International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com
Volume 1, Issue 4, December 2012 ISSN 2319 - 4847

An Integrated Approach of Data storage and Security in Cloud Computing

K.SHIRISHA REDDY[1], Dr.M.BALARAJU[2]

[1]Associate Professor, CSE, VIGNAN BHARATHI INSTITUTE OF TECHNOLOGY, Hyderabad, Andhra Pradesh, India,

[2]Principal, Professor &, HOD CSE, VIDYA VIKAS INSTITUTE OF TECHNOLOGY, Hyderabad, Andhra Pradesh, India

ABSTRACT

Cloud computing is a new computing paradigm that is attracting many computer users, business, and government agencies. Cloud computing brings a lot of advantages especially in ubiquitous services where everybody can access computer services through internet. With cloud computing, there is no need of physical hardware or servers that will support the company's computer system, internet services and networks. One of the core services provided by cloud computing is data storage. In the past decades, data storage has been recognized as one of the main concerns of information technology. The benefits of network-based applications have led to the transition from server-attached storage to distributed storage. Based on the fact that data security is the foundation of information security, a great quantity of efforts has been made in the area of distributed storage security. In this article, we focus on cloud data storage security, the threats and attacks that may possibly occur in cloud computing data storage. We then propose a security mechanism to maintain the quality of service. To ensure the correctness of users' data in the cloud, we suggest an effective, flexible and dispersed design with two salient features. By utilizing the hemimorphy token with dispersed authentication of erasure coded data, our scheme achieves the incorporation of storage correctness, insurance and data error localization, i.e., the identification of misbehaving server(s).

Keywords: Cloud Computing, Secret Key Sharing

1. INTRODUCTION

Cloud computing is a new computing paradigm where in computer processing is being performed through internet by a standard browser [1]. Cloud computing builds on established trends for driving the cost out of the delivery of services while increasing the speed and agility with which services are deployed. It shortens the time from sketching out application architecture to actual deployment. Cloud computing incorporates virtualization, on-demand deployment, Internet delivery of services, and open source software [2]. The Cloud Computing Architecture of a cloud solution is the structure of the system, which comprises on-premise and cloud resources, services, middleware, and software components, geo-location, the externally visible properties of those, and the relationships between them. The term also refers to documentation of a system's cloud computing architecture. Documenting facilitates communication between stakeholders, documents early decisions about high-level design, and allows reuse of design components and patterns between projects [3]. The benefits of cloud computing are many. One is reduced cost, since you pay as you go. Other benefits are the portability of the application is that users can work from home, work, or at client locations. This increased mobility means employees can access information anywhere they are. There is also the ability of cloud computing to free-up IT workers who may have been occupied performing updates, installing patches, or providing application support. Along with the good services of Cloud Computing has to offer, there are security problems which make users anxious about the safety, reliability and efficiency of migrating to cloud computing. Big companies have second thought whether to move into the cloud because they might compromise the operation and the important information of the company. After analyzing and calculating the possible risk. Migrating into the "Cloud" will make computer processing much more convenient to the users. One of the considerations when moving to cloud is the security problem. The unique issues associated with cloud computing security have not been recognized yet. Some researchers think that cloud computing security will not be much different from existing security practices and that the security aspects can be well-managed using existing techniques such as digital signatures, encryption, firewalls, and/or the isolation of virtual environments, and so on.

The cloud makes it possible for you to access your information from anywhere at any time. While a traditional computer setup requires you to be in the same location as your data storage device, the cloud takes away that step. The cloud removes the need for you to be in the same physical location as the hardware that stores your data. Your cloud provider can both own and house the hardware and software necessary to run your home or business applications.

