Mathematical Statisticianand Engineering Applications
ISSN:2094-0343
2326-9865

Enhancing Data Integrity in Multi-Cloud Storage with Identity-Based Distributed Provable Data Possession

[1]Naveen Reddy Nemili, [2] Dr. Narendra Sharma,[3]Sanjeev Shrivastava

[1]Research Scholar, Dept. of Computer Science & Engineering, Sri Satya Sai University of Technology & Medical Sciences, Sehore, Bhopal-Indore Road, Madhya Pradesh, India

[2]Supervisor, Dept. of Computer Science & Engineering, Sri Satya Sai University of Technology & Medical Sciences, Sehore, Bhopal-Indore Road, Madhya Pradesh, India

[3]Co-Supervisor, Professor in Dept. of Computer Science & Engineering, Guru Nanak Institute of Technology, Hyderabad, Telangana, India

Article Info
Page Number: 2137 - 2144
Publication Issue:
Vol.71 No.3 (2022)

Abstract

Cloud services have gained popularity due to numerous companies offering a wide range of cloud-based solutions tailored to user needs. However, the proliferation of cloud services has brought about an increase in security concerns, such as data breaches and unauthorized data alterations. This paper specifically addresses the issue of data integrity. Provable Data Possession (PDP) schemes, while effective in ensuring data integrity, suffer from a drawback in terms of time overhead, especially when dealing with dynamic data that requires additional processing time. To tackle this problem, this paper explores the utilization of Counting Bloom Filters for data updates, which proves to be a more efficient solution compared to PDP when modifying a portion of data within a simulated environment. The approach yields better results when safeguarding data that experiences alterations within dynamic datasets. Identity-Based Provable Data Possession (PDP) stands as a technique for preserving data integrity in the context of outsourcing data storage to distributed cloud service providers. This technique aims to construct an efficient scheme for distributed cloud storage that supports service scalability and data migration, where multiple cloud service providers collaborate to securely store and manage clients' data. The advent of cloud computing has significantly transformed the computer industry by offering information processing as a service, encompassing functions like storage and computation. In this landscape, data integrity plays a critical role, particularly when clients need to store their data, whether it be images or text, across multiple cloud servers. When clients opt for multi-cloud storage, ensuring distributed storage and integrity checks becomes paramount. To address these concerns, this paper introduces an Identity-Based Distributed Provable Data Possession (ID-DPDP) protocol tailored for multi-cloud storage. Remote data integrity checks play a pivotal role in cloud storage as they empower clients to verify their data's integrity without the need to download the entire dataset. In various application scenarios, clients are compelled to store their data across multiple cloud servers. Simultaneously, the integrity checking protocol must exhibit efficiency to minimize the costs incurred by verifiers.

Keywords: Cloud computing, Provable data possession, Identity-based cryptography, distributed computing, bilinear pairings.

Article History
Article Received: *12 January 2022*
Revised: *25 February 2022*
Accepted: *20 April 2022*
Publication: *09 June 2022*

Introduction

In recent years, cloud computing has emerged as a significant topic within the field of computer science. Essentially, it offers information processing as a service, encompassing functions such as storage and computing. This approach not only relieves the responsibility of managing storage but also provides universal data access from geographically diverse locations. Simultaneously, it eliminates the need for substantial capital investments in hardware, software, and personnel maintenance. Consequently, cloud computing has garnered increasing attention from enterprises. The core concept of cloud computing revolves around outsourcing computing tasks to third-party providers. However, this outsourcing introduces security risks related to the confidentiality, integrity, and availability of data and services. Ensuring cloud clients that their data remains unaltered is of utmost importance, particularly since clients don't maintain these data locally. Remote data integrity checks serve as a fundamental solution to address this concern. In the typical scenario where clients store their data on multiple cloud servers, distributed storage and integrity checks become indispensable. In parallel, it's crucial for the integrity checking protocol to be efficient to accommodate devices with limited capacity. Therefore, leveraging distributed computation, our study will delve into a model for distributed remote data integrity checking and present a concrete protocol suitable for multi-cloud storage.

