ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

Public Auditing of Dynamic Big Data Storage with Efficient High Memory Utilization and ECC Algorithm

G.Janani[1], C.Kavitha[2]

P.G Scholar, Department of CSE, Sri Shanmugha College of Engineering and Technology, Pullipalayam, Salem (Dt), India[1]

Assistant Professor, Department of CSE, Sri Shanmugha College of Engineering and Technology, Pullipalayam, Salem (Dt), India[2]

ABSTRACT: Big data is an evolving term that describes any voluminous amount of structured, semi-structured and unstructured data. The term is often used when speaking about peta-bytes and exa-bytes of data. The security and privacy is the static and huge challenging issue in big data storage. There are many ways to compromise data because of insufficient authentication, authorization, and audit (AAA) controls, such as deletion or alteration of records without a backup of the original content. The existing research work showed that it can fully support authorized auditing and fine-grained update requests. However, such schemes in existence suffer from several common drawbacks. First maintaining the storages can be a difficult task and second it requires high resource costs for the implementation. This paper, Propose 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.

I. INTRODUCTION

More organizations are running into problems with processing big data every day. The larger the data, the lengthier the processing time in most cases. Several projects have tight time.

Constraints that must be meet because of contractual agreements. When the data size increasing, can mean that the processing time will be longer than the allotted time to process the data. Since the amount of data cannot be condensed (except in rare cases), the best solution is to seek out methods to reduce the run time of programs by making them more efficient. This is also an expensive method than simply spending a lot of money to buy bigger/faster hardware, which may or may not speed up the processing time. Traditional data and new big data can be quite different in terms of content, structure, and intended use, and each category has many variations within it.

A. What is big data?

There are many different definitions of “big data” and more definitions are being created every day. At SAS Solutions on Demand (SSO) have many projects that would be considered big data projects. Some of these projects have works that run anywhere from 16 to 40 hours because of the large amount of data and complex calculations that are performed on each record or data point.

Big data is high-volume, high-velocity and high-variety information assets that demand reduced cost, original forms of information processing for enhanced insight and decision making. A massive volume of both structured and unstructured data that is so large that it's difficult to process using traditional database and software techniques. There are three generic properties of big data: Volume, Velocity and Variety. Big data can be an eclectic mix of structured data (relational data), unstructured data (human language text), semi-structured data (XML), and streaming data (machines, sensors, Web applications, and social media). The term multi-structured data refers to data sets or data environments that include a mix of these data types and structures. High velocity/speed data capture from variety of sensors and data sources and those data are delivered to different visualization and actionable systems and consumers.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

Big data requires new data-centric models such as Data location, search, access, Data integrity and identification, Data lifecycle and variability [1]. Cloud computing as a natural platform for Big Data. Big data has the target use in the areas of Scientific Discovery, New Technologies, Manufacturing Processes and Transport, Personal Services and Campaigns, Living Environment Support, Healthcare Support and Social Networking.

II. RELATED WORKS

Big Data is about the volume, variety and velocity of information being generated today and the chance that results from efficiently leveraging data for insight and good advantage. The Big Data describes a new generation of technologies and architectures designed to carefully extract value from these very large and diverse volumes of data by enabling high-velocity capture, detection, and/or analysis.

A. Need of Security in Big Data

For marketing and research, many of the businesses uses big data, but may not have the essential assets particularly from a security perspective. If a security break occurs to big data, it would result in even more severe legal consequences and reputational damage than at present.

In this new era, many companies are using the technology to store and analyze peta bytes of data about the company, business and the customers [1]. As a result, information sorting becomes even more critical. For making big data protected, techniques such as encryption, logging, and honey pot detection must be necessary.

The challenge of detecting and preventing advanced threats and malicious intruders must be solved using big data style analysis. These methods help in detecting the threats in the premature stages using more sophisticated pattern analysis and analyzing multiple data sources. There should be stability between data privacy and national security.

III. EXISTING SYSTEM

The description of the existing scheme in the aim of supporting variable-sized data blocks, authorized third party auditing and fine-grained dynamic data updates.
The scheme is described in three parts:

- Setup: the client will generate keying materials via KeyGen and FileProc, and then upload the data to CSS. Different from previous schemes, the client will store a rank based Merkle Hash Tree (MHT) as metadata. Moreover, the client will authorize the TPA by sharing a value sigAUTH.
- Verifiable DataUpdating: the CSS performs the client's fine-grained update requests via PerformUpdate, then the client runs VerifyUpdate to check whether CSS has performed the updates on both the data blocks and their corresponding authenticators (used for auditing) honestly.
- Challenge, Proof Generation and Verification: Describes how the integrity of the data stored on CSS is verified by TPA via GenChallenge, GenProof and Verify.