From the perspective of data security, which has always been an important aspect of quality of service, Cloud Computing inevitably poses new challenging security threats for number of reasons. Firstly, traditional cryptographic primitives for the purpose of data security protection can not be directly adopted due to the users' loss control of data under Cloud Computing. Therefore, verification of correct data storage in the cloud must be conducted without explicit knowledge of the whole data. Considering various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safety, the problem of verifying correctness of data storage in the cloud becomes even more challenging. Secondly, Cloud Computing is not just a third party data warehouse. The data stored in the cloud may be frequently updated by the users, including insertion, deletion, modification, appending, reordering, etc. To ensure storage correctness under dynamic data update is hence of paramount importance. However, this dynamic feature also makes traditional integrity insurance techniques futile and entails new solutions. Last but not the least, the deployment of Cloud Computing is powered by data centers running in a simultaneous, cooperated and distributed manner. Individual user's data is redundantly stored in multiple physical locations to further reduce the data integrity threats. Therefore, distributed protocols for storage correctness, assurance will be of most important in achieving a robust and secure cloud data storage system in the real world. However, such important area remains to be fully explored in the literature. Recently, the importance of ensuring the remote data integrity has been highlighted by the following research works [3]–[7]. These techniques, while can be useful to ensure the storage correctness without having users possessing data, can not address all the security threats in cloud data storage, since they are all focusing on single server scenario and most of them do not consider dynamic data operations. As a complementary approach, researchers have also proposed distributed protocols [8]–[10] for ensuring storage correctness across multiple servers or peers. Again, none of these distributed schemes is aware of dynamic data operations. As a result, their applicability in cloud data storage can be drastically limited. In this paper, we propose an effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud.

2. TYPES OF CLOUDS

There are different types of clouds that you can subscribe to depending on your needs. As a home user or small business owner, you will most likely use public cloud services.

1. Public Cloud - A public cloud can be accessed by any subscriber with an internet connection and access to the cloud space.
2. Private Cloud - A private cloud is established for a specific group or organization and limits access to just that group.
3. Community Cloud - A community cloud is shared among two or more organizations that have similar cloud requirements.
4. Hybrid Cloud - A hybrid cloud is essentially a combination of at least two clouds, where the clouds included are a mixture of public, private, or community.

2.1 Cloud Computing Models

There are three types of cloud providers that you can subscribe to: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). These three types differ in the amount of control that you have over your information, and conversely, how much you can expect your provider to do for you. Briefly, here is what you can expect from each type.

a. SaaS: To use the provider's applications running on a cloud infrastructure and accessible from various client devices through a thin client interface such as a Web browser.
b. PaaS: To deploy onto the cloud infrastructure consumer-created applications using programming language sand tools supported by the provider (java, python, .Net)
c. IaaS: To provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications.

3. PROBLEM STATEMENT

3.1. System Model

The Representative network architecture for cloud data storage is illustrated in Figure 1. Three different network entities can be identified as follows:

User: users, who have data to be stored in the cloud and rely on the cloud for data computation, consist of both individual consumers and organizations.

Cloud Service Provider (CSP): a CSP, who has significant resources and expertise in building and managing distributed cloud storage servers, owns and operates live Cloud Computing systems.

Third Party Auditor (TPA): an optional TPA, who has expertise and capabilities that users may not have, is trusted to assess and expose risk of cloud storage services on behalf of the users upon request. In cloud data storage, a user stores his data through a CSP into a set of cloud servers, which are running in a simultaneous, cooperated and distributed manner. Data redundancy can be employed with technique of erasure-correcting code to further tolerate faults or server crash as user's data grows in size and importance. Thereafter, for application purposes, the user interacts with the cloud servers via CSP to access or retrieve his data. In some cases, the user may need to perform block level operations on his data. The most general forms of these operations that we are considering are block update, delete, insert and append. As users no longer possess their data locally, it is of critical importance to assure users that their data are being correctly stored and maintained. That is, users should be equipped with security means so that they can make continuous correctness assurance of their stored data even without the existence of local copies. In case that user do not necessarily have the time, feasibility or resources to monitor their data, they can delegate the tasks to an optional trusted TPA of their respective choices. In our model, we assume that the point-to-point communication channels between each cloud server and the user is authenticated and reliable, which can be achieved in practice with little overhead. Note that we don't address the issue of data privacy in this paper, as in Cloud Computing, data privacy is orthogonal to the problem we study here.

