A Parallel Graph Mining Library for Joint Cloud Computing

G.Balamurugan[1], N.Balasubramanian[2], V.Santhosh Kumar[3]
[1,2,3]*Department of MCA, Mohamed Sathak, Engineering College, Kilakarai, India*

Abstract-- Recent years witness the development of cloud computing technology. With the explosive growth of unstructured data, cloud storage technology gets more attention and better development. However, in current storage schema, user's data is totally stored in cloud servers. In other words, users lose their right of control on data and face privacy leakage risk. Traditional privacy protection schemes are usually based on encryption technology, but these kinds of methods cannot effectively resist attack from the inside of cloud server. In order to solve this problem, we propose a three-layer storage framework based on fog computing. The proposed framework can both take full advantage of cloud storage and protect the privacy of data. Besides, Hash-Solomon code algorithm is designed to divide data into different parts. Then, we can put a small part of data in local machine and fog server in order to protect the privacy. Moreover, based on computational intelligence, this algorithm can compute the distribution proportion stored in cloud, fog, and local machine, respectively. Through the theoretical safety analysis and experimental evaluation, the feasibility of our scheme has been validated, which is really a powerful supplement to existing cloud storage scheme.

Keywords: Cloud security system, Fog Computing, Hash-Solomn code.

1. INTRODUCTION

With the rapid development of graph data, single-thread algorithms are getting overwhelmed to process TB or even PB-scale data in reasonable time. To overcome this challenge, cloud storage and cloud computation platforms with distributed system arise. While more and more applications are deployed on various cloud platforms, there are many real world scenarios that require several applications cooperating, such as complexity network analysis. In complexity network analysis, graph computation is a vital topic, including graph structure analysis, community detection, ranking, and centralities. Most of these applications consume enormous computation resources when processing large scale data, running far out of a single machine's capacity. Although some distributed graph processing frameworks provide several graph algorithms to deal with complexity network analysis, it takes great learning cost for cross-platform migration due to highly platform- coupled implementations. In addition, the parallel graph algorithms are needed for distributed graph processing frameworks to improve their performance. As such, achieving efficient graph computation in joint cloud environment is challenging.

In the rapidly evolving landscape of data storage and processing, the integration of cloud computing with emerging paradigms like fog computing has become instrumental in addressing the growing demands for efficiency, scalability, and responsiveness. Three-layer cloud storage with fog computing represents a sophisticated architecture that combines the advantages of traditional cloud storage with the decentralized processing capabilities of fog computing, offering a comprehensive solution for modern data management challenges.

Benefits and Applications:

The three-layer cloud storage with fog computing architecture offers several benefits and finds applications across various domains:

Enhanced Performance: By leveraging fog computing's proximity to data sources, the architecture reduces latency and improves the responsiveness of data-intensive applications, such as IoT, real-time analytics, and edge AI.

Scalability and Flexibility: The distributed nature of the architecture enables seamless scalability, allowing organizations to adapt to changing data storage and processing requirements dynamically.

Resilience and Reliability: With redundancy built into both cloud storage and fog computing layers, the architecture enhances data resilience and ensures high availability, even in the face of network disruptions or

hardware failures.

Privacy and Security: By processing sensitive data at the edge and employing encryption and access control mechanisms, the architecture enhances privacy and security, mitigating risks associated with centralized data processing. The scope of this project encompasses the design, development, and deployment of a three-layer fog computing architecture within J2EE applications.

2. RELATED WORKS

Recent years witness the development of cloud computing technology. With the explosive growth of unstructured data, cloud storage technology gets more attention and better development. However, in current storage schema, user's data is totally stored in cloud servers. In other words, users lose their right of control on data and face privacy leakage risk.

Each layer of the scheme adds computational overhead, potentially impacting system performance, especially in real-time applications where low latency is crucial. Implementing a three-layer scheme introduces complexity to the system, requiring additional computational resources and potentially complicating system maintenance and troubleshooting.

