EFFICIENT DATA INTEGRITY ASSURANCE IN CLOUD SYSTEMS USING DISTRIBUTED MACHINE LEARNING

[1]Dr.MOHAMMAD SIRAJUDDIN, [2]Dr. P. VENKATESHWARLU, [3]KARABATHULA KEERTHIPRIYA

[1]Associate Professor,Department of MCA,Vaageswari College of Engineering, Thimmapur, Karimnagar,Telangana,India-505527 ,Email-id: mohdsiraj569@gmail.com.

[2]Professor &HOD,Department of MCA,Vaageswari College of Engineering, Thimmapur, Karimnagar,Telangana,India-505527,Email-id: venkateshwarlupurumala@gmail.com

[3]Research Scholar H.no:23S41F0050,Department of MCA,Vaageswari College of Engineering, Thimmapur, Karimnagar,Telangana,India-505527,Email-id: karabathulakeerthipriya22@gmail.com.

ABSTRACT

Distributed Machine Learning (DML) is a foundational technology in the field of Artificial Intelligence (AI). However, current DML frameworks often overlook the importance of data integrity. When network attackers tamper with, forge, or corrupt training data, the performance and reliability of the learning model can be severely compromised, leading to inaccurate results. To address this challenge, we propose a Data Integrity Verification scheme for Distributed Machine Learning (DML-DIV) to ensure the trustworthiness of training data.First, we incorporate the Provable Data Possession (PDP) sampling-based auditing technique to detect and defend against data forgery and tampering. Second, to protect data privacy during the Third Party Auditor (TPA) verification process, we introduce a randomly generated blinding factor and leverage the hardness of the Discrete Logarithm Problem (DLP) to construct secure proofs. Third, our scheme utilizes identity-based cryptography along with a two-step key generation mechanism to eliminate the key escrow issue and reduce certificate management overhead.Theoretical security analysis and experimental evaluations demonstrate that the proposed DML-DIV scheme is both secure and efficient.

Index Terms:- Distributed Machine Learning, Data Integrity, Provable Data Possession, Identity-Based Cryptography, Discrete Logarithm Problem, Privacy Protection.

1. INTRODUCTION

Artificial Intelligence (AI) has emerged as a major focus of research in both academia and the IT industry in recent years. AI technologies are increasingly applied to solve real-world problems such as personalized shopping recommendations, navigation systems, facial recognition, and autonomous driving. As a result, AI research holds both significant theoretical value and broad practical relevance.Machine Learning (ML), the core component of AI, plays a pivotal role in enabling systems to learn from data and make intelligent decisions. ML techniques are at the heart of various AI applications, including face recognition, route planning, and self-driving technologies.

However, traditional machine learning methods face significant challenges when processing massive datasets. These conventional approaches often lack the efficiency and scalability required to handle data volumes at the petabyte (PB) level or beyond. To overcome these limitations, leading technology companies such as Google and Microsoft have established specialized research centers dedicated to large-scale data-driven machine learning and AI development.In response to the growing demands of big data, distributed machine learning (DML) has gained prominence as a scalable and efficient solution. Recognizing its importance, institutions such as the Chinese Computer Society have identified DML as a key research direction aligned with the evolution of big data technologies.

2. LITERATURE SURVEY

J. Dean and S. Ghemawat introduced MapReduce, a programming model designed for processing and generating large datasets efficiently. Users define a map function to process key/value pairs into intermediate key/value pairs, followed by a reduce function to merge values with the same key. Developed in 2003 to build an inverted index for Google's search engine, the MapReduce framework has since supported over 10,000 programs at Google, including large-scale graph processing, text analysis, machine learning, and statistical machine translation. The open-source Hadoop implementation of MapReduce has further enabled widespread adoption beyond Google.

M. Zaharia, M. Chowdhury, and M. Franklin highlighted limitations in MapReduce for iterative and interactive applications, which are common in machine

learning and data analysis. To address these, they proposed Spark, a new framework that introduces Resilient Distributed Datasets (RDDs)—immutable collections partitioned across clusters, capable of being recomputed if lost. Spark retains the fault-tolerance and scalability of MapReduce while offering superior performance, often outperforming Hadoop by up to 10x in iterative tasks. It also supports sub-second response times for querying large datasets interactively.

