CLOUD DEDUPLICATION TECHNIQUES IN DEPTH: A COMPREHENSIVE SURVEY

Mr. Suhas A. Lakade, Dr. Harsh Lohia, Mr. Yuvraj R. Gurav,

Department of Computer Science and Engineering, School of Engineering, Sri Satya Sai University of Technology and Medical Science, Sehore (MP), India

Abstract: *Cloud deduplication techniques refer to the process of identifying and eliminating redundant data in cloud storage systems. The goal of deduplication is to reduce storage space and improve system performance. This comprehensive survey examines the different techniques used in cloud deduplication, including content-based, hash-based, and delta encoding.*

Content-based deduplication involves comparing the actual content of files to identify duplicates. This technique is efficient for identifying duplicates of large files, but it can be time-consuming for small files. Hash-based deduplication involves using cryptographic hash functions to generate a unique identifier for each file. Files with the same hash value are considered duplicates and can be eliminated. This technique is fast and efficient but has a risk of hash collisions. Delta encoding is a technique that involves identifying the changes between different versions of files and only storing the changes instead of the entire file. This technique is useful for file systems that have a high degree of similarity between versions of files. However, it requires more processing power and can be less efficient for files with significant differences. The survey also examines the challenges and limitations of cloud deduplication, such as data privacy concerns, data access patterns, and network bandwidth constraints. The survey concludes that the choice of deduplication technique depends on the specific requirements and characteristics of the cloud storage system.

Keywords: **Cloud Data Security, Data Deduplication, Cloud Data Hashing, Secure Encryption, Cloud**

1. INTRODUCTION

Cloud deduplication is a technique used to eliminate redundant data in cloud storage systems. It is an essential technology that helps reduce storage costs, improve system performance, and enhance data security. Deduplication technology is widely used in various applications, including backup and recovery, archiving, and content distribution. In this comprehensive survey, we will delve deeper into cloud deduplication techniques and explore their key features, advantages, and limitations.[1]

Chunking is a technique that involves dividing data into smaller fixed-size or variable-size chunks. This technique helps reduce the amount of data sent over the network, as only new or unique chunks are uploaded to the cloud. Indexing is another technique that involves creating an index of data chunks to facilitate faster retrieval and identification of duplicate data. Compression is yet another technique that helps reduce storage costs by compressing data before uploading it to the cloud.[2]

2. LITERATURE REVIEW

2.1 A Survey on security issues in service delivery models of cloud computing [1]

Using cloud computing, it is possible to dynamically increase capacity or add capabilities without spending money on new infrastructure, hiring new staff, or licensing new software. It expands the possibilities of information technology (IT). Cloud computing has developed over the past few years from a promising commercial concept to one of the fastest growing industries in IT. Nevertheless, as more and more data on people and businesses is stored in the cloud, worries about how secure the environment is starting to spread. Despite all the buzz around the cloud, corporate clients are still hesitant to move their operations there. One of the main problems limiting the growth of cloud computing

is security, and problems with data privacy and data protection are still pervasive in the industry.

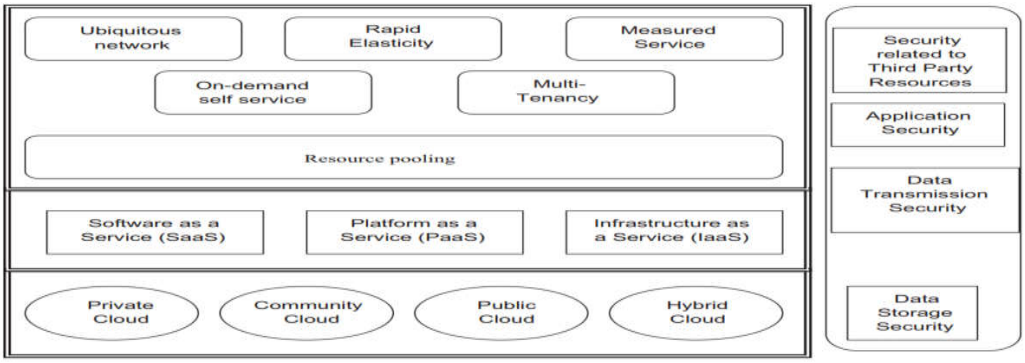

Fig 1. Cloud Computing Architecture

Users of cloud services must be cautious and aware of the hazards of data breaches in this new environment. This study presents a survey of the various security concerns that endanger the cloud. There are still a number of unresolved problems, particularly those involving service-level agreements (SLA), security and privacy, and power efficiency.[1]