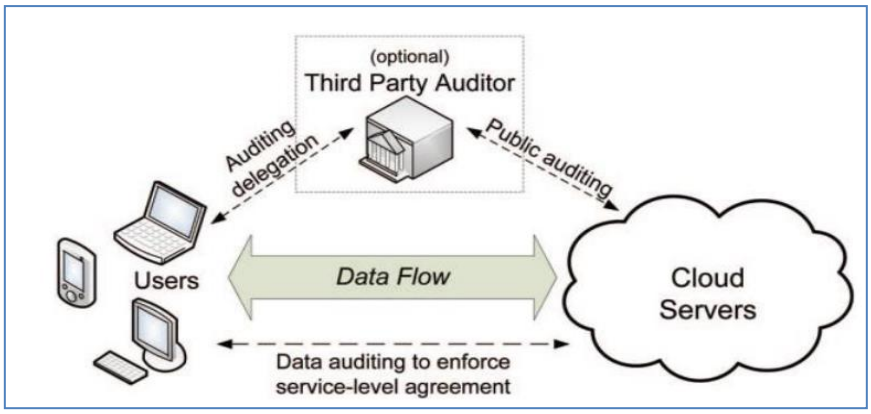

Figure Cloud Storage Service Architecture

In the context of Public Key Infrastructure (PKI), the Provable Data Possession Protocol necessitates the distribution and management of public key certificates. This results in significant overhead because the verifier must check the certificate whenever verifying the integrity of remote data. Additionally, the system grapples with the complexities of managing various certificates, including certificate generation, delivery, revocation, renewals, and more. Cloud computing often involves verifiers with limited computational capacity. Identity-based public key cryptography offers a solution by simplifying certificate management. To enhance efficiency, Identity-Based Provable Data Possession (ID-DPDP) becomes a more attractive option. Therefore, researching ID-DPDP is of considerable significance. Cloud computing, a novel form of internet-based computing, offers convenient on-demand network access, enabling users to easily access resources such as networks, servers, storage, and application software. With secure payment methods, rapid resource provisioning, and minimal interaction with service providers, cloud computing has rapidly

gained popularity. Cloud storage, a fundamental service within cloud computing, allows individuals and organizations to store their data on cloud servers, eliminating the need for purchasing and maintaining storage devices. Users can access their files at any time and from any location, facilitating file sharing. However, the increasing adoption of cloud storage has brought to light potential security issues, including data loss incidents. Data loss and leakage, as reported by the Cloud Vulnerabilities Working Group of the Cloud Security Alliance, ranks as the second significant threat in the cloud computing domain. Data loss incidents, while relatively small compared to the overall data stored, can be alarming for anyone running a website. Due to concerns about the trustworthiness of cloud servers, ensuring data integrity becomes a primary concern for data users. Strong guarantees of data integrity are essential because cloud servers may not immediately inform users about data loss incidents to maintain their reputation. To address these concerns, the Provable Data Possession (PDP) model was introduced in 2007, which verifies data integrity through random sampling of data blocks. Several efficient and secure PDP schemes based on the RSA assumption were presented. Proof of Retrievability (PoR) was also introduced, enabling concise proof generation that a user can recover a target file in its entirety. As the cloud storage market has expanded, various cloud auditing protocols and their variants have been proposed, accommodating different requirements such as privacy preservation, public verification, and dynamic operations. However, most proposals assume users store files on a single cloud server. With the rise of multi-cloud storage options, extending existing auditing protocols to support multi-cloud auditing presents challenges due to increased communication and computation overhead. Furthermore, many PDP schemes rely on Public Key Infrastructure (PKI), which involves complex and costly key management and certificate maintenance, leading to low computational efficiency. To enhance PDP efficiency in multi-cloud auditing, Zhu et al. proposed a cooperative PDP protocol for distributed data integrity verification. However, Wang and Zhang demonstrated security vulnerabilities in this protocol, wherein a malicious server could generate a valid proof even after removing all stored data. To eliminate the reliance on PKI in PDP schemes, Zhao et al. designed an identity-based public verification scheme for secure cloud storage. More recently, Wang introduced the concept of an Identity-Based Distributed PDP scheme (ID-DPDP) for multi-cloud storage. ID-DPDP offers private, delegated, and public verification simultaneously. In this novel approach, digital certificates are eliminated, and a Combiner is introduced to distribute block-tag pairs and challenges to various cloud servers, combining the proofs from different servers and forwarding them to the verifier.

