IJERST

International Journal of Engineering Research and Science & Technology

ISSN : 2319-5991
Vol. 1, No. 2
April 2015

2nd National Conference on "Recent Advances in Science Engineering & Technologies" RASET 2015

Organized by
Department of EEE, Jay Shriram College of Technology, Tirupur, Tamil Nadu, India.

www.ijerst.com

Email: editorijerst@gmail.com or editor@ijerst.com

ISSN 2319-5991 www.ijerst.com
Special Issue, Vol. 1, No. 2, April 2015
2nd National Conference on "Recent Advances in Science,
Engineering&Technologies" RASET-2015

Research Paper

DYNAMIC BIG DATA STORAGE USING DYNAMIC AUDITING PROTOCOL WITH MERKEL HASH TREE FOR BLOCK TAG AUTHENTICATION

M Karthick[1]*, P Lakshmanan[1], V M Naveen[1], T Sivakumar[1] and C Rajavenkateswaran[1]

Corresponding Author: **M Karthick*

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:1. Maintaining the storages can be a difficult task and 2. 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 focus on designing the auditing protocol to improve the server-side protection for the efficient data confidentiality and data availability.

INTRODUCTION

More organizations are running into problems with processing big data every day. The bigger the data, the longer the processing time in most cases. Many projects have tight time constraints that must be met because of contractual agreements. When the data size increases, it can mean that the processing time will be longer than the allotted time to process the data. Since the amount of data cannot be reduced (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 a cheaper method than simply spending a lot of money to buy bigger/ faster hardware, which may or may not speed up the processing time.

WHAT IS BIG DATA?

There are many different definitions of "big data." And more definitions are being created every day. If you ask 10 people, you will probably get 10 different definitions. At SAS®

Solutions on Demand (SSO) we have many projects that would be considered big data projects. Some of these projects have jobs that run anywhere from 16 to 40 hours because of

[1] Department of Information Technology, Nandha College of Technology, Erode, TamilNadu.

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 cost - effective, innovative 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 (from 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 datas are delivered to different visualization and actionable systems and consumers. Big Data requires new data-centric models such as Data location, search, access, Data integrity and identification, Data lifecycle and variability. Cloud Computing as a natural platform for Big Data. 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.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.

Since big data collecting is persistent, it is not efficient to move high volumes of collected data around for centralized storing. To this end, dynamic big data storage is suggested; that is, big data will be stored and organized in their original location.

It incorporates an additional authorization process with the aim of eliminating threats of unauthorized audit challenges from malicious or pretended third-party auditors, which we term as 'authorized auditing'.

File Search is a multi-threaded documents searcher. No indexes need to be updated; no background service is required. The more drives the more search speed is increased. Major problem of this system is updation in Big data server. To solve this problem our proposed system introduced an new concept called Full-grained Update in Big data Server.

Block-level operations in full-grained dynamic data updates may contain operations like partial modification; whole-block modification; whole block needs to be replaced by a new set of data; block deletion; block insertion.

The section II describes the related works about this project and section III describes proposed system and its implementation and finally section IV describes experimental result and its discussion.

RELATED WORKS

Big Data is about the volume, variety and velocity of information being generated today and the opportunity that results from efficiently leveraging data for insight and competitive advantage. The Big Data describes a new generation of technologies and architectures designed to economically extract value from these very large and diverse volumes of data by enabling high-velocity capture, discovery, and/or analysis. In big data is term important of three main characteristics. First it involves a great volume of data next one that data cannot be stored (unstructured) in regular database table, and last velocity of speed data. But in this paper we find fourth char of data ex: oracle. Because this data only process of low value density sometime we process big volume of data. Google, Yahoo and Facebook to analyse a Big amount of unstructured data. The framework for processing Big Data consists of a number of software tools that will be presented in the paper, and briefly listed here. There is Hadoop, an open source platform that consists of the Hadoop kernel, Hadoop Distributed File System (HDFS), Map Reduce and several related instruments. The main advantages offered by Grid computing are the storage capabilities and the processing power and the main advantages of using Hadoop, especially HDFS, are reliability(offered by replicating all data on multiple Data Nodes and other mechanism to protect from failure), the scheduler's ability to collocate the jobs and the data offering high throughput for data for the jobs processed on the grid.

Big Data

Big Data is the word used to describe massive volumes of structured and unstructured data that are so large that it is very difficult to process this data using traditional databases and software technologies. The term "Big Data "who had to query loosely structured very large distributed data. The three main termsthat signify Big Data have the following properties:

Volume: Many factors contribute towards increasing Volume streaming data and data collected from sensors etc.,

Variety: Today data comes in all types of formats emails, video, audio, transactions etc.,

Velocity: This means how fast the data is being produced and how fast the data needs to be processed to meet the demand.

