ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 9, Issue 1 , January 2022

Attribute-Based Storage Supporting Secure Deduplication

Ajay S. Chavan, Rushikesh P. Kale, Sujit Ghute, Nikhil Jagdale

Bachelor of Engineering Student, Department of Computer Engineering, JSPM Bhivarabai Sawant Institute of Technology & Research, Pune, Maharashtra, India
Bachelor of Engineering Student, Department of Computer Engineering, JSPM Bhivarabai Sawant Institute of Technology & Research, Pune, Maharashtra, India
Bachelor of Engineering Student, Department of Computer Engineering, JSPM Bhivarabai Sawant Institute of Technology & Research, Pune, Maharashtra, India
Bachelor of Engineering Student, Department of Computer Engineering, JSPM Bhivarabai Sawant Institute of Technology & Research, Pune, Maharashtra, India

ABSTRACT: Tenants in SaaS typically customize several duplicates and place them on separate data nodes of the service provider to ensure data dependability in the cloud. However, service providers who are untrustworthy may exist. Tamper with, destroy, or forge tenant data on the other hand, since the untrustworthy service provider's duplication appears to be identical. Could only save one data copy rather than the required number to deceive tenants As a result, tenants must guarantee that the service is available. supplier processes their data duplicated honestly, which is not being tampered with or partially deleted This paper describes a tenant duplication integrity protection mechanism. TDIC (Tenant-oriented Duplication Integrity Checking Scheme). TDIC is an example of tuples. Based challenge-response approach and creates a new tenant duplication authentication structure (TDAS) based on tenant physical tuples. TDIC, when combined with TDAS and a homomorphism label, allows data to be collected. Verification of duplication in the absence of local copies on a regular basis TDIC minimizes the complexity of service providers by random sampling. side verification of object formation and elimination of communication consumption waste.

I.INTRODUCTION

Tenants' data is stored and processed at remote service providers in SaaS, which is based on the single instance multitenancy method. Meanwhile, cloud tenants can customise many duplications and pay for use to ensure data reliability. However, service providers may be untrustworthy, and they may tamper with, erase, or falsify tenants' data. Plain-text data duplication, on the other hand, is vulnerable to service provider conspired attacks; because all replicas seem the identical, an untrustworthy service provider may retain only one data copy rather than the specified number to defraud renters. For the top issues, renters must guarantee that the service provider processes their data duplication honestly and that it is not manipulated with or partially destroyed. At the same time, because renters no longer have local copies of their data, they should be equipped with particular security measures that allow them to verify the correctness of the remote data even in the absence of local copies. Existing research focuses on the scenario in which users own an independent data storage mode rather than a shared one for secure data storage. However, in SaaS, most multitenant applications adopt the single instance multi-tenancy technique to make full use of the resource, which results in numerous tenants' data being kept in a single data table, such as a universal table. The old solution that required obtaining file partitions [1,4,10] would not perform effectively in this circumstance on the shared physical storage mode in SaaS. Because there may be data from multiple tenants in a single data block partition, and this partition not only violates the data isolation requirement of distinct tenants but also complicates integrity verification.

II. LITERATURE SURVEY

Shafi Goldwasse,Silvio Micali,Charles Rackoff, ” The Knowledge Complexity of Interactive Proof-Systems.” [1] We offer a novel theorem-proving procedure, which is a new efficient method of communicating a proof, in the first half of the paper: Any approach of this nature presupposes a definition of proof, either directly or indirectly. Our "proofs" are based on probability. We may erroneously be convinced of the truth of an II-bits long statement with a very little probability, say, if we enter it. We propose categorising languages based on the amount of additional knowledge required to demonstrate membership. We show that it is possible to interactively verify that a number is quadratic non-residue

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 9, Issue 1 , January 2022

mod m while releasing 0 additional knowledge in the case where this additional knowledge is essentially zero. This is surprising, because when m's factorization is unknown, there is no efficient solution for determining quadratic rcsiduosity mod m. Furthermore, the prime factorization of tn appears in all known NP proofs for this problem. This suggests that including interaction into the proving process may reduce the quantity of information that needs to be shared in order to prove a theorem.

Darren Quick , Kim-Kwang Raymond Choo," :Google Drive: Forensic analysis of data remnants." [2] Cloud storage is posing a new challenge to digital forensic investigators. Consumers, businesses, and governments are increasingly utilising the services, which have the ability to store massive amounts of data. Due to virtualization, a lack of understanding about the location of digital evidence, privacy concerns, and legal or jurisdictional limits, retrieving digital evidence from cloud storage services (especially from overseas providers) can be difficult in a digital forensic inquiry. Google Drive is a popular service that allows users to access, save, share, and disseminate material at a low cost, and in certain circumstances for free. Using Google Drive as a case study, artefacts were detected on a computer hard drive and an Apple iPhone3G that are likely to persist after the use of cloud storage in the context of the experiments, as well as the potential access point(s) for digital forensics examiners to protect evidence.

Yanjiang Yang a, Haiyan Zhub, Haibing Luc, Jian Weng d, Youcheng Zhang e,Kim-Kwang Raymond Choof." Cloud based data sharing with fine-grained proxy re-encryption" [3] Conditional proxy re-encryption (CPRE) allows for fine-grained decryption privileges delegation and has a variety of real-world uses. In this research, we provide a ciphertext-policy-attribute-based CPRE scheme, as well as a formalisation and security analysis of the primitive. We show how the method works in a cloud deployment, where finegrained data exchange is possible. This application implements cloud server-enabled user revocation, providing a more efficient answer to the user revocation problem in the context of fine-grained cloud data encryption. Another notable characteristic of the application is its high user-side efficiency, which allows users to access cloud data using resourceconstrained devices such as smartphones. The performance of the proposed strategy appears to be promising based on our evaluations.