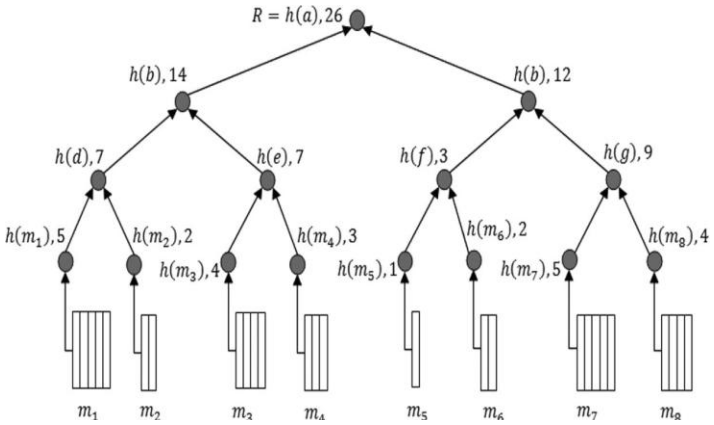

Fig.1. Merkle Hash Tree- rank based

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

In the rank based Merkle Hash Tree (Fig.1.) each node N will have a maximum of 2 child nodes. In fact, according to the update algorithm, every non-leaf node will constantly have 2 child nodes. Each child nodes have varied in their block size. The time for retrieving the data from the block may vary according to the size of the block.

If any user wants to update their file in the block, the server will return the block which is unstayed for the long time.

IV. PROPOSED SYSTEM

This paper will investigate the problem of integrity verification for big data storage in server and focus on better support for minor dynamic updates, which welfares the scalability and efficiency of a server. This scheme only focuses on big data.

To achieve this, this scheme utilizes a flexible data segmentation strategy and a data auditing protocol. Meanwhile, it address a potential security problem in supporting public verifiability [1] to make the scheme more protected and robust, which is achieved by adding an additional authorization process among the three participating parties of client, server and a Manager.

A. Architecture Diagram

The manager is responsible for the big data server to preserve the data integrity in the server. Fig.2. The manager can challenge the big data for the proof of data integrity. The big data server will produce response to the manager. The authorized user can share the data from the big data server [5] and can do data updates operation like insertion, updation and deletion. The manager will audit the server for every data update operations.

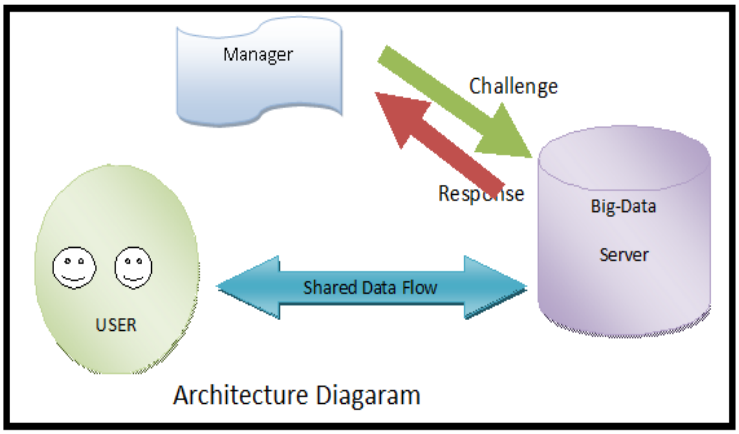

Fig.2. Architecture Diagram

Research aids of this paper can be summarized as follows:

- For the first time, this scheme formally analyzes different types of full-grained dynamic data update requests on bulk-sized file blocks in a single dataset. This scheme proposes of a public auditing scheme based on ECC signature and data auditing protocol that can support full-grained update requests. Compared to existing schemes, this scheme supports updates with a size that is not restricted by the size of file blocks; thereby enhance the flexibility and scalability.
- For better security against server side, this scheme incorporates an additional authorization process with the aim of eliminating threats of unauthorized audit challenges from malicious or pretended third-party auditors, which term as 'authorized auditing'.
- Further investigate how to improve the efficiency in verifying frequent small updates which exist in many popular cloud and big data contexts such as social media. Accordingly, this paper a further enhancement to make it more suitable for this situation than existing schemes. Compared to existing schemes, both theoretical analysis and experimental results demonstrate that this modified scheme can lower communication overheads.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

- Additionally, this scheme allows the user to efficient searching and downloading the desired file from the large data set.

