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

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 9, September 2013

Verify Correctness of Trusted data in Clouds

G. Kranthi Kumar[1], V V N V Phani Kumar[2]

Sr. Assistant Professor[1], Department of computer science and Engineering, in VR Siddhartha Engineering College.

Post Graduate Student[2], Department of Computer Science and Engineering, VR Siddhartha Engineering College.

Abstract: Cloud computing has been conceived of tangible solution to the uprising storage economy of IT enterprises. Cloud computing is the long dreamed vision of computing as a utility, where users can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. By data outsourcing, users can be relieved from the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the possibly large size of outsourced data makes the data integrity protection in Cloud Computing a very challenging and potentially formidable task, especially for users with constrained computing resources and capabilities. In this paper we propose a technique based on Metadata of the files, by using this metadata, we can verify the data that is stored by a user at remote storage in the cloud is not modified by the archive and Third party verifier. To perform cryptographic operations (encryption and decryption) required lot of computing power, but computation power is different for Personal computers, laptops and handheld devices, with this technique there may be a chance to reduce the number of cryptographic operations.

Keywords: Cloud storage, Trustness of data, Third party verifier (TPV), Cryptography.

I. INTRODUCTION

Traditional business applications and platforms of IT enterprises are too complicated and expensive. They need a data center, a complex software stack and a team of experts to run them. With the rising costs of data storage devices as well as sudden in change of rate at which data is maintained and generated, it proves for all IT enterprises as more in expensive. Moving data into the cloud offers great convenience to users since they don't have to care about the complexities of direct hardware management. The pioneer of Cloud Computing vendors, Amazon Simple Storage Service (S3) and Amazon Elastic Compute Cloud (EC2) [1] are both well-known examples. While these internet-based online services do provide huge amounts of storage space and customizable computing resources, this computing platform shift, however, is eliminating the responsibility of local machines for data maintenance at the same time. As a result, users are at the mercy of their cloud service providers for the availability and integrity of their data. Recent downtime of Amazon's S3 is such an example [2].

Apart from the cost effectiveness the management of the data and services may not be fully trustworthy. Here cloud computing poses many new security challenges that are need to be clearly understood and resolved. The traditional cryptographic primitives for the purpose of data security protection cannot 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.

II. CLOUD STORAGE

In the cloud architecture, key components are Cloud Service Provider, User (Data Owner) and Third Party Verifier.

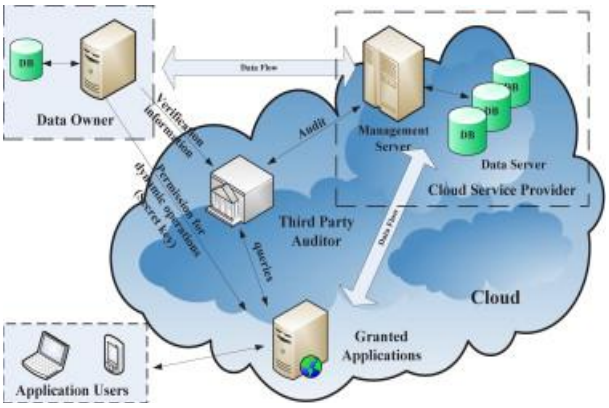

Fig 1: Cloud data storage architecture

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

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 9, September 2013

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

User: User may be a person or an organization who have data to be stored in the cloud and rely on the cloud for data computation.

Third Party Verifier (TPV): TPV has expertise and capabilities that users may not have, is trusted to assess, audit and expose risk of cloud storage services on behalf of the users upon request from the users.
This cloud storage is composed of five essential characteristics, four deployment models and three service models.

A. Essential Characteristics

i) Resource pooling: The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or data center). Examples of resources include storage, processing, memory, and network bandwidth.

ii) On-demand self-service: A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.

iii) Rapid elasticity: Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.

iv) Broad network access: Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).

v) Measured service: Cloud systems automatically control and optimize resource use by leveraging a metering capability1 at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

B. Deployment Models

i) Private cloud: The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.

ii) Public cloud: The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider.

iii) Hybrid cloud: The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds).

iv) Community cloud: The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.

C. Service Models

i) Infrastructure as a Service (IaaS): The capability provided to the consumer is 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. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).

ii) Software as a Service (SaaS): The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure2. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface.The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

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

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 9, September 2013

iii) Platform as a Service (PaaS): The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider.3 The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

III. RELATED WORK

In cloud computing, enormous threats are raised. One of the threats is data privacy and integrity. A lot of researchers focused on proving data correctness in the cloud and introduce many solutions to decrease the threat of the data integrity in the cloud and introduce many solutions to decrease the threat of the data privacy and integrity. Paul Zimski [7] says about cloud computing, putting everything into a single box will only make it easier for backers. Moving to a virtual environment to save on costs automatically introduces fresh risk on top of existing risk. PriyaMetri [3] introduces the threat model to treat the privacy problem in the clouds. One of the services is third party verification (auditing) because it notify the threats in cloud computing is tampering with data in the cloud that interfere with the unauthorized modifications for the data, which lead to an effectiveness processors, data storage and data flow.