Y. Low, J. E. Gonzalez, and A. Kyrola developed GraphLab, a parallel framework tailored to machine learning. Unlike MapReduce, which lacks expressiveness for asynchronous iterative algorithms, GraphLab targets typical ML patterns such as sparse computational dependencies and iterative updates. It enables efficient, parallel implementations of algorithms like belief propagation, Gibbs sampling, Lasso, and Co-EM. GraphLab achieves high parallel performance while ensuring data consistency, making it ideal for large-scale ML tasks.

C. Dehnert, I. Horn, N. Leiser, and G. Czajkowski focused on large-scale graph processing. They proposed a vertex-centric computational model where in each iteration, vertices process incoming messages, update states, and communicate with other vertices. This iterative approach abstracts distribution complexities behind a simple API, enabling scalable, fault-tolerant computation on graphs with billions of nodes and trillions of edges. The model supports a wide range of graph algorithms while simplifying the development process.

A. Smola and S. Narayanamurthy presented a high-performance distributed architecture for latent topic model inference. Their system leverages a distributed key-value store to synchronize the sampler state across nodes, avoiding the need for separate synchronization phases. By integrating disk I/O, CPU, and network usage, the architecture achieves high throughput and scalability. It handles hundreds of millions of documents and thousands of topics efficiently, and can be extended to more complex models like n-grams and hierarchical structures.

3.EXISTING SYSTEM:

Traditional machine learning technology exhibits poor efficiency when handling large data, particularly when the training data reaches the petabyte (PB) level or beyond. To address this challenge, renowned companies like Google and Microsoft have established large-data-based machine

learning and artificial intelligence research institutions to further explore distributed machine learning technology..

Drawbacks

1. There is a chance to forge the data, modify the data, or destroy the data by attackers.
2. The training model in the distributed machine learning system will be greatly affected.

4.PROPOSED SYSTEM:

To safeguard the integrity of training data in distributed machine learning systems, this paper introduces the Distributed Machine Learning-oriented Data Integrity Verification Scheme (DML-DIV). To the best of our knowledge, our DML-DIV scheme is the first of its kind in the distributed machine learning domain to utilize public sampling auditing algorithms, ensuring the integrity of training data.

Advantages:

- Addressing the issue of sensitive information sharing during the data integrity auditing process.
- Enhancing the robustness and reliability of training data, thereby improving the accuracy of distributed machine learning models.

5. SYSTEM MODEL

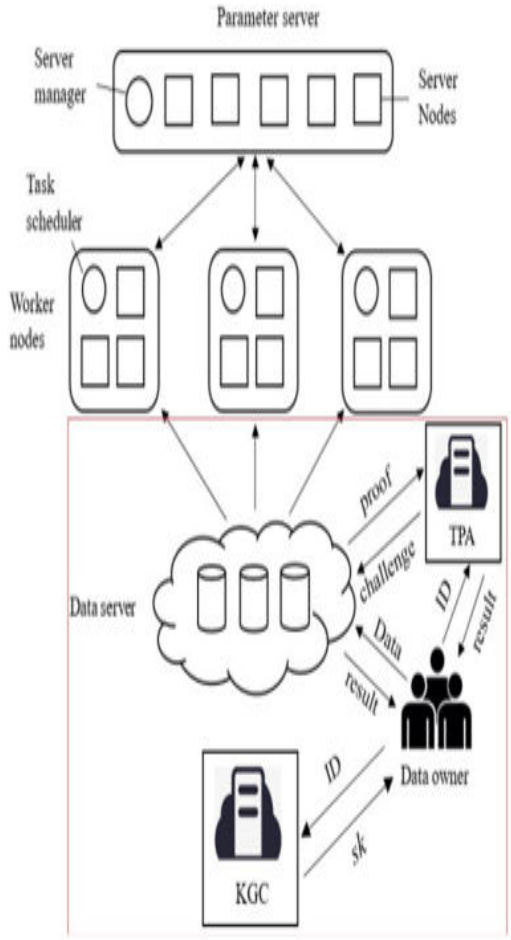


Figure 1. System model of our DML-DIV Scheme.

