reddy M.Tech(S.E) B.Tech(C.S.E), currently she is Assistant Professor of C.S.E Department in SriDevi Women's Engineering College.

Ms. Priyanka Rakam B.Tech (C.S.E) from Sindhura College of engineering and Technology