Contributions

In this paper, we illustrate that the ID-DPDP scheme falls short of achieving the essential property of soundness, which is the most coveted characteristic a PDP protocol should possess. In essence, this implies that a malicious cloud server has the potential to mislead a verifier into believing it has maintained the complete file's integrity when, in reality, it relies solely on hash values of data blocks and their corresponding tags, rather than the actual file content. Consequently, the server could dispose of the data and reclaim storage space for economic gain. Furthermore, the ID-DPDP scheme with public verification described here

deviates from the definition of identity-based cryptography because, in addition to the user's identity (ID), a verifier must request an additional element, denoted as R, from the data owner to validate the proof. Subsequently, we propose a straightforward yet effective solution to address these limitations. Specifically, we introduce a comprehensive construction of an identity-based Provable Data Possession (ID-PDP) by amalgamating general signatures with standard PDP protocols, and provide a security analysis of this generic approach.

Literature Review

Hongyu Liu et al (2017), Provable Data Possession (PDP) is a crucial feature for cloud storage, allowing users to verify data integrity without the need to retrieve the entire file. Notably, all existing PDP schemes rely on Public Key Infrastructure (PKI). However, Wang introduced an innovative identity-based distributed Provable Data Possession (ID-DPDP) scheme that offers the advantages of eliminating complex certificate management and application in multi-cloud scenarios. This scheme is characterized by efficiency, flexibility, and support for various verification methods, including private, delegated, and public verification. This paper identifies a flaw in the ID-DPDP scheme as it falls short of achieving soundness. To rectify this issue, we propose a generic construction for an identity-based Provable Data Possession (ID-PDP) protocol, derived from secure digital signature schemes and traditional PDP protocols. We establish that the soundness of the generic ID-PDP construction depends on the security of the underlying PDP protocols and signature schemes. We provide an instance of this generic construction by utilizing a state-of-the-art PDP protocol by Shacham and Waters and the BLS short signature scheme. Furthermore, we extend the basic ID-PDP to create a new ID-DPDP protocol suitable for multiple cloud environments. Our implementation demonstrates the efficiency of the proposed ID-PDP protocol.

Ieuan Walker et al (2022), In recent years, there has been a noticeable increase in the practice of storage outsourcing, where the idea of entrusting third-party data warehouses has gained significant traction. This growing trend has raised various intriguing privacy and security concerns. One of the foremost worries associated with third-party data storage providers is the issue of accountability. This article takes a critical look at two schemes or algorithms that offer users the means to verify the integrity and availability of their data that is stored on untrusted data repositories, namely, third-party data storage services. The two schemes under review are Provable Data Possession (PDP) and Proofs of Retrievability (POR). Both of these cryptographic protocols have been developed to provide clients with the assurance that their data remains secure on these untrusted data storage platforms. Additionally, the article introduces a conceptual framework designed to address the vulnerabilities present in current storage solutions.