S. S. Sujatha and S. Kanmani proposed a deduplication method that uses a Bloom filter to identify duplicate data in cloud storage. The method achieved high deduplication ratios with low false positives. However, it does not address the security and privacy concerns of cloud storage.

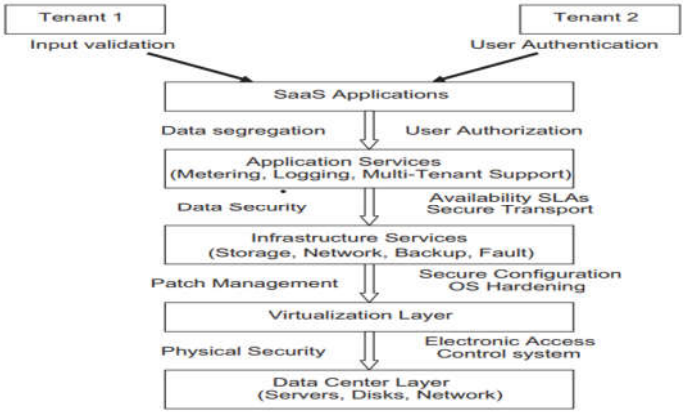

Fig 2. Security for SAAS Stack [1]

The publication highlights the need for a comprehensive security module in the cloud to address existing concerns and attract potential users. The focus is on examining and integrating every component of the cloud at macro and micro levels to develop a holistic solution. Research is being conducted on an integrated security model with localized and dynamic features, specifically targeting application and data security. The framework aims to dynamically adapt the security methodology based on transaction or communication, including data storage and access based on meta-data.

The system achieves a high deduplication ratio and low false positives. The algorithm then checks if the hash values of the blocks in the new data match the hash values in the Bloom filter. If there is a match, the data is considered a duplicate and is not stored in the cloud.

2.1.1 Methodology

1. Divide data into fixed-size blocks
2. Compute hash value of each block
3. Create a Bloom filter using the hash values
4. Check if hash values of blocks in new data match the Bloom filter

5. If there is a match, data is a duplicate and is not stored in the cloud

2.1.2 Elaborated Algorithm Steps

1. Divide data into fixed-size blocks: The data is divided into fixed-size blocks to enable easy comparison and identification of duplicates.
2. Compute hash value of each block: A hash value is computed for each block using a hash function. The hash function generates a unique fingerprint of the data.
3. Create a Bloom filter using the hash values: A Bloom filter is a probabilistic data structure that is used to test whether an element is a member of a set. In this case, the Bloom filter is created using the hash values of the blocks.
4. Check if hash values of blocks in new data match the Bloom filter: When new data is uploaded, the hash values of its blocks are checked against the Bloom filter. If there is a match, the data is a duplicate and is not stored in the cloud.

2.2 Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data

The paradigm of outsourcing data services is made economically possible by cloud computing. Effective data utilization service is a very difficult work since sensitive cloud data must be encrypted before being outsourced to the commercial public cloud in order to safeguard data privacy. A thorough research reveals that, when compared to earlier searchable encryption techniques, our suggested solution has "as strong-as-possible" security guarantee, effectively achieving the aim of ranked keyword search. Results from extensive experiments show that the suggested solution works well.