A formal "Proof of Retrievability" (POR) model for ensuring the remote data integrity was described by A. Juels and J. Burton S. Kaliski in October 2007. Their scheme combines two methods spot-checking and error- correcting code to ensure both possession and retrievability of files on archive or backup service systems. H. Shacham and B. Waters in 2008 built on this model and constructed a random linear function based homomorphic authenticator which enables unlimited number of queries and requires less communication overhead [8]. An improved framework for POR protocols that generalizes Juels work was illustrated [4]. All these schemes are focusing on static data. The effectiveness of their schemes rests primarily on the pre-processing steps that the user conducts before outsourcing the data file F. Any change to the contents of F, even few bits, must propagate through the error-correcting code, thus introducing significant computation and communication complexity was proposed by Bowers in 2009.

A research on "Remotely stored data verification" Ateniese et al. [6] have proposed a PDP (Provable Data Possession) model for verifying if an untrusted server stores a client's data. As shown in figure 2, the data owner processes the data file to generate some metadata to store it locally. The file is then sent to be store in the server, and the owner may delete the local copy of the file. Clients may encrypt the file earlier to upload it to the storage. After that, the client asks the server to reduce a proof of the stored file, which it returned to the client. For security issues, the client verifies the "yes" or "no" answer from the server to make sure from his behaviour.

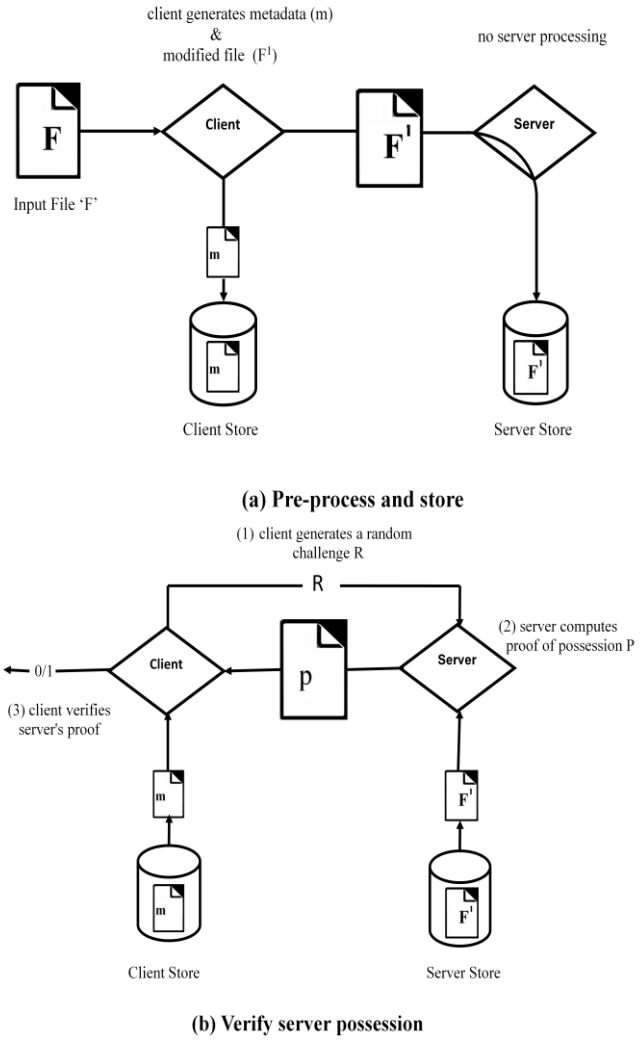

Fig 2: Protocol for Provable Data Possession

The PDP scheme works in two stages, setup stage and challenge stage. In the setup stage, the client starts by generating the public key and the secret key, and then the client computes the tags for each file block and stores it at the server. The client then sends the public key and the file to the server for storage and deletes the file from its local storage. In challenge stage, the client requests from the server a proof of possession for a subset of the blocks in the

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

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 9, September 2013

file. Then the client checks the validity of the proof. One of the adaptation tools for creating an online service is third party auditing because it allows customers to evaluate risks, and it increases the efficiency of risk reduction insurance.
C. Wang, Q. Wang, K. Ren, and W. Lou proposed an effective and flexible distributed scheme [5] with ensure the correctness of users data in the cloud. C. Wang, Q. Wang, K. Ren, and W. Lou rely on erasure correcting code in the file distribution preparation to provide redundancies and guarantee the data dependability. This construction might drastically reduce the communication and storage overhead as compared to the traditional replication-based file distribution techniques. Their scheme achieves the storage correctness insurance as well as data error localization, that is, whenever data corruption has been detected during the storage correctness verification, their scheme can almost guarantee the simultaneous localization of data errors.