3. PROPOSED WORK

It proposes a three-layer storage framework based on fog computing. The proposed framework can both take full advantage of cloud storage and protect the privacy of data. Besides, Hash-Solomon code algorithm is designed to divide data into different parts. Then, we can put a small part of data in local machine and fog server in order to protect the privacy. Moreover, based on computational intelligence, this algorithm can compute the distribution proportion stored in cloud, fog, and local machine, respectively

Advantages:

- By employing multiple layers of privacy-preserving mechanisms, the proposed system can provide stronger protection for sensitive data stored in the cloud, reducing the risk of unauthorized access or data breaches.
- The computational intelligence techniques utilized in the scheme can enhance security by detecting and mitigating potential threats or attacks more effectively compared to traditional methods

System Architecture

This comprehensive architecture provides a detailed overview of Cloud server holds and stores 95% of data, fog server stores 4% of data and client's local machine stores 1% of data of client's file when they uploaded into the cloud server.

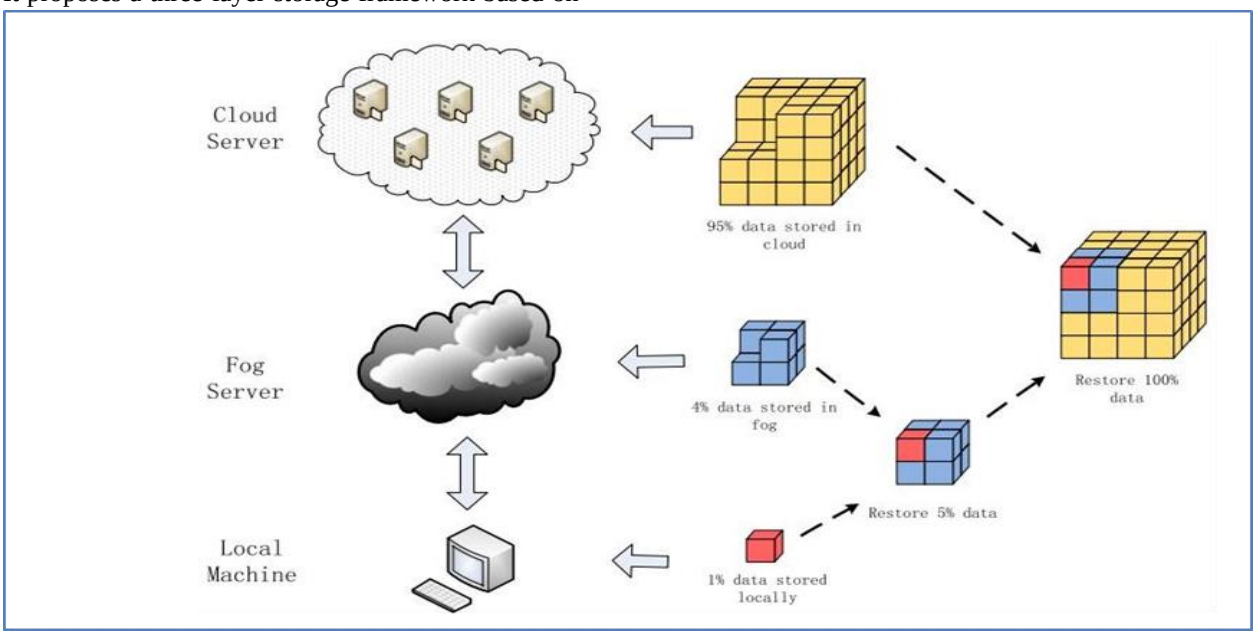

Fig.1 System Architecture

Tools

JAVA TECHNOLOGY

Initially the language was called as "oak" but it was renamed as "Java" in 1995. The primary motivation of this language was the need for a platform-independent (i.e., architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

- Java is a programmer's language.
- Java is cohesive and consistent.
- Except for those constraints imposed by the Internet environment, Java gives the programmer, full control.
- Finally, Java is to Internet programming where C was to system programming.

Importance of Java to the Internet

Java has had a profound effect on the Internet. This is because; Java expands the Universe of objects that can move about freely in Cyberspace. In a network, two categories of objects are transmitted between the Server and the Personal computer.

Java can be used to create two types of programs

Applications and Applets: An application is a program that runs on our Computer under the operating system of that computer. It is more or less like one creating using C or C++. Java's ability to create Applets makes it important. An Applet is an application designed to be transmitted over the Internet and executed by a Java –compatible web browser. An applet is actually a tiny Java program, dynamically downloaded across the network, just like an image. But the difference is, it is an intelligent program, not just a media file. It can react to the user input and dynamically change.

Features of Java Security

Every time you that you download a "normal" program, you are risking a viral infection. Prior to Java, most users did not download executable programs frequently, and those who did scan them for viruses prior to execution.