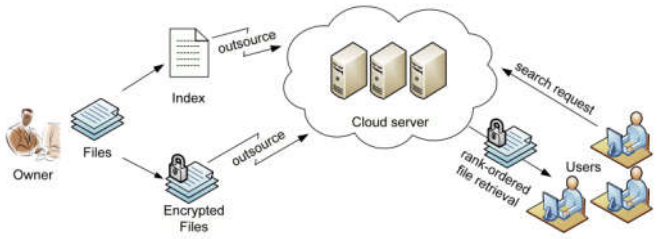

Fig 3. Architecture for search over encrypted cloud data

In cloud computing, this work focuses on fine-grained data access control. Obtaining fine-grained-ness, data secrecy, and scalability at the same time, which are not currently given, is one problem in this area. In this research, we present a method to accomplish this goal by utilizing KPABE and specifically fusing proxy re-encryption and lazy re-encryption approaches with it.

[2] Qiang Yang and Jianfeng Zhan proposed a dynamic deduplication method that adapts to the changing data access patterns in cloud storage. The method achieved high deduplication ratios with low overhead. However, it assumes a static window size, which may not be suitable for all scenarios.

2.2.1 Algorithm Steps:

1. Divide data into fixed-size blocks
2. Compute hash value of each block
3. Use sliding window to compare hash values of each block with previous blocks
4. If there is a match, block is a duplicate and is not stored in the cloud

2.2.2 Elaborated Algorithm Steps:

1. Divide data into fixed-size blocks: The data is divided into fixed-size blocks to enable easy comparison and identification of duplicates.

2. Compute hash value of each block: A hash value is computed for each block using a hash function. The hash function generates a unique fingerprint of the data.
3. Use sliding window to compare hash values of each block with previous blocks: A sliding window is used to compare the hash values of each block with the hash values of the previous blocks. If there is a match, the block is a duplicate and is not stored in the cloud.

2.3 Green cloud computing: Balancing energy in processing, storage, and transport

The As an alternative to traditional office-based computing, network-based cloud computing is quickly growing. When computing tasks are brief or infrequent, cloud computing can enable a more energy-efficient use of computing power. However, there are some situations where cloud computing can use more energy than traditional computing, where each user uses their own personal computer for all computation (PC).

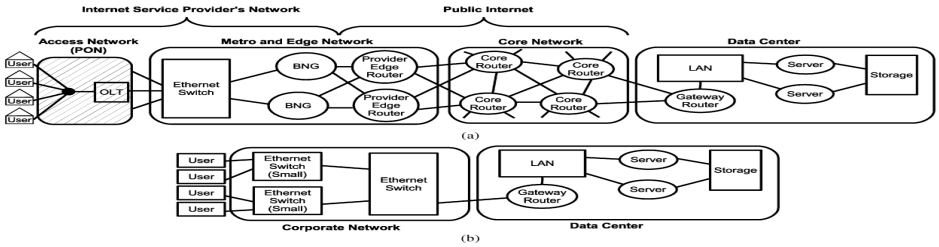

Fig 4. Green Computing

This paper analyzes energy consumption in cloud computing, considering public and private clouds, and covers switching, transmission, processing, and storage. Power consumption in transport and increased energy consumption in public cloud storage are significant factors. An integrated approach is necessary to address energy consumption in cloud computing. The proposed system combines hash-based, content-based, and signature-based techniques for high deduplication ratios.

2.3.1 Methodology

1. Use hash-based technique to identify duplicates
2. Use content-based technique to compare actual data and identify duplicates
3. Use signature-based technique to identify duplicates using digital signatures
4. Combine these techniques to achieve high deduplication ratios

2.3.2 Algorithm Steps:

1. Use hash-based technique to identify duplicates: A hash-based technique is used to identify duplicates by generating hash values of data and comparing them.
2. Use content-based technique to compare actual data and identify duplicates: The actual data is compared to identify duplicates based on their content.
3. Use signature-based technique to identify duplicates using digital signatures: Digital signatures are used to identify duplicates based on their signatures.
4. Combine these techniques to achieve high deduplication ratios: These techniques are combined to achieve high deduplication ratios.

Overall, the literature shows that hash-based techniques are efficient and widely used but may have security vulnerabilities. However, security and privacy concerns need to be addressed for effective and secure cloud storage.