Research Methodology

The research and data collection methods employed in this study are detailed below. The primary research approach utilized is inductive, as the goal is to formulate a conceptual framework based on an analysis of sample literature. The review is conducted using publicly available secondary data sources that address various aspects of data validation and recovery

from untrusted third-party storage providers. The primary data sources for this review include repositories such as the SCOPUS library, Web of Science (WoS) citation database, ACM library, IEEE Explorer, Google Scholar, Research Gate, and others. Several keyword searches were conducted to locate relevant studies and reviews essential for addressing the research questions in this article. The main keyword combinations encompass terms like "Data integrity," "Cloud storage," "Data retrievability," "Validating data," and other pertinent keywords. No exclusion criteria were applied. In terms of the schemas under scrutiny, they were selected for analytical review based on the number of references identified across all keyword combinations. In addition to the keyword search conducted by the authors, recommendations from previously published research, tutorials, surveys, and reviews were considered when choosing which schemas to focus on in this review. The Provable Data Possession (PDP) and Proofs of Retrievability (POR) schemas have been thoroughly analyzed, discussed, and summarized. Academic papers from the literature on each schema were arranged chronologically, with selection based on their contributions to the overall schema and the number of journal papers that cited them..

Provable Data Possession (PDP)

Provable Data Possession (PDP) serves as a method to offer tenants the capability to validate the integrity of their data stored on untrusted storage platforms, all without the need to download the actual data. In a concise summary of the PDP model, the process involves the client pre-processing the data and transmitting it to an untrusted data repository, while retaining only a small set of metadata for future reference. Subsequently, the client can request the storage provider to demonstrate that the data they originally submitted remains unaltered and hasn't been deleted. All of this can be accomplished without the necessity of downloading the entire file.

Provable Data Possession Review

In 2007, a group of authors introduced a novel scheme referred to as 'Provable Data Possession' or simply PDP. This scheme offers a solution to individuals who have placed their files on an untrusted storage platform, allowing them to verify the data's integrity and ensure it hasn't been tampered with, all without the need to download the actual file. This feature is particularly significant in today's landscape, where users frequently utilize multiple third-party data storage services like Google Drive, OneDrive, and Dropbox. The primary objective of this scheme is to swiftly assess the integrity of files, achieving this by utilizing a minimal amount of metadata that tenants themselves maintain.