Most users still worried about the possibility of infecting their systems with a virus. In addition, another type of malicious program exists that must be guarded against. This type of program can gather private information, such as credit card numbers, bank account balances, and passwords. Java answers both these concerns by providing a "firewall" between a network application and your computer. When you use a Java-compatible Web browser, you can safely download Java applets without fear of virus infection or malicious intent.

Portability

For programs to be dynamically downloaded to all the various types of platforms connected to the Internet, some means of generating portable executable code is needed .As you will see, the same mechanism that helps ensure security also helps create portability. Indeed, Java's solution to these two problems is both elegant and efficient.

MYSQL

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by Oracle Company. MySQL database that provides for how to manage database and to manipulate data with the help of various SQL queries. These queries are: insert records, update records, delete records, select records, create tables, drop tables, etc. There are also given MySQL interview questions to help you better understand the MySQL database.

4. SYSTEM COMPONENTS

1. Cloud Storage:

At the core of the three-layer architecture lies cloud storage, a proven and reliable mechanism for storing vast amounts of data. Cloud storage offers features such as virtually unlimited scalability, high availability, and accessibility over the internet. It serves as the foundation for the entire system, providing a centralized repository for data storage and management.

2. Fog Computing:

Complementing cloud storage is the concept of fog computing, which extends the cloud's capabilities to the edge of the network, closer to where data is generated and consumed. Fog computing distributes computational resources and services across a heterogeneous network of devices, including routers, gateways, and edge servers. By processing data closer to the source, fog computing reduces latency,

conserves bandwidth, and enhances the overall efficiency of the system.

3. Three-Layer Architecture:

The three-layer architecture integrates cloud storage and fog computing into a cohesive framework, leveraging the strengths of both paradigms to optimize data storage and processing. This architecture consists of the following layers:

Cloud Storage Layer: This layer encompasses the traditional cloud storage infrastructure, comprising data centers located at geographically distributed locations. Cloud storage offers scalability, redundancy, and robust data management capabilities, making it ideal for storing large volumes of structured and unstructured data.

Fog Computing Layer: Situated at the edge of the network, the fog computing layer consists of a network of edge devices and servers deployed in proximity to data sources. These fog nodes perform computation, data preprocessing, and analysis tasks closer to the point of data generation, reducing latency and improving real-time responsiveness.

Integration Layer: Serving as the interface between cloud storage and fog computing, the integration layer orchestrates data movement, synchronization, and processing across the distributed architecture. It ensures seamless communication and coordination between cloud and fog resources, enabling efficient data workflows and workload management.

5. TESTING AND IMPLEMENTATION

1. Admin Login

The administrator is the controller of the site. The administrator creates the user account which needs user information like name, address, city, mobile, user id and password etc… The administrator can view the created users and can delete the unwanted users from the list.

2. View Uploads

The administrator can view the uploaded file list which are uploaded by the data owners (users of the system). The list contains owner name, date of upload, topic of upload and description etc…

3. File Upload

The file upload is done by the user of the system. He is the uploading user and the download of the file from the server. The upload form contains file topic, description, file and the fog server key, main server key and the own key. The fog server key and the main server key is automatically generated by the fog server and the main server. The own key is entered by the user. After all the information is correctly supplied, the file is uploaded. The uploading process is done in step by step. First the file is split up into two parts. The first part is around five percentage of the original size and the remaining part is around ninety-fivepercentages. Then the file is encrypted by the own key generated by the system and the five percentage of data is stored in the fog server and the remaining ninety-five percentage of data is stored in the main server.

4. Upload file list

The upload file list is shown to the user. The files which are uploaded by the current user is shown to that user. It includes the file information such as name, date of upload, etc… with the download and delete link. The download link navigates to another form which prompts for the main server key, fog server key and the own key for the particular file. If all of the supplied information is correct, then the file is downloaded. Otherwise the error message is shown to the user that which part of the password is incorrect.

5. Download file

The data owner login to the page, they can download the file through each server file key.