B. Modules

For providing audit ability and data dynamics in the big data server the following modules are involved.

a. Network Architecture for Big Data Storage

Big data is an all-encompassing term for any collection of data sets so large and complex that it becomes difficult to process using traditional data processing applications [3].

The front end is the part seen by the client, i.e. the computer user. This comprises the client's network (or computer) and the applications used to access the big data via a user interface such as a web browser. The back end is the sql server which comprises the large data set.

b. Merkle Hash Tree for Block Tag Authentication(MHBTA)

A common form of hash trees is the Merkle hash tree, hence the name. The root hash along with the total size of the file set and the piece size are now the only information in the system that needs to come from a trustworthy source [2]. A client that has only the root hash of a file set can check any piece as follows. It heads calculates the hash of the piece it received. Merkle hash tree block Tag Authentication (MHBTA) that can support full-grained update requests.

Specifically, the server.

- Replaces the block
- Replaces outputs and
- Replaces the hash functions

The use of MHBTA, the block size of the child nodes is equally divided and the user can update their desired block which they want since all the blocks are tagged. The time for retrieving the data from the blocks is same since all the block size are same.

Full-Grained Data Update

The existing fine-grained systems consist of fewer, larger components than full-grained systems; a fine-grained description of a system regards large subcomponents while a full-grained description regards smaller components of which the larger ones are composed. The full-grained date data updates can be achieved through MHBTA.

c. Big Data Feature Extraction

Feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is very large to be processed and it is suspected to be very redundant then the input data will be transformed into a reduced representation set of features Converting the input data into the set of features is called feature extraction. If the features extracted are sensibly chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

d. Block Modification Operations

This scheme can explicitly and efficiently handle dynamic data operations for bigdata storage.

They are as follows:

Data Modification

In data modification, which is one of the most frequently used operations in big data storage. A basic data modification operation rises to the replacement of quantified blocks with new block or blocks.

Data Insertion

Related to data modification, which does not change the logic structure of client's data file, another general form of data operation is the data insertion, refers to inserting new blocks after some specified positions in the data file F.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

Data Deletion

Data deletion is just the opposite operation of data insertion. For single block deletion, it deletes the specified block and moving all the latter blocks. The details of the protocol procedures are similar to that of data modification and insertion.

e. File Search and Download

A significant amount of the world's enterprise data resides in databases. It is important that users be able to seamlessly search and browse information stored in these databases. Searching the databases on the internet and intranet today is primarily enabled by customized web applications closely tied to the schema of the underlying databases, allowing users to direct searches in a structured manner. In this going search in keyword based file search. File search is a multi-threaded documents searcher [7]. No indexes need to be restructured, no requirement of background service. The more driven the more search speed is increased thanks to its multi-threading technique.

This module allows the user to retrieve the uploaded file from big data server using search based technique.

V. ALGORITHM USED

A. Merkle Hash Tree for Block Tag Authentication (MHTBA)

A common form of hash trees is the Merkle hash tree, hence the name. The root hash along with the total size of the file set and the piece size are now the only information in the system that needs to come from a confidential source [2]. A client that has only the root hash of a file set can check any piece as follows. It calculates the hash of the piece it received.

Specifically, the server

- Replaces the block.
- Replaces outputs and
- Replaces the hash functions.

KeyGen (1k): This probabilistic algorithm is run by the client. It takes as input security parameter 1k, and returns public key pk and private key sk. (Ω, sig sk (H(R))).

SigGen(sk; F) Φ ← This algorithm is run by the client. It takes as input private key sk and a file F which is an ordered collection of blocks{ *i m* }and outputs the signature set Ω, which is an ordered collection of signatures { Ω } *on* { *m i* } *iσon m* .

It also outputs metadata—the signature sig sk (H(R)) sk of the root R of a Merkle hash tree.

B. Elliptic curve cryptography (ECC)

Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. Elliptic curves are also used in several integer factorization algorithms that have applications in cryptography, such as Lenstra elliptic curve factorization. ECC can yield a level of security with 164 bit key [11].