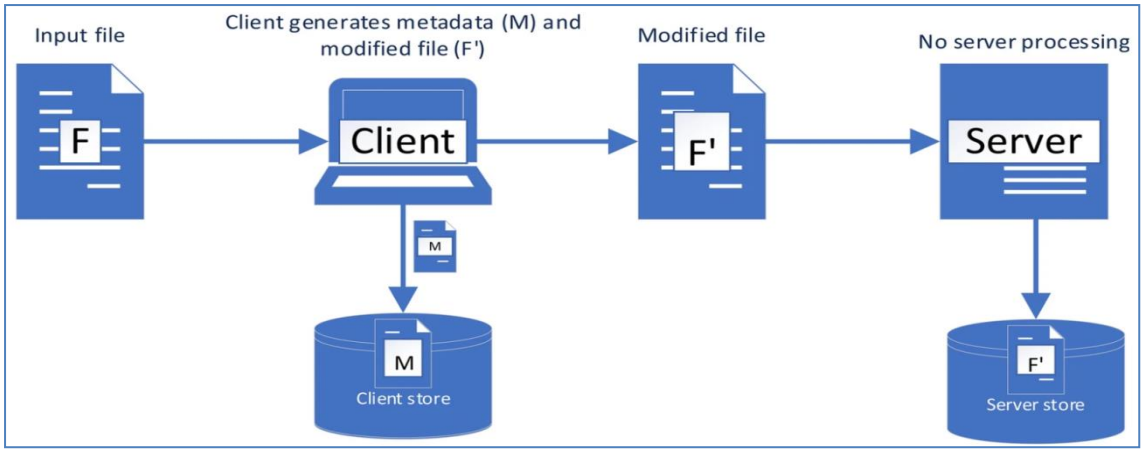

Figure PDP Pre-Process and Store Diagram

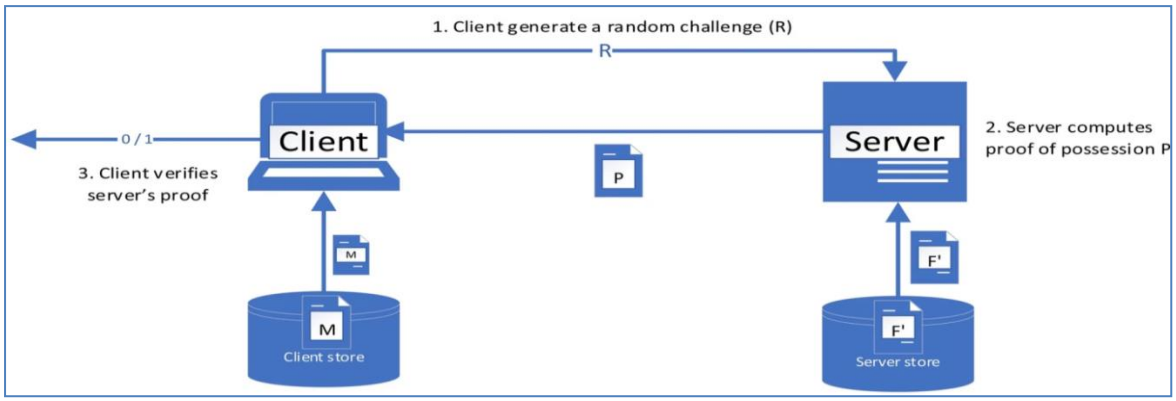

Figure PDP Verify Server Possession Diagram

The below pseudo code that outlines the process of preparing the file for upload to the server.

1. Break the file into n blocks (F1, F2, ..., Fn).

2. Pre-process the file (F'1, F'2, ..., F'n) and generate metadata (M).

3. Store the metadata (M) at the client side.

4. Transfer the pre-processed file (F') to the server.

5. Delete the local copy of the file.

The below provided pseudo code for verifying the file.

1. Issue a random challenge (R) to the server to ensure the server has retained the file.

2. Request the server to compute a function of the stored file.

3. The server sends back the response (P, the proof of possession) to the client.

4. Use the local metadata (M) to verify the response.

The primary limitation of the proposed PDP scheme is its restriction to static data. This means that if a client intends to make changes to the data, they must initiate the PDP scheme

anew from the beginning. In response to this constraint, Ateniese and colleagues introduced a dynamic PDP framework known as Scalable PDP, which allows for somewhat limited dynamic data operations. Specifically, it permits appending, modifying, and deleting blocks, although it does not support the insertion of blocks. The scheme comprises two key phases, namely setup and verification (referred to as a challenge in the literature), reminiscent of the schema by Guiseppe et al. However, the innovative aspect the authors introduce to the PDP field is the concept of generating all future challenges during the setup phase and storing the pre-computed responses as metadata on the client's end. In this setup, the data owner (OWN) proactively generates a set of potential random challenges and their corresponding responses. This approach imposes limitations on the number of updates and challenges the client can perform and has the effect of prohibiting block insertions at any point. Clients can only append blocks. The authors acknowledge this limitation by highlighting "one potentially glaring drawback of our scheme is the prefixed (at setup time) number of verifications t." They further note that the only way to increase the number of challenges and updates is to restart the setup phase, which would necessitate the client (OWN) to retrieve the entire file (D) from the server (SVR). However, this approach becomes problematic and impractical, particularly for large files. In 2009, Erway and colleagues presented a research paper titled 'Dynamic Provable Data Possession (DPDP),' building upon the provable data possession model (PDP) while extending its capabilities to support deletion, modification, and insertion of data. This represents a significant improvement compared to earlier work. Notably, their work closely followed the proposed Scalable PDP scheme by Ateniese et al. Both papers draw upon the foundational work of Guiseppe et al. However, the key distinction between Scalable PDP and DPDP is that Ateniese et al. employ a random oracle model, whereas the DPDP scheme is "provably secure in the standard model." To illustrate these differences, the authors provide a summary table comparing PDP, Scalable PDP, DPDP I, and DPDP II.