Implementation

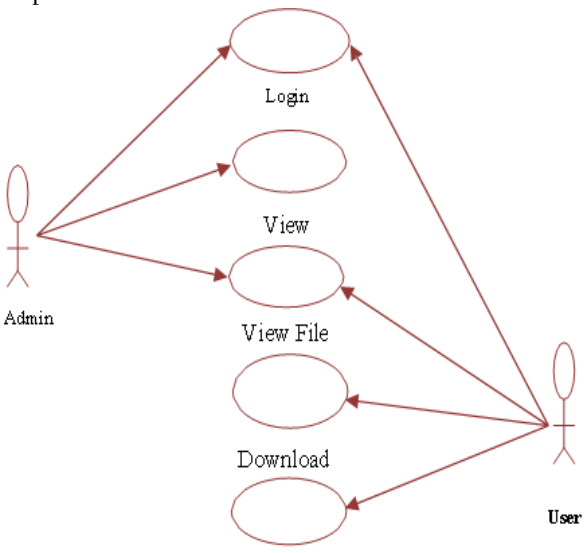

Fig.2 Use Case Diagram

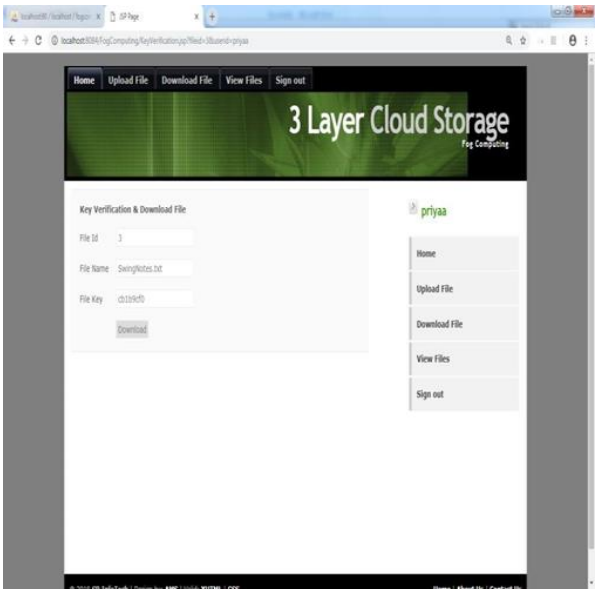

Fig.3 Interface Diagram

6. CONCLUSION AND FUTURE ENHANCEMENT

The development of cloud computing brings us a lot of benefits. Cloud storage is a convenient technology which helps users to expand their storage capacity. However, cloud storage also causes a series of secure problems. When using cloud storage, users do not actually control the physical storage of their data and it results in the separation of ownership and management of data. In order to solve the problem of privacy protection in cloud storage, we propose a TLS framework based on fog computing model and design a Hash-Solomon algorithm. Through the theoretical safety analysis, the scheme is proved to be feasible. By allocating the ratio of data blocks stored in different servers reasonably, we can ensure the privacy of data in each server. On another hand, cracking the encoding matrix is impossible theoretically. Besides, using hash transformation can protect the fragmentary information. Through the experiment test, this scheme can efficiently complete encoding and decoding without influence of the cloud storage efficiency.

6.1 Future Enhancements

Incorporating blockchain technology for decentralized authentication, access control, and data auditing could further enhance the security and integrity of the system, providing a tamper-proof and transparent mechanism for data management. Implementing algorithms that dynamically adjust the privacy levels based on contextual factors such as the sensitivity of the data, user preferences, and environmental conditions can optimize privacy protection while minimizing computational overhead. Developing more sophisticated data anonymization techniques that balance privacy protection with data utility, enabling organizations to share anonymized data for analysis and collaboration while preserving individual privacy.

REFERENCES

[1] Y. Zomaya and S. Olariu, "Handbook of Bioinspired Algorithms and Applications", Chapman and Hall/CRC, 2006.
[2] M. Satyanarayanan, "The Emergence of Edge Computing", Computer Magazine, vol. 50, no. 1, pp. 30-39, 2017.
[3] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities", Proceedings of the 7th High Performance Computing and Simulation Conference (HPCS 2009), pp. 1-11, 2009.
[4] Y. Zhou, Q. Zhang, and L. Zhu, "Spatio-Temporal Big Data Management in Fog Computing Environment", IEEE Transactions on Industrial Informatics, vol. 14, no. 12, pp. 5444-5452, 2018.
[5] M. Armbrust et al., "A View of Cloud Computing", Communications of the ACM, vol. 53, no. 4, pp. 50-58, 2010.
[6] S. Kamara and K. Lauter, "Cryptographic Cloud Storage", Proceedings of the 14th International Conference on Financial Cryptography and Data Security (FC 2010), pp. 136-149, 2010.