The Data Server (DS), Data Owner (DO), Third Party Auditor (TPA), and Key Generation Center (KGC) are the four organizations that make up our DML-DIV scheme, as seen in Fig. 1.

6.IMPLEMENTATION

- **Data Owner:**
  In this application, the data owner is a module. Here, the owner should register with the application and login. After successful login, they can perform operations such as uploading files, viewing files, viewing results, and logging out.

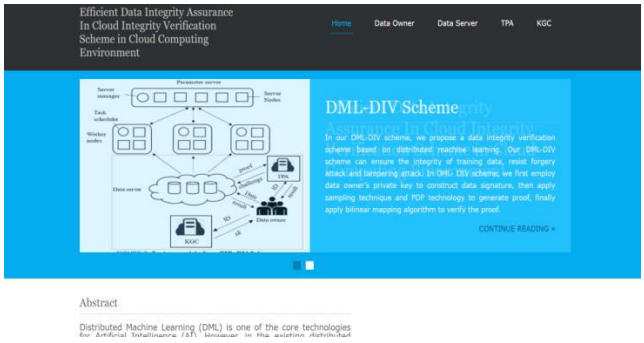
- **Key Generation Center (KGC):**
  Here, the KGC can directly log in to the application. After successful login, they can generate public keys and master keys for owners, and finally, they can log out.

- **Third Party Auditor (TPA):**
  Here, the TPA can directly log in to the application. After successful login, they can view all files uploaded by owners. Afterward, the TPA sends a verification request to the data server. Upon receiving the response from the data server, they can audit the files to check if they have been modified by attackers. Finally, they send the verification results to the data owner and log out.

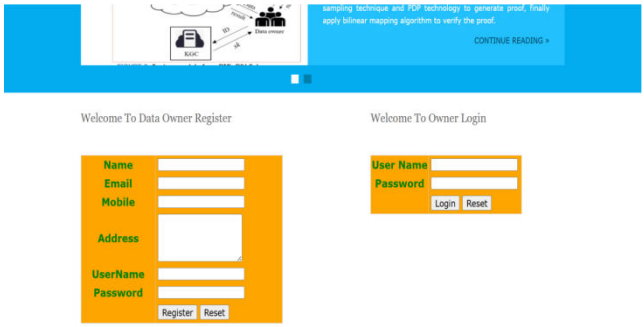
- **Data Server:**
  Here, the data server can log in to the application. The data server can view all owner details, file details, and challenges. Finally, they log out.