The other two dimensions that need to consider with respect to Big Data are Variability andComplexity.

Variability: Along with the Velocity, the data flows is highly inconsistent with periodic peaks.

Complexity:Complexity of the data also needs to be considered when the data is coming from multiple sources. The data must be linked, matched, cleansed and transformed into required formats before actual processing.

Need Of Security In Big Data

For marketing and research, many of the businesses uses big data, but may not have the fundamental assets particularly from a security perspective. If a security breach occurs to big data, it would result in even more serious legal repercussions and reputational damage than at present. In this new era, many companies are using the technology to store and analyze petabytes of data about their company, business and their customers. As a result, information classification becomes even more critical. For making big data secure, techniques such as encryption, logging, honey pot detection must be necessary. In many organizations, the deployment of big data for fraud detection is very attractive and useful.

The challenge of detecting and preventing advanced threats and malicious intruders, must be solved using big data style analysis. These techniques help in detecting the threats in the early stages using more sophisticated pattern analysis and analyzing multiple data sources.

Not only security but also data privacy challenges existing industries and federal organizations. With the increase in the use of big data in business, many companies are wrestling with privacy issues. Data privacy is a liability, thus companies must be on privacy defensive. But unlike security, privacy should be considered as an asset, therefore it becomes a selling point for both customers and other stakeholders. There should be a balance between data privacy and national security.

File Encryption

Since the data is present in the machines in a cluster, a hacker can steal all the critical information. Therefore, all the data stored should be encrypted. Different encryption keys should be used on different machines and the key information should be stored centrally behind strong firewalls. This way, even if a hacker is able to get the data, he cannot extract meaningful information from it and misuse it. User data will be stored securely in an encrypted manner.

Network Encryption

All the network communication should be encrypted as per industry standards. The RPC procedure calls which take place should happen over SSL so that even if a hacker can tap into network communication packets, he cannot extract useful information or manipulate packets.

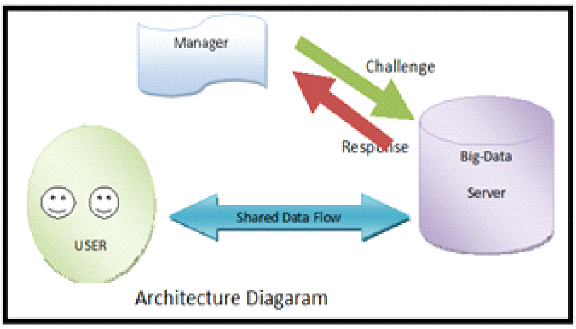
Logging

All the map reduce jobs which modify the data should be logged. Also, the information of users, which are responsible for those jobs should be logged. These logs should be audited regularly to find if any, malicious operations are performed or any malicious user is manipulating the data in the nodes.

PROPOSED SYSTEM

Architecture Diagram

Figure 1: Architecture Diagram



This paper will investigate the problem of integrity verification for big data storage in server and focus on better support for small dynamic updates, which benefits the scalability and efficiency of a server. 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 to make the scheme more secure and robust, which is achieved by adding an additional authorization process among the three participating parties of client, server and a Manager.

Research contributions of this paper can be summarized as follows:

1. For the first time, this scheme formally analyze different types of full-grained dynamic data update requests on bulk-sized file blocks in a single dataset. The propose of a public auditing scheme based on ABE 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 offers extra flexibility and scalability.

2. 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'.

3. 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 propose 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 our modified scheme can significantly lower communication overheads.

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

MODULES

Merkle Hash Tree For Block Tag Authentication

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 trusted source. A client that has only the root hash of a file set can check any piece as follows. It first calculates the hash of the piece it received. Specifically, the server.

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

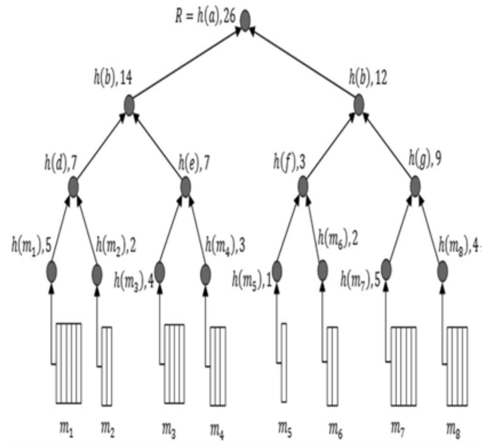
Merkle hash tree block Tag Authentication (MHBA) that can support full-grained update requests.

