Auditing of Dynamic Big Data Storage on Cloud Using TPA

Prof. S. M. Bhadkumbhe[1], Vikas M. Mehta[2], Mohit S. Kudale[2], Sourabh D. Kurhade[2],Akash D. Pardeshi

[1] *Professor, Computer Engineering, PDEACOEM, Maharashtra, India*
[2] *Student, Computer Engineering, PDEACOEM, Maharashtra, India*

ABSTRACT

Cloud computing has includes many areas like IT companies, business line , all online shopping sites ,it also including mobile phone service providers etc. But because of such widely speeded area it has arises issues related to storage capacity and security. Cloud user can no longer physically maintain direct control over their data, which makes data security one of the major concerns of using cloud. Existing research work already allows data integrity to be varied without possession of the actual data. The trusted third party known as auditor. And verification done by this auditor is known as authorized auditing. The existing system has many drawbacks regarding third party like any one can challenge to the cloud service provider for proof of data integrity. Also in it includes re-search in BLS signature algorithm to supporting fully dynamic data updates. This algorithm is used to update an only red-sized block known as coarse-grained updates. Though this system takes more time for updating data. Here we are providing a system which support authorized auditing and ne-grained update request. Thus, our system dose not only increases security and edibility but also providing a new big data application to all cloud service providers for large data frequent small updates.

Maintaining the storages is very difficult task and it requires higher resource costs for the implementation. Here we have proposed a formal analysis technique called full grained updates. It includes the efficient searching for downloading the uploaded file and also focuses on designing the auditing protocol to improve the server-side protection for the efficient data confidentiality and data availability.

Keywords: *Big data; fine grained; full grained and auditing protocol.*

1. INTRODUCTION

In today's world of digitization, cloud computing has emerged as a concept of handling Big data. This paper focuses on the nature of Big Data, origin of Big data and security related issues with big data. Data are originated from various domains like science, education, industry, healthcare and many more. The features of data generated from different sources are different. The definition of Big data includes 5 V: Velocity, Volume, Variety, Value, Veracity. Big data is supported by new infrastructure and tools. Cloud based infrastructure, storage, network, high computing performance helps to manage the feature of Big data . New data centric security models for trusted infrastructure and data processing and storage are also proposed for the above purpose.
Big Data is not a simple Database rather it contains large scale data processing and data analytics. The most important part of Big data is its support to Dynamicity. Big data require different data centric operational models and protocols .Sometimes object or event related data go through a number of transformations and became more distributed between traditional security domains.

2. LITERATURE SURVEY

Cloud computing is on high demand today because of its features like scalability, elasticity and efficiency in supporting dynamic data. Cloud users are able to conveniently scale up/ down their virtual allocated resources according to their current need with minimal management effort and service interruption. The most existing problem in cloud is the data security and privacy. Integrity verification for outsourced data storage is the main area of today's research. Jules et al. [1] proposed a model based on POR which is only applicable to static data storage. Ateniese, et al. proposed a similar model based on PDP which verifies the integrity of a proportion of the outsourced file through verifying a combination of pre- computed file tags which they call

homomorphic verifiable tags(HVT). Sachem, et al. [2] proposed another model which is based on BLS signature scheme. BLS signature is shorter than RSA signature. In 2009, Erway, et al. proposed the first PDP scheme based on skip list that support full dynamic data updates. In any of this proposal public auditability and variable sized data block are not supported by default. Wang, et al. [3] proposed a scheme based on BLS Signature which supports the above but does not provide the facility of fine-grained data updates and authorized auditing.

3. PROPOSED SYSTEM ARCHITECTURE

It is important to assure customer about the integrity of their data in cloud. Storage correctness to ensure that there exists no cheating cloud server that can pass the TPA's audit without indeed storing users' data intact. The relationship between the cloud user and cloud service provider is transparent.