7.RESULTS

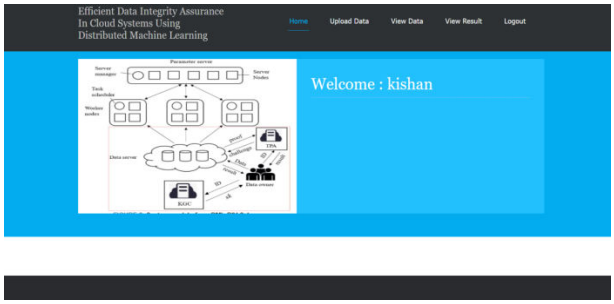
Home Screen



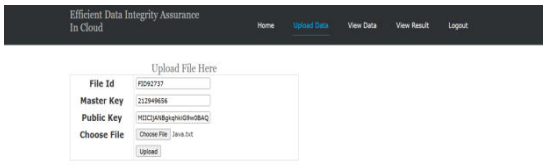
Data Owner



Data Owner Home Page

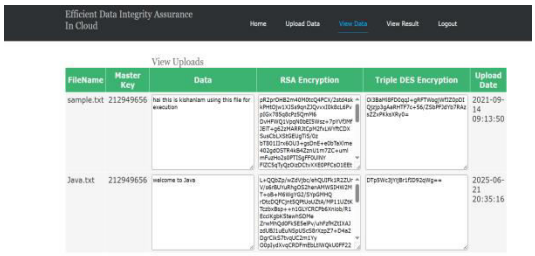


Upload Data



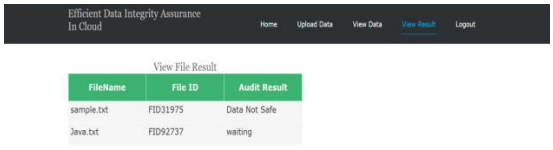
Server Home Page

View Data

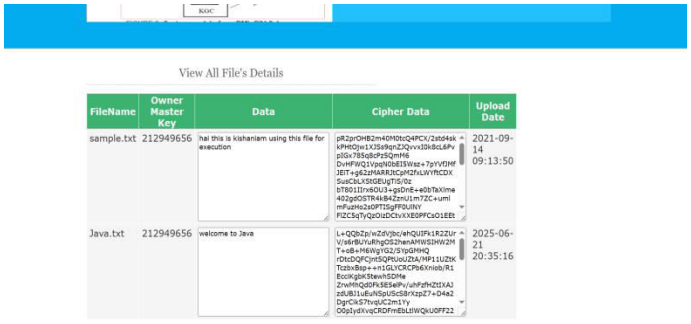


View Owners Details

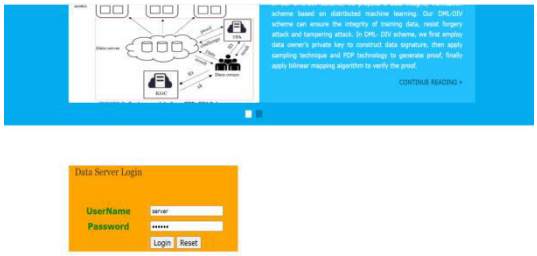
View Results



View Data

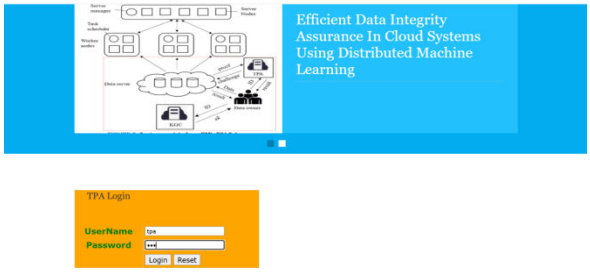


Data Server Login



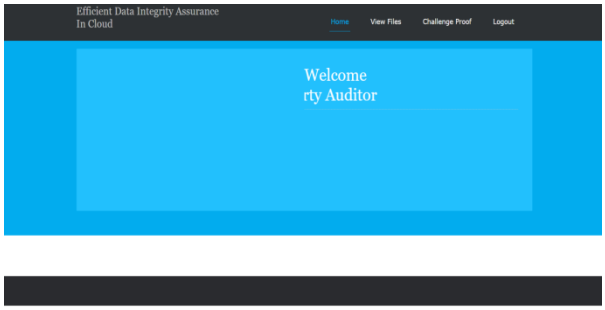
View Challenge

Tpa Login

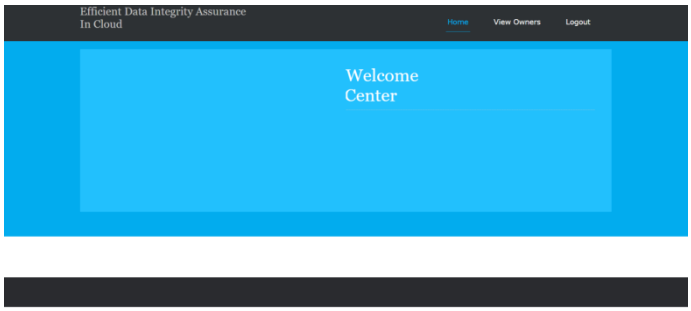


Kgc Login

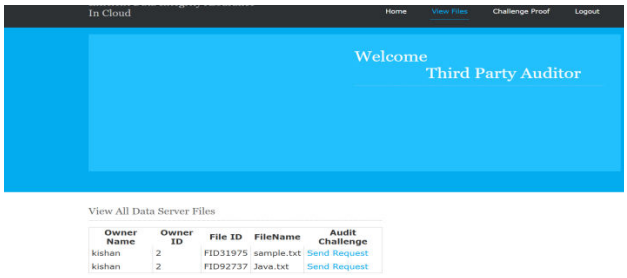
Tpa Home Page



Kgc Home Page

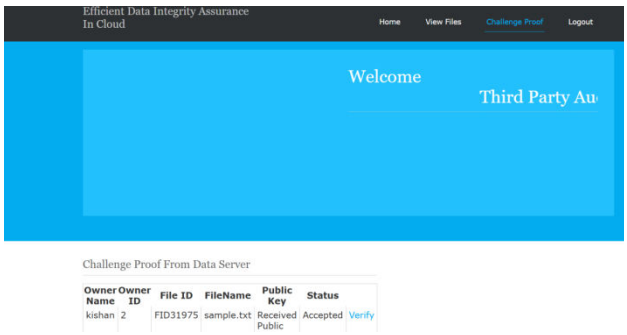


View Files

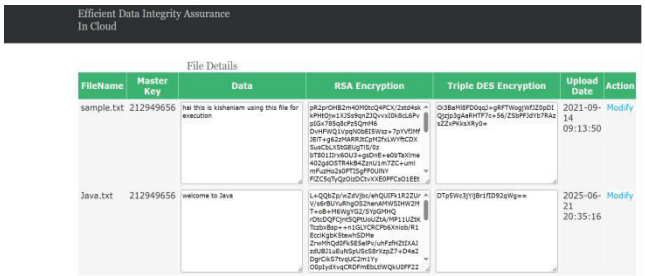


View Owners

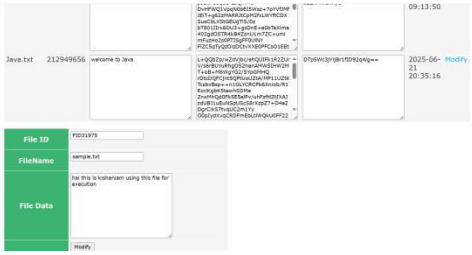
View Challenge Proof



Attacker Page



Modification File



8.CONCLUSION

In this paper, we propose a distributed machine learning oriented data integrity verification scheme (DML-DIV) for the parameter server framework. Our DML-DIV scheme can ensure the integrity of the training data stored in the data server, and resist forgery attack and tampering attack. Additionally, our DML-DIV scheme provides privacy protection, solves the key escrow problem, and reduces the cost of managing the certificates. Finally, the simulation results show our DML-DIV scheme performs more efficiently than other schemes.

9. FUTURE ENHANCEMENT

In future work we are going to add attackers; Here attacker can directly access our application through URL, after access our page he can view all uploaded files from data server and he can able to modify the files and stores at the same server's database.And also we adding another facility to data server which is analysis graph, here data server able to view graph on files which are attacked file and which are not attacked files.Here we are adding one more algorithm i.e. triple des to provide more security to the data which is stored into the cloud computing environment.

10. REFERENCES

[1] J. Dean and S. Ghemawat, ''MapReduce: A flexible data processing tool,'' Commun. ACM, vol. 53, no. 1, pp. 72–77, 2010.

[2] M. Zaharia, M. Chowdhury, and M. Franklin, ''Spark: Cluster computing with working sets,'' in Proc. 2nd USENIX Conf. Hot Topics Cloud Comput., 2010, pp. 1–7.

[3] Y. Low, J. E. Gonzalez, and A. Kyrola, ''GraphLab: A new framework for parallel machine learning,'' Comput. Sci., vol. 31, no. 1, pp1-4, 2004.