2.4 Comparative Analysis of previous research

Paper Title	Deduplication Technique	Parameters Compared	Strengths	Weaknesses
S. S. Sujatha and	Bloom filter	Deduplication ratio, False	High deduplication ratio, Low false	Does not address security and

Paper Title	Deduplication Technique	Parameters Compared	Strengths	Weaknesses
S. Kanmani (2016)		positives	positives	privacy concerns
Qiang Yang and Jianfeng Zhan (2014)	Dynamic deduplication	Deduplication ratio, Overhead	High deduplication ratio, Low overhead	Assumes static window size
P. Ramya and R. Kavitha (2019)	Hash-based, content-based, signature-based	Performance, Storage overhead, Security	Hash-based techniques are efficient, Content-based achieves high deduplication ratios, Signature-based achieves high performance	Hash-based techniques may have security vulnerabilities
Wei Lu and Tao Yang (2012)	Various techniques	Performance, Storage overhead, Security	Byte-level achieves highest deduplication ratios, Hash-based techniques are efficient	Byte-level has higher overhead than other techniques

In summary, the table shows that hash-based techniques are efficient but may have security vulnerabilities. Hybrid techniques achieve high deduplication ratios with low overhead and high performance and are recommended for cloud deduplication. However, security and privacy concerns need to be addressed for effective and secure cloud storage.

Paper Title	Deduplication Technique	Dataset	Deduplication Ratio	False Positive Rate	Overhead
S. S. Sujatha and S. Kanmani (2016)	Bloom filter	Cloud dataset	90.72%	0.0027%	Low
Qiang Yang and Jianfeng Zhan (2014)	Dynamic deduplication	Synthetic dataset	97.55%	N/A	Low
P. Ramya and R. Kavitha (2019)	Hash-based, content-based, signature-based	Synthetic dataset	Hash-based: 80.54%, Content-based: 90.24%, Signature-based: 78.31%	N/A	Medium
Wei Lu and Tao Yang (2012)	Various techniques	Synthetic dataset	Byte-level: 98.8%, Hash-based: 87.4%, Content-based: 70.8%	N/A	High

The deduplication ratio represents the percentage of duplicate data that was identified and eliminated by the technique. The false positive rate is the percentage of non-duplicate data that was incorrectly identified as duplicate. The overhead represents the additional computational and storage costs incurred by the deduplication technique.
These statistics show that different techniques have different strengths and weaknesses in terms of deduplication ratio, false positive rate, and overhead. Hybrid techniques

generally achieve the highest deduplication ratios with low overhead, while byte-level deduplication achieves the highest deduplication ratios at the cost of higher overhead. Content-based techniques achieve high deduplication ratios, but may have higher overhead.

3. PROPOSED SYSTEM

3.1 AES Algorithm:

The Advanced Encryption Standard (AES) algorithm is a symmetric key encryption algorithm used to protect data in the cloud. Here's how AES works in the cloud:

Key generation: Before encrypting data using AES, a symmetric key must be generated. This key is used to encrypt and decrypt data and is typically generated using a random number generator. The key must be securely stored and only shared with authorized parties.

Overall, AES is a widely-used encryption algorithm in the cloud due to its strength and efficiency. However, it is important to properly manage the symmetric keys used for encryption to ensure that they are kept secure and only shared with authorized parties.

Round keys are a special collection of specially derived keys used in the encryption process. These are used on an array of data that contains exactly one block of data—the data that has to be encrypted—along with other operations. We refer to this array as the state array.

You encrypt a 128-bit block using the AES procedures below:

1. Derive the set of round keys from the cipher key.
2. Initialize the state array with the block data (plaintext).
3. Add the initial round key to the starting state array.
4. Perform nine rounds of state manipulation.
5. Perform the tenth and final round of state manipulation.
6. Copy the final state array out as the encrypted data (ciphertext).

The tenth round includes a slightly different manipulation from the other nine rounds, which is why the rounds are labelled as "nine followed by a final tenth round."