Full-grained Data Update

The 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.

Example For Merkle-hash Tree

Figure 2: Example of MHT



Big Data Feature Extraction

Feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be very redundant (e.g. the same measurement in both feet and meters, or the repetitiveness of images presented as pixels or length or bytes), then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called feature extraction. If the features extracted are carefully 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.

Block Modification Operations

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

Data Modification

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

Data Insertion

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

Data Deletion

Data deletion is just the opposite operation of data insertion. For single block deletion, it refers to deleting the specified block and moving all the latter blocks one block forward. The details of the protocol procedures are similar to that of data modification and insertion, which are thus omitted here.

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 as well. Searching 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 we going search in keyword based file search.File Search is a multi-threaded documents searcher. No indexes need to be updated; no background service is required. The more you have drives the more search speed is increased thanks to its multi-threading technique.

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

Merkle Hash Tree For Block Tag Authentication

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 trusted source. A client that has only the root hash of a file set can check any piece as follows. It first 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* } *i σ on m .*

It also outputs metadata—the signature sig sk (H(R)) sk of the root R of a Merkle hash tree. In our construction, the leaf nodes of the Merkle hash tree are hashes of *H (mi* **)**straightforwardly as the verification covers all the data blocks. However, the number of verifications allowed to be performed in this solution is limited by the number of secret keys. Once the keys are exhausted, the data owner has to retrieve the entire file of F from the server in order to compute new MACs(Message Authentication Code), which is usually impractical due to the huge communication overhead. Moreover, public auditability is not supported as the private keys are required for verification. Another basic solution is to use signatures instead of MACs to obtain public auditability. The data owner precomputes the signature of each block *mi* (*i* [1,*n*]) and sends both F and the signatures to the server for storage.

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 a modification that can dramatically reduce communication overheads for verifications of small updates. Theoretical 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 large data and can do compression on those data. Also, enhance to measure the Quality of Service(QoS).

REFERENCES

1. Griffith R, A D Joseph, R.Katz, A.Konwinski,G. Lee, D. Patterson, A. Rabkin, I. Stoica, andM. Zaharia (2010), "AViewof Cloud Computing," Commun. ACM, Vol. 53, No. 4, pp. 50-58.

2. Wang C, K Ren, W Lou, and J Li (2011), ''Enabling PublicAuditability and Data Dynamics for Storage Security in CloudComputing,'' IEEE Trans. Parallel Distrib. Syst., Vol. 22, No. 5, pp. 847-859.

3. Hu H, G-J Ahn, and M Yu (2012), ''Cooperative ProvableData Possession for Integrity Verification in Multi-Cloud Storage,''IEEE Trans. Parallel Distrib. Syst., Vol. 23, No. 12, pp. 2231-2244.

4. Chen S, J Yao, and D Thilakanathan (2011), ''DIaaS: Dataintegrity as a Service in the Cloud,'' in Proc. 4th Int'l Conf. onCloud Computing (IEEE CLOUD), pp. 308-315.

5. Schmidt S E (2013), ''Security and Privacy in the AWS Cloud,''presented at the Presentation Amazon Summit Australia,Sydney,Australia,May 2012, accessed on: March 25, 2013. [Online]. Available: http://aws.amazon.com/apac/aws summit-au/.

6. Liu C, X Zhang, C Yang, and J Chen (2013), ''CCBKEVSession KeyNegotiation for Fast and Secure Scheduling of ScientificApplications in Cloud Computing,'' Future Gen. Comput. Syst., Vol. 29, No. 5, pp. 1300-1308.

7. Zhang X, C Liu, S Nepal, S Panley, and J Chen (2013), ''A PrivacyLeakage Upper-Bound Constraint Based Approach for Cost-

Effective Privacy Preserving of Intermediate Datasets in Cloud,'' *IEEE Trans. Parallel Distrib. Syst.,* Vol. 24, No. 6, pp. 1192-1202.

8. Zissis D and D Lekkas (2011), "Addressing Cloud Computing SecurityIssues," Future Gen. Comput. Syst., vol. 28, no. 3, pp. 583-592.

9. Wang C, Q Wang, K Ren, and W Lou (2010), ''Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing, ''in Proc. 30st IEEE Conf. on Comput.andCommun. (INFOCOM), pp. 1-9.

International Journal of Engineering Research and Science & Technology
Hyderabad, INDIA. Ph: +91-09441351700, 09059645577
E-mail: editorijlerst@gmail.com or editor@ijerst.com
Website: www.ijerst.com

ISSN 2319-5991