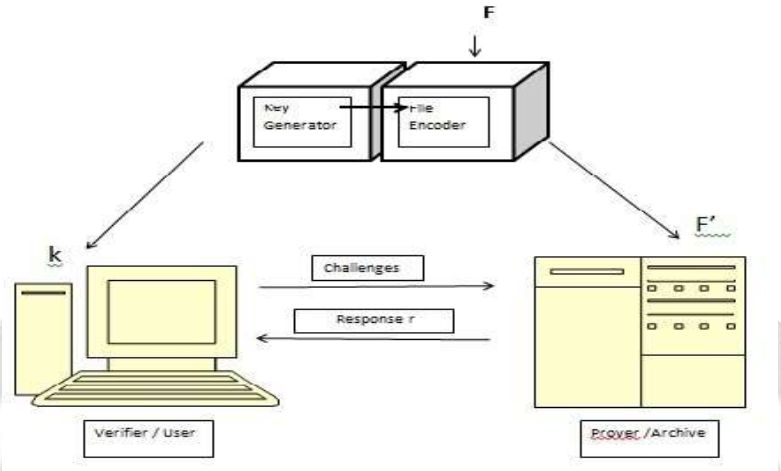

Fig-1: Architecture Diagram

Cloud user will utilize the resource of cloud on pay as you use basis. The user and service provider is not transparent in system. This agreement includes the Cloud service provider's quality of service, Standard of the service, service monitoring and controlling. TPA is there to audit the SLA and check if CSS is violating any rule to hide its fault. TPA has the list of auditing strategy and can check the integrity of data stored in cloud storage. Privacy preserving ensure that TPA cannot derive user's data content from the information collected during auditing process.

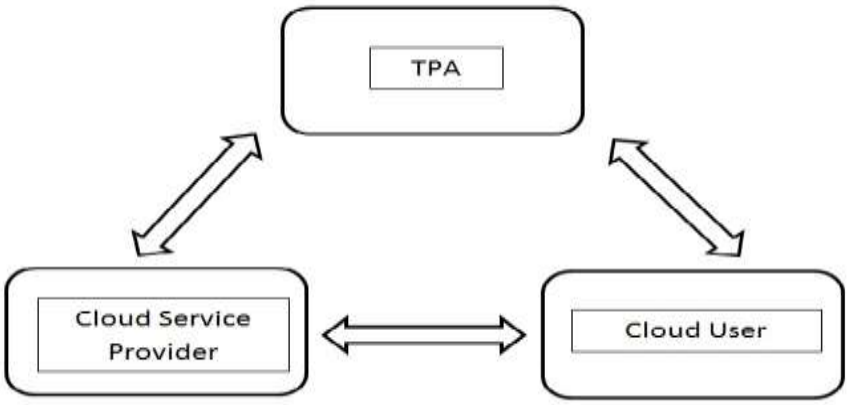

Fig-2: Flow Diagram

4. ALGORITHMS USEDsss

4.1 Message Digestion (MD5):

The MD5 algorithm is a widely used hash function producing a 128-bit hash value. Although MD5 was initially designed to be used as a cryptographic hash function, it has been found to suffer from extensive vulnerabilities. It can still be used as a checksum to verify data integrity, but only against unintentional corruption. Like most hash functions, MD5 is neither encryption nor encoding. It can be cracked by brute-force attack and suffers from extensive vulnerabilities as detailed in the

security section below. MD5 was designed by Ronald Rivest in 1991 to replace an earlier hash function MD4.[3] The source code in RFC 1321 contains a "by attribution" RSA license. The abbreviation "MD" stands for "Message Digest." The security of the MD5 has been severely compromised, with its weaknesses having been exploited in the field, most infamously by the Flame malware in 2012. The CMU Software Engineering Institute considers MD5 essentially "cryptographically broken and unsuitable for further use"

4.2 Message-Digest algorithms characteristics

Message-Digest (Fingerprint) algorithms are special functions which transform input of (usually) arbitrary length into output (so called "fingerprint" or "message digest") of constant length. These transformation functions must fulfil these requirements:

1. No one should be able to produce two different inputs for which the transformation function returns the same output

2. No one should be able to produce input for given prespecified output

Message-Digest algorithms serve in digital signature applications for guaranteeing consistency (integrity) of data. Commonly used model is as follows (message-digest in cooperation with asymmetric cryptography):

1. Sender creates input message (M) and computes its message digest (sMD). Then he uses his private key and encrypts message digest (esMD).

2. Encrypted message digest (esMD) is attached to the input message (M) and the whole message (M-esMD) is sent to receiver.

3. Receiver gets the message (M-esMD) and extracts the encrypted message digest (esMD). Then he computes his own message digest (rMD) of the received message (M). He also decodes received message digest (esMD) with sender's public key and gets decoded message digest (desMD). Then he compares both message digests (rMD ?= desMD). When both message digests are equal, the message was not modified during the data transmission.