[4] G. Malewicz, M. H. Austern, A. J. Bik, J. C. Dehnert, I. Horn, N. Leiser, and G. Czajkowski, ''Pregel: A system for large-scale graph processing,'' in Proc. ACM SIGMOD Int. Conf. Oil Manage. Data, 2010, pp. 135–146.

[5] A. Smola and S. Narayanamurthy, ''An architecture for parallel topic models,'' Proc.

VLDB Endow., vol. 3, nos. 1–2, pp. 703–710, Sep. 2010.

[6] J. Dean, G. S. Corrado, and R. Monga, ‘‘Large scale distributed deep networks,’’ in Proc. Int. Conf. Neural Inf. Process. Syst. Red Hook, NY, USA: Curran Associates, 2013, pp. 1223–1231. [7] (2018). Douban Paracel. [Online]. Available: http://paracel.io/ [8] M. Li, ‘‘Scaling distributed machine learning with the parameter server,’’ in Proc. Int. Conf. Big Data Sci. Comput. (BigDataSci), 2014, pp. 583–598.

[9] M. Li, Z. Li, and A. Smola, ‘‘Parameter server for distributed machine learning,’’ in Proc. Big Learn. NIPS Workshop, 2013, pp. 1–10.

[10] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, ‘‘Provable data possession at untrusted stores,’’ in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), 2007, pp. 1–25.

[11] Q. Zheng, S. Xu, and G. Ateniese, ‘‘Efficient query integrity for outsourced dynamic databases,’’ in Proc. ACM Workshop Cloud Comput. Secur. Workshop (CCSW), 2012, pp. 71–82.

[12] C. C. Erway, A. Küpçü, and C. Papamanthou, ‘‘Dynamic provable data possession,’’ ACM Trans. Inf. Syst. Secur., vol. 17, no. 4, pp. 1–29, 2009.

[13] C. Wang, Q. Wang, K. Ren, and W. Lou, ‘‘Privacy-preserving public auditing for data storage security in cloud computing,’’ in Proc. IEEE INFOCOM, Mar. 2010, pp. 525–533.

[14] Y. Zhu, H. Hu, G.-J. Ahn, and S. S. Yau, ‘‘Efficient audit service outsourcing for data integrity in clouds,’’ J. Syst. Softw., vol. 85, no. 5, pp. 1083–1095, May 2012.

[15] C. Wang, S. S. M. Chow, and Q. Wang, ‘‘Privacy-preserving public auditing for secure cloud storage,’’ IEEE Trans. Comput., vol. 62, no. 2, pp. 362–375, Dec. 2013.

[16] H. Yan, J. Li, J. Han, and Y. Zhang, ‘‘A novel efficient remote data possession checking protocol in cloud storage,’’ IEEE Trans. Inf. Forensics Security, vol. 12, no. 1, pp. 78–88, Jan. 2017.

[17] M. Sookhak, F. R. Yu, and A. Y. Zomaya, ‘‘Auditing big data storage in cloud computing using divide and conquer tables,’’ IEEE Trans. Parallel Distrib. Syst., vol. 29, no. 5, pp. 999–1012, May 2018.

[18] H. Zhao, X. Yao, X. Zheng, T. Qiu, and H. Ning, ‘‘User stateless privacypreserving TPA auditing scheme for cloud storage,’’ J. Netw. Comput. Appl., vol. 129, pp. 62–70, Mar. 2019. [19] H. Yan, J. Li, and Y. Zhang, ‘‘Remote data checking with a designated verifier in cloud storage,’’ IEEE Syst. J., to be published, doi: 10.1109/jsyst.2019.2918022.

[20] H. Wang, ‘‘Identity-based distributed provable data possession in multicloud storage,’’ IEEE Trans. Services Comput., vol. 8, no. 2, pp. 328–340, Mar. 2015.

Traditional machine learning technology exhibits poor efficiency when handling large data, particularly when the training data reaches the petabyte (PB) level or beyond. To address this challenge, renowned companies like Google and Microsoft have established large-data-based machine learning and artificial intelligence research institutions to further explore distributed machine learning technology.