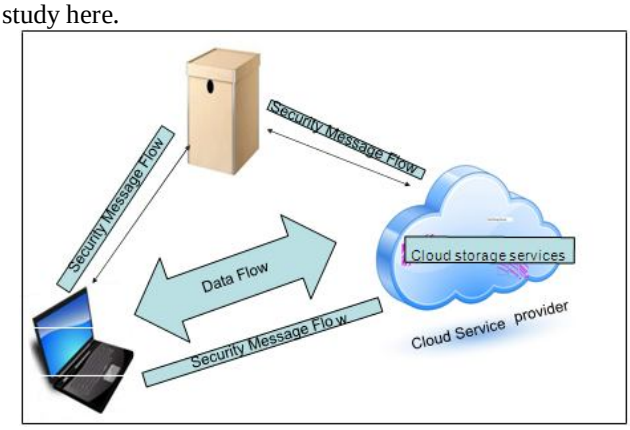

Figure 1: Cloud data storage architecture

4. CLOUD STORAGE OVERVIEW

Just like Cloud Computing, Cloud Storage has also been increasing in popularity recently due to many of the same reasons as Cloud Computing. Cloud Storage delivers virtualized storage on demand, over a network based on a request for a given quality of service (QoS). There is no need to purchase storage or in some cases even provision it before storing data. You only pay for the amount of storage your data is actually consuming.

4.1. Cloud Data Storage using Homomorphic Authenticator

In cloud data storage system, users store their data in the cloud and no longer possess the data locally. Thus, the correctness and availability of the data files being stored on the cloud servers must be assured. One of the key issues is to effectively detect any unconstitutional data variation and corruption, possibly due to server compromise and/or random intricate failures. Hence we propose this model.

The proposed model contains four modules:-

1. Key generation module: - In this module we use a key generation algorithm. This is run by the user to setup the scheme.
2. Signature generation module: - In this module we use a signature generation algorithm.

It is used by the user to generate verification metadata, which may consist of MAC, signatures or other information used for auditing.

3. Proof Generation module: - In this module we use a Proof generation algorithm. It is run by the cloud server to generate a proof of data storage correctness.
4. Proof Verification Module: - In this module we use a Proof Verification algorithm. It is run by the TPA to audit the proof from the cloud server.

The following are the steps used in the framework:-

1. Public key & secret key generation.
2. File blocks code generation.
3. Blocks Migration in to cloud.
4. Challenge message generation.
5. Cloud Service Provider authentication.
6. Verification.

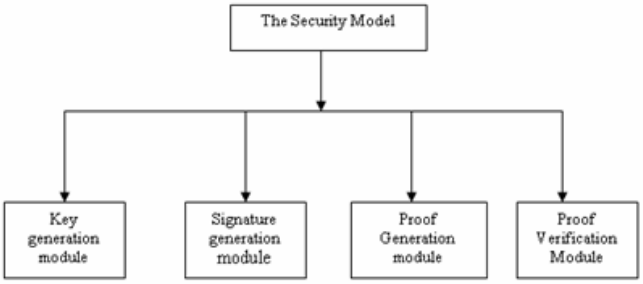

Figure 2: The Security Framework

The first three steps are termed as the setup phase & the last three are termed as the audit phase.

1. Public key & secret key generation: - The user generates public and secret parameters. Key generation algorithm is used.
2. File blocks code generation:- A code is generated by the user for each file block using homomorphic authenticator. We also use a random mask achieved by a Pseudo Random Function (PRF). A linear combination of data blocks can be verified by looking only at the aggregated authenticator.

$$\mu' = \sum_{i \in I} \nu_i m_i$$

Where vi are random number, mi are file blocks.

If TPA sees many linear combinations of the same blocks, it might be able to infer the file blocks. A random mask provided by the Pseudo Random Function (PRF) is used.

$$\mu = r + \gamma \mu'$$

Where r is the mask.

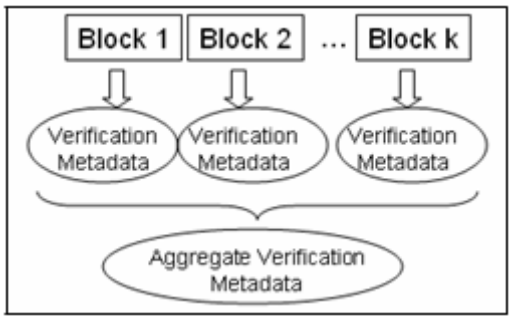

Figure 3: Homomorphic authenticator verifying file blocks

The PRF function masks the data. It has a property of not affecting the Verification Metadata. The Blocks without PRF mask & with PRF mask are verified. If both of them are equal then only they are authenticated by the aggregate authenticator. Figure 4 depicts this.