4.3 MD5 algorithm description

MD5 algorithm takes input message of arbitrary length and generates 128-bit long output hash. MD5 hash algorithm consist of 5 steps:

Step 1. Append Padding Bits

Step 2. Append Length

Step 3. Initialize MD Buffer

Step 4. Process Message in 16-Word Blocks

Step 5. Output

4.4 Advanced Encryption Standards(AES)

128-bit encryption is a data/file encryption technique that uses a 128-bit key to encrypt and decrypt data or files. It is one of the most secure encryption methods used in most modern encryption algorithms and technologies. 128-bit encryption is considered to be logically unbreakable.

Steps:

1. Key Expansions - round keys are derived from the cipher key using Rijndael's key schedule. AES requires a separate 128-bit round key block for each round plus one more.

2. Initial Round – Add Round Key each byte of the state is combined with a block of the round key using bitwise xor.

3. Rounds – SubByte -a non-linear substitution step where each byte is replaced with another according to a lookup table.

1. Shift Rows - a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.

2. Mix Columns- a mixing operation which operates on the columns of the state, combining the four bytes in each column.

3. Add Round Key

4. Final Round (no MixColumns)

1. Sub Bytes

2. Shift Rows

3. Add Round Key

5. FUTURE SCOPE

The Future Scope of our project is the uploading of media file such as audio, video (all types of format), Rar file etc. will easily uploaded using our methodologies and techniques. Hence, in our project data security, storage and computation client security plays important role under cloud computing context.

6. CONCLUSION

Cloud computing is a big computing paradigm. Cloud data security is the important aspect for the cloud user. A trusted third party can ensure the security and integrity of data. This paper presents an overview of trusting a third party. It focuses on privacy-preserving which means TPA cannot derive user's data during the process of public data auditing. The proposed system uses a signature scheme which cannot be forged so that it will prevent malicious TPAs. It provides a feature of fine-grained dynamic data update which increases the efficiency of update process.

7. REFERENCES

[1]. Rongxing Lu ; Nanyang Technol. Univ., Singapore, Singapore ; Hui Zhu ; XimengLiu ; Liu J.K. ―Toward efficient and privacypreserving computing in big data era‖, IEEE Transactions Volume:28 , Issue: 4 ,2014.
[2]. Chang Liu, Rajiv Ranjan, Chi Yang, Xuyun Zhang, Lizhe Wang, JinjunChen―MuR-DPA: Top-down Levelled Multi-replica Merkle Hash Tree Based Secure Public Auditing for Dynamic Big Data Storage on Cloud‖ Co, IEEE Transactions on Vol:PP , Issue: 99, 2014.
[3] VenkataNarasimhaInukollu, SailajaArsi and Srinivasa Rao Ravuri ―Security Issues Associated With Big Data In Cloud Computing‖ International Journal of Network Security and Its Applications (IJNSA), Vol.6, No.3, May 2014.
[4].Garlasu,D.Sandulescu,V,― A big data implementation based on Grid computing‖ ,RoedunetInternational Conference (RoEduNet), 2011.
[5]. X. Zhang, C. Liu, S. Nepal, S. Panley, and J. Chen, __A Privacy Leakage Upper-Bound Constraint Based Approach for Cost- Effective Privacy Preserving of Intermediate Datasets in Cloud‖ IEEE Transactions. Parallel Distributed System, vol. 24, no. 6, pp. 1192-1202, June 2013.
[6]. F.C.P, Muhtaroglu, Demir S, Obali M, and Girgin C. "Business model canvas perspective on big data applications", Big Data, IEEE International Conference,2013
[7]. A, Katal, Wazid M, and Goudar R.H. "Big data: Issues, challenges, tools and Good practices." Noida: 2013, pp. 404 – 409, 8-10 Aug. 2013.
[8]. A. Cavoukian and J. Jonas, ―Privacy by Design in the Age of Big Data‖, Office of the Information and Privacy Commissioner, 2012.
[9]. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D.
Song, ―Provable data possession at untrusted stores,‖ in Proc. of CCS'07. New York, NY, USA: ACM, pp. 598–609, 2007.
[10]. R. Lu et al., ―EPPA: An Efficient and Privacy-Preserving Aggregation Sheme for Secure Smart Grid Communications‖, IEEE Trans. Parallel Distributed System, vol. 23, no. 9, 2012.
[11]. Certicom, Standards for Efficient Cryptography, SEC 1: Elliptic Curve Cryptography, Version 1.0, September 2009.