File data is large and are stored at remote sites, accessing an entire file is expensive in I/O costs to the storage server and in transmitting the file across a network. Reading an entire archive, even periodically, greatly limits the scalability of network stores. Furthermore, I/O incurred to establish data possession interferes with on demand bandwidth to store and retrieve data. Previous solutions do not meet these requirements for proving data correctness in clouds. Some schemes provide a weaker guarantee by enforcing storage complexity moreover all previous techniques require the server to access the entire file, which is not feasible when dealing with large amounts of data. This paper conclude that clients need to be able to verify that a server has retained file data without retrieving the data from the server and without having the server access the entire file.

IV. PROBLEM STATEMENT

In a regular File sharing system, a user upload a file, if that file may be stored long time without accessing or may be harmful to the server that file was deleted by the administrator without intimating to the owner of the data. If user shares that link, third person want to access that file that error message is “The file you are looking may be deleted by the user or by the administrator!” Similar type of messages is show by the some of the cloud service providers. As a cloud service provider he needs to intimate to the user or owner of the data about the modifications or deletion.

In the cloud storage system, the user wants to check the data correctness, he need to access the entire file so it's expensive to the cloud server. Also transmitting the file across a network may consume high bandwidth. It is further complicated for the owner of the data whose devices like Mobile phones and tablets, because these devices can have only a limited amount of battery power, CPU power, storage capacity and communication bandwidth. Basically using cloud storage, the owner stores their files in the cloud. User can check over the data correctness by enabling a new rile which is TPV [9] because it possesses experience capabilities that the customer does not.
Third Party Verifier can understand the threats and they know best practices to identify the threats. Also they have the resources to check for process adherence and service quality. The TPV will be able to verify over any threats in online storage services that are represented in the cloud server. Thus, the user who owns the data can rely on the TPV to verify the data in the cloud without involving with the procedure. The encryption idea is based in scrambling the information that only the one who have the secret key can expose it by decryption. The encryption concept will not be enough to ensure the data integrity over the cloud. Sometimes TPV may modify file either unfortunately or fortunately.

V. VERIFY CORRECTNESS OF DATA ALGORITHM

File Upload:

STEP 1: Browse a File ‘F’
STEP 2: Generate a random number, encrypt the random number, use this encrypted random number as a Key (key1) to access the file.
STEP 3: Collect the Metadata information of File ‘F’
File Last Modification Date and time (FLM)
File Size (FS)
STEP 4: ENCRYPT (FLM + FS)
STEP 5: Encrypted Metadata is stored in the cloud for the future use.
STEP 6: key1 is sent to the owner of the file and stored in the cloud along with file ‘F’

Third Party Verifier:

TPV can verify the files in two different forms, first approach is Direct verification in this approach verifier simply check that file is exist in the user data or not. Second approach is Download and verify, in this approach TPV physically verify the file i.e., download the file see the content in the file ‘F’. To do this TPV need a key1, for that he send the key request to the owner. After taking the permission TPV download the file. Unfortunately or fortunately there is a chance of modify the data by the TPV.TPV Just before save the file into the cloud, CSP generate the metadata information and compare with the old metadata if those two metadata information’s are different, file was modified by the TPV, then CSP send the warning message to the owner and make it as a separate copy. If those two metadata information’s are same that file was not

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

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 9, September 2013

modified by the TPV, then CSP simple stores the file in the cloud.

Advantage of proposed system:

If TPV try to modify the data that information is passed to the owner of the data at low computational cost and less time. And at the same time modified copy of the file was saved in the separate storage.

V. CONCLUSIONS

In this paper we briefly explain about verification of trusted data in cloud storage. This proposed system provides the proof of data correctness and the owner can check the integrity of their data in efficient manner. If any modifications did by the TPV, cloud will immediately intimate to the owner of the file by the CSP. So security and data correctness verification is done properly. And it reduces the access time at the cloud server and reduces the cost for retrieving the file and bandwidth consumption across the network.

REFERENCES

[1] Amazon.com, "Amazon Web Services (AWS)," Online at http://aws.amazon.com, 2008.
[2] N. Gohring, "Amazon's S3 down for several hours," Onlineathttp://www.pcworld.com/businesscenter/article/142549/amazons s3down for several hours.html, 2008.
[3] Metri P. and Sarote G., "Privacy Issues and Challenges in Cloud computing," International Journal of Advanced Engineering Sciences and Technologies, vol. 5, no. 1, pp. 5-6, 2011.
[4] 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, pp. 43–54.
[5] Cong Wang, Qian Wang, and Kui Ren "Ensuring Data Storage Security in Cloud Computing" in IEEE 2009
[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson and D. Song, "Provable data possession at untrusted stores" in CCS '007.
[7] Paul Zimski, "Cloud computing faces security strom" in 2009.
[8] H. Shacham and B. Waters, "Compact proofs of retrievability," in Proc. of Asiacrypt'08, volume 5350 of LNCS, 2008, pp. 90–107.
[9] Dalia Attas and Omar Batrafi "Efficient integrity checking technique for securing client data in cloud computing" in IJECS-IJENS, 2011.