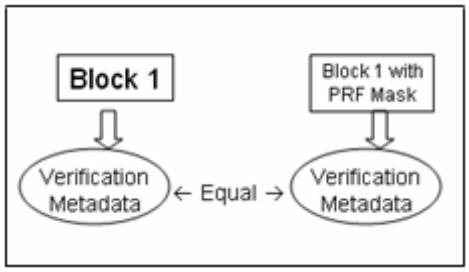

Figure 4: Random Mask by PRF

3. Blocks Migration in to cloud: - The file blocks and their codes are transmitted to the cloud.

Figure 5 depicts the first 3 steps of the process i.e. the setup phase.

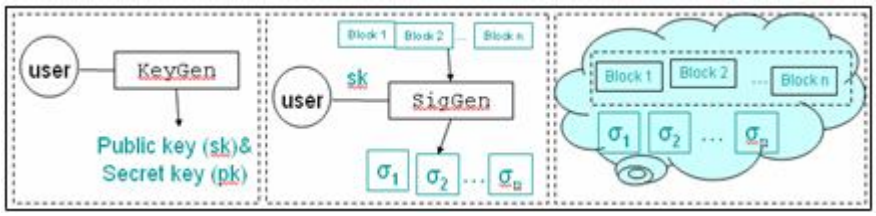

Figure 5: Setup phase

4. Challenge Message Generation: - The third party auditor sends a challenge message to the Cloud Service Provider. Basically the challenge message contains the position of the blocks that will be checked in this phase.
5. Cloud Service Provider authentication: - A proof generation algorithm is run by the Cloud server. It generates a proof of data storage correctness. The CSP picks the file blocks generated in the challenge, applies the Proof Generation algorithm and generates the proof. In addition to this the CSP also makes a linear combination of selected blocks and applies a mask. Separate PRF key is used for each audit. The CSP sends aggregate authenticator & masked combination of the blocks to TPA for further processing.
6. Verification: - In this step the proof is verified by the Verification algorithm. The Third Party Authenticator generates an aggregate authenticator. It verifies aggregate authenticator & masked combination of blocks received from the Cloud Service Provider by comparing it with the obtained Aggregate authenticator. Figure 4 depicts it.

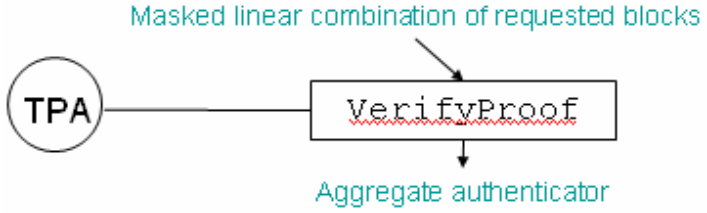

Figure 6: TPA Verification

```
Algorithm keyGen (p,s)
 {
    //user generates the public, secret keys. Input public
    parameters p;
    Input private parameters s;
    Generate Public key p(k); Generate
    Secret key s(k); End;
 }

Algorithm SigGen (m1,m2,m3…mn)
 {
  //user generates aggregated authenticator for the file blocks. Use
  homomorphic authenticator.
  Use random mask.
  Generate σ1, σ2, σ3……σn for m1,m2,m3…mn and aggregate them. Send it to
  CSP with respective file blocks.
 }

 Algorithm Genproof (TPAchallenge blocks)
 {
  //CSP generates aggregated authenticator Receive
  challenge blocks from TPA. Combine linearly the
  blocks.
 Apply mask on the received blocks
 Generate aggregate authenticator.
 Send the aggregate authenticator + masked combination of blocks to TPA for verication
 End
 }

 Algorithm Verifyproof(aggregate authenticator, masked combination of blocks)
 {
 //TPA verifies the received aggregate authenticator and generated authenticator. Receive
 CSP computed aggregate authenticator.
 If Received aggregated authenticator == generated aggregated authenticator
 Return File block secure
 Else
 File block tampered.
}
```

5. ANALYSIS & RESULTS

Table 1: Performance analysis of the proposed model.

Factors	Our Model		Public Auditing Model	
Selected Blocks	380	300	380	300
Server Computing Time(ms)	339.52	270.20	407.66	265.87
TPA Computing Time(ms)	419.47	476.81	504.25	472.55
Cost per Byte	132	40	132	40

The Table 1 shows a comparison of our model with the existing Public Auditing Model. We need 300 or 380 blocks to detect that with a probability larger than 95% or 99%, respectively if the server is missing 1% of the data. The data sent from CSP to TPA is independent of the data size and the Linear combination with mask.