Elliptic curves are believed to provide good security with smaller key sizes, something that is very useful in many applications. Smaller key sizes may result in faster execution timings. It establishes low computing power and battery resource usage.

Using this algorithm the files are retrieved efficiently and securely in cryptographic manner.

VI. RESULTS AND DISCUSSION

Since, the blocks are equally divided in the full-grained updates, the time for retrieving the data from various blocks and communication overhead will be same.

In Fig.3. the use of rank based Merkel Hash Tree (Existing system in blue line) is not have the static retrieved percentage since the blocks are vary in size. In proposed system (red line) the percentage of retrieving the data is static since all the blocks sizes are equally divided.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

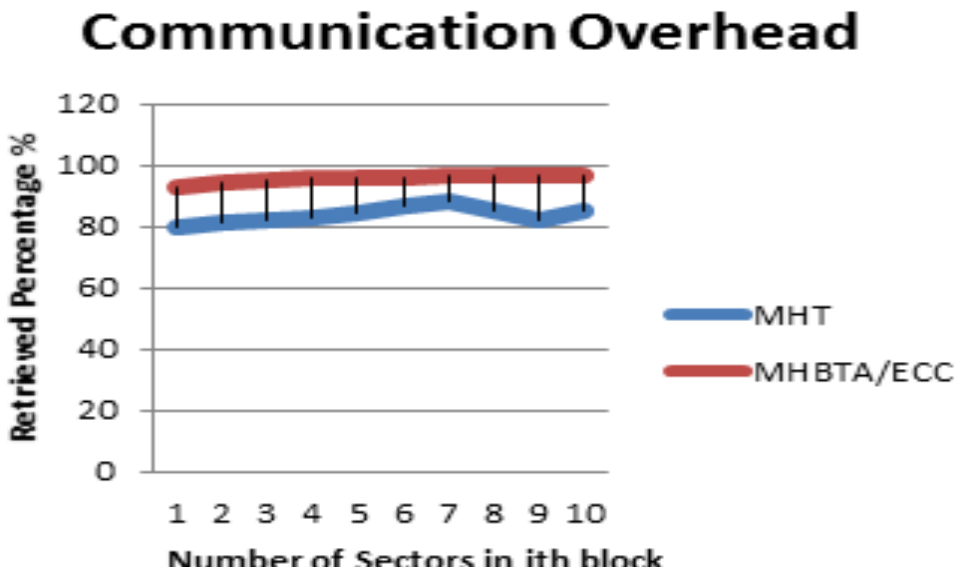

Fig.3. Performance Evaluation

VII. CONCLUSION

This paper, have provided a formal analysis on possible types of full-grained data updates and proposed a scheme that can fully support authorized auditing and full-grained update requests in big data server. Based on the full-grained, also proposed an alteration that can dramatically reduce communication overheads for verifications of small updates. Theoretic analysis and experimental results have demonstrated that this scheme can offer not only enhanced security and flexibility, but also significantly lower overheads and efficient searching and downloading the desired file from the big data server. The proposed applications support a large number of frequent small updates such as applications in social media and business transactions.

This scheme can be further enhanced by uploading very large data and can do compression on those data. Also, enhance to measure the Quality of Service (QoS).

REFERENCES

1. Rongxing Lu ; Nanyang Technol. Univ., Singapore, Singapore ; Hui Zhu ; XimengLiu ; Liu, J.K. "Toward efficient and privacy-preserving 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.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

BIOGRAPHY

1. **Anjum Asma Mohammed** is a Research Assistant in the Information Technology Department, College of Computer and Information Sciences, King Saud University. She received Master of Computer Application (MCA) degree in 2005 from BAMU, Aurangabad, MS, India. Her research interests are Computer Networks (wireless Networks), HCI, Algorithms, web 2.0 etc.

2. **Chang Liu** Department of Computer Science. His research interest in Programming Language, Security, Knowledge Representation, Semantic Web, Database, Distributed System. Working experience in Microsoft Research, Redmond, Intern, University of Maryland, Research Assistant, IBM China Research Lab, Intern Awards won: Best Paper Award, ASPLOS 2015, SoCC '14 Student Scholarship, Programming Language Mentoring Workshop Schoarship Award, 2015, 2013 The NSA Best Scientific Cyber security Paper Award, IEEE Symposium on Security and Privacy 2014 Student Travel Grants.