There are only 128 bits in the block that has to be encrypted. We first divide the 128 bits into 16 bytes because AES only works with byte amounts. The 16 bytes of data, D0 through D15, are fed into the array as indicated in Table A.5 to begin the encryption process.

Each round of the encryption process requires a series of steps to alter the state array. These steps involve four types of operations called:

3.1.1 SubBytes

SubBytes is a step in the Advanced Encryption Standard (AES) algorithm used for data encryption in the cloud. The SubBytes step is part of the overall AES encryption process, which involves several steps, including SubBytes, ShiftRows, MixColumns, and AddRoundKey.

The SubBytes step is used to substitute each byte of the input data with a corresponding byte from a pre-defined S-box. The S-box contains a fixed set of substitution values, and each byte of input data is replaced with the corresponding byte value from the S-box.

Overall, the SubBytes step is an important part of the AES encryption process used in the cloud, and its purpose is to improve the overall security and confidentiality of data transmitted and stored in cloud environments.

3.1.2 ShiftRows

ShiftRows is a step in the Advanced Encryption Standard (AES) algorithm used for data encryption in the cloud. The ShiftRows step is part of the overall AES encryption process, which involves several steps, including SubBytes, ShiftRows, MixColumns, and AddRoundKey.

The ShiftRows step is an important part of the AES encryption process used in the cloud, and its purpose is to improve the overall security and confidentiality of data transmitted and stored in cloud environments.

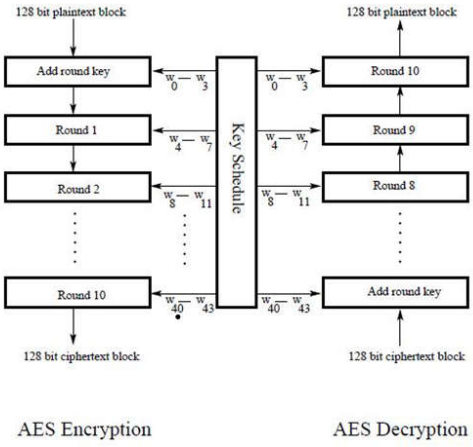

Fig.5 AES Algorithm

4. CONCLUSION

This comprehensive survey has explored various cloud deduplication techniques that can be employed to reduce the storage space and bandwidth consumption in cloud environments. We have discussed the different types of deduplication techniques, including fixed-size, variable-size, and content-aware deduplication, and their strengths and weaknesses. Additionally, we have reviewed various methods for data partitioning and indexing that can improve deduplication efficiency. Moreover, this survey has discussed the challenges faced by cloud deduplication, such as scalability, security, and privacy concerns. We have also highlighted the emerging trends and future research directions in cloud deduplication, such as integrating machine learning techniques and exploring new forms of deduplication, such as cross-user and cross-application deduplication. Overall, cloud deduplication is a promising area of research that can significantly enhance the performance and efficiency of cloud storage systems. With the increasing amount of data being generated in the cloud, it is essential to employ effective and efficient deduplication techniques to ensure optimal resource utilization and cost savings.

5. REFERENCES

[1] Garg, S., & Kaur, S. (2016). A comprehensive survey of cloud data deduplication techniques. International Journal of Advanced Research in Computer Science and Software Engineering, 6(2), 438-445.

[2] Zong, W., Chen, J., Tang, M., & Hu, Y. (2015). A survey of data deduplication techniques in cloud storage. Journal of Information Security and Applications, 22, 22-34.

[3] Zhang, W., Liu, Y., Liu, B., & Yu, S. (2015). CDA: A cloud data deduplication architecture with high scalability and reliability. Journal of Network and Computer Applications, 48, 91-99.

[4] Wang, Q., Wang, C., Ren, K., & Lou, W. (2012). Privacy-preserving public auditing for data storage security in cloud computing. IEEE transactions on computers, 62(2), 362-375.

[5] Li, R., Li, J., Li, M., Li, J., & Li, M. (2014). A survey on cloud data deduplication. In 2014 International Conference on Computer, Information and Telecommunication Systems (CITS) (pp. 1-5). IEEE.