6. FUTURE WORK

We can extend our model into a multi-user setting, where the TPA can perform multiple auditing tasks in a batch manner for better efficiency. Even the dynamics of the data on the cloud can be modified so as to adapt to any type of application.

7. CONCLUSION

In this paper, we investigated the problem of data security in cloud data storage, which is fundamentally a dispersed storage system. To ensure the correctness of users' data in cloud data storage, we proposed an effective model. By utilizing the Homomorphic token with Random masking, our model achieves the abstraction of the data stored on the cloud. Our process not only eliminates the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage. Through detailed security and performance analysis, we show that our scheme is highly efficient and resilient to intricate failure, malicious data modification attack, and even server colluding attacks.

REFERENCES

[1] Amazon.com, "Amazon Web Services (AWS)," Online at http://aws.mazon.com, 2008.
[2] N. Gohring, "Amazon's S3 down for several hours," Online at http://www.pcworld.com/businesscenter/ article/142549/amazons s3 down for several hours.html, 2008.
[3] A. Juels and J. Burton S. Kaliski, "PORs: Proofs of Retrievability for Large Files," Proc. of CCS '07, pp. 584–597, 2007.
[4] H. Shacham and B. Waters, "Compact Proofs of Retrievability," Proc. of Asiacrypt '08, Dec. 2008.
[5] K. D. Bowers, A. Juels, and A. Oprea, "Proofs of Retrievability: Theory and Implementation," Cryptology ePrint Archive, Report 2008/175, 2008, http://eprint.iacr.org/.
[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. Of CCS '07, pp. 598–609, 2007.
[7] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession, " Proc. of SecureComm '08, pp. 1– 10, 2008.
[8] T. S. J. Schwarz and E. L. Miller, "Store, Forget, and Check: Using Algebraic Signatures to Check Remotely Administered Storage," Proc. of ICDCS '06, pp. 12–12, 2006.
[9] M. Armbrust et al., "Above the Clouds: A Berkeley View of Cloud Computing." 2009.
[10] A. Ruiz-Alvarez and M. Humphrey, "An Automated Approach to Cloud Storage Service Selection," in 2nd Workshop on Scientific Cloud Computing (ScienceCloud 2011), 2011.
[11] M. Berkelaar, K. Eikland, and P. Notebaert, "lpsolve: Open source (mixed-integer) linear programming system," 2011. [Online]. Available: http://lpsolve.sourceforge.net/.
[12] W. W. Chu, "Optimal File Allocation in a Multiple Computer System," IEEE Transactions on Computers, vol. 18, no. 10, pp. 885- 889, Oct. 1969.
[13] R. G. Casey, "Allocation of copies of a file in an information network," in In Proceedings of the AFIPS Joint Computer Conferences, 1972, pp. 617-625.
[14] E. Grapa and G. G. Belford, "Some theorems to aid in solving the file allocation problem,"

Communications of the ACM, vol. 20, no. 11, p. 878, 1977.
[15] K. Lam and C. T. Yu, "An approximation algorithm for a fileallocation problem in a hierarchical distributed system," in In
Proceedings of the 1980 ACM SIGMOD International Conference on Management of Data, 1980, pp. 125 - 132.
[16] S. Mahmoud and J. S. Riordon, "Optimal allocation of resources in distributed information networks,"
ACM Transactions on Database Systems (TODS), vol. 1, no. 1, p. 66, 1976.
[17] L. W. Dowdy and D. V. Foster, "Comparative Models of the File Assignment Problem," ACM Computing Surveys (CSUR), vol. 14, no. 2, p. 287, 1982.

Mrs. K. SHIRISHA REDDY, B.Tech(CSIT), M.Tech(SE) ,Ph.D(CSE) pursuing from JNTUniversity Hyderabad and having a 7 years of Teaching Experience. Presently iam working as a Associate Professor, CSE, VIGNAN BHARATHI INSTITUTE OF TECHNOLOGY, Hyderabad,Andhra Pradesh, India,

Dr.M.BALARAJU, B.E(E.C.E), M.Tech(CSE), PhD(CSE) and having 19 years of Teaching Experience. He is working as a Principal , Professor &, HOD CSE, VIDYA VIKAS INSTITUTE OF TECHNOLOGY , Hyderabad, Andhra Pradesh, India