Conclusion

We have introduced the development of a highly efficient Provable Data Possession (PDP) scheme tailored for distributed cloud storage. This scheme leverages homomorphic verifiable response and hash Index hierarchy, providing the foundation for a cooperative PDP scheme that supports dynamic scalability across multiple storage servers. Importantly, we have established that our scheme encompasses all the essential security properties required by a zero knowledge interactive proof system, making it robust against various attacks, even when deployed as a public audit service in cloud environments. Our experiments have unequivocally shown that our approach introduces only minimal computational and communication overheads. Consequently, our solution emerges as a strong contender for the verification of data integrity in outsourced data storage systems. In the realm of multi-cloud storage, this paper formalizes the ID-DPDP system model and security model. Simultaneously, we introduce the first ID-DPDP protocol, which can be provably secured under the assumption that the Computational Diffie-Hellman (CDH) problem is challenging. Notably, besides eliminating the complexities of certificate management, our ID-DPDP protocol is characterized by its flexibility and high efficiency. Moreover, the proposed ID-

DPDP protocol accommodates private verification, delegated verification, and public verification based on the client's authorization.

References

[1]. Anthoine G, Jean-Guillaume D, Michael H, Mélanie de J, Aude M, Clément P, Daniel R. Dynamic proofs of retrievability with low server storage. CoRR. 2021. arXiv: 2007.12556.

[2]. Armknecht F, Barman L, Bohli J, Karame GO. Mirror: enabling proofs of data replication and retrievability in the cloud. USENIX Security Symposium. 2016.

[3]. Ateniese G, Pietro RD, Mancini LV, Tsudik G. Scalable and efficient provable data possession. Istanbul: ACM; 2008. p. 91–910.

[4]. Barsoum AF, Hasan AM. Provable multicopy dynamic data possession in cloud computing systems. IEEE Trans Inf Forensics Secur. 2014;10(3):485–97

[5]. Chen R, Li Y, Yu Y, Li H, Chen X, Susilo W. Blockchain-based dynamic provable data possession for smart cities. IEEE Internet Things J. 2020;7(5):4143–54.

[6]. Etemad M, Kupcu A. Transparent, distributed, and replicated dynamic provable data possession. Berlin: Springer-Verlag; 2013. p. 1–18.

[7]. Guo W, et al. Improved proofs of retrievability and replication for data availability in cloud storage. Comput J. 2020;63(8):1216–30.

[8]. Ieuan W. Provable Data Possession (PDP) and Proofs of Retrievability (POR) of current big user data. MPhil Thesis, Cardiff Metropolitan University, UK. 2021.

[9]. Liu J, Tong J, Mao R, Bohn J, Messina L, Badger D. Leaf, NISTCloud computing reference architecture NIST Special Publication; 2011. p. 500–292.

[10]. Liu Y, et al. New provable data transfer from provable data possession and deletion for secure cloud storage. Int J Distrib Sens Netw. 2019.

[11]. Wei G, Hua Z, Sujuan Q, Fei G, Zhengping J, Wenmin L, Qiaoyan W. Improved Proofs of Retrievability and replication for data availability in cloud storage. Comput J. 2020

[12]. Y. Zhu, H. Wang, Z. Hu, G. J. Ahn, H. Hu. ZeroKnowledge Proofs of Retrievability.Sci China InfSci, 54(8):1608-1617, 2011.

[13]. Y. Zhu, H. Hu, G.J. Ahn, M. Yu. Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage IEEE Transactions on Parallel and Distributed Systems, 23(12):2231-224, 2012.

[14]. Y. Zhu, H. Wang, Z. Hu, G. J. Ahn, H. Hu, S. S. Yau. Efficient Provable Data Possession for Hybrid Clouds.CCS'10, 756-758,2010.

[15]. Huaqun Wang, Identity-Based Distributed Provable Data Possession in Multicloud Storage, IEEE Transactions on Services Computing.