Benjamin Zhu,Kai Li, "Avoiding the Disk Bottleneck in the Data Domain Deduplication File System. "[4] Disk-based deduplication storage has emerged as the next-generation enterprise data protection storage technology to take the role of tape libraries. Deduplication eliminates duplicate data segments, compresses data to an extremely compact format, and enables the storage of backups on disc rather than tape. High throughput, often greater than 100 MB/sec, is a critical need for enterprise data protection, as it enables backups to be completed quickly. A significant problem is identifying and removing duplicate data segments at this pace on a low-cost system that lacks the RAM necessary to retain an index of the stored segments and may be compelled to seek an on-disk index for each incoming segment. This article outlines three strategies used to alleviate the disc bottleneck in the production Data Domain deduplication file system. These techniques include (1) the Summary Vector, a compact in-memory data structure for identifying new segments; (2) Stream-Informed Segment Layout, a data layout technique for improving on-disk locality for sequentially accessed segments; and (3) Locality Preserved Caching, which preserves the locality of duplicate segment fingerprints to achieve high cache hit ratios. They can eliminate 99accesses for deduplication of real-world workloads when used in conjunction. These strategies enable a modern two-socket dual-core machine to operate at multistream throughput rates of 90 and 210 MB/sec.

Hugo Patterson., "Cloud Cryptography: Theory, Practice and Future Research Directions" [5] Cloud computing, a convenient method of accessing services, resources, and applications via the Internet, shifts the focus of industries and organisations away from the deployment and day-to-day operation of their IT facilities by offering an on-demand, self-service, and pay-as-you-go business model. As a result, it's no surprise that cloud computing has grown in popularity in recent years. While cloud computing offers numerous advantages to users, it also poses security and privacy concerns. Multitenancy, resource pooling, and shareability capabilities, for example, can be abused by cybercriminals and others with malicious intent, to the cost of both cloud customers and cloud service providers. It's no surprise, then, that cloud computing has emerged as a hot topic for security researchers to investigate. For example, when user data (e.g., documents, videos, and photos) is uploaded or stored in a cloud computing service, the data owners are unlikely to be aware of the path of the transmitted data or whether the data is being collected and analysed by a third party, including a government agency (see Edward Snowden's revelations [8]). As Choo and Sarre [6] argue, it is critical to strike a balance between privacy, legitimate surveillance, and lawful data access in order to ensure that the privacy of innocent people is

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 9, Issue 1 , January 2022

not jeopardised (e.g., that fine-grained aspects of an individual's life cannot be derived or inferred from intelligence collection and analysis).

III. PROBLEM STATEMENT

The challenge is to figure out how to develop safe deduplication systems that are more reliable in cloud computing. As a result, it has been proposed to incorporate deduplication technologies into distributed cloud storage servers to improve fault tolerance. The secret sharing mechanism, which is also compatible with distributed storage systems, is used to secure data confidentiality. A brief cryptographic hash value of the content will also be computed and delivered to each storage server as the fingerprint of the fragment saved at each server to allow deduplication.

SYSTEM ARCHITECTURE

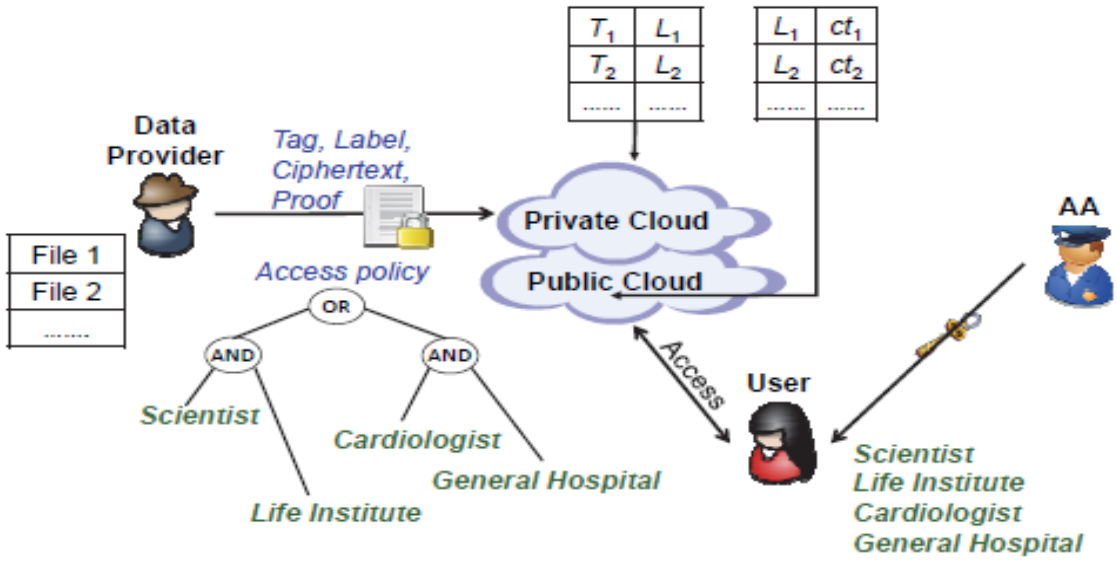

Fig. System Architecture

IV. ALGORITHM

Advanced Encryption Standard

To generate ciphertext, the AES algorithm employs a substitution-permutation (SP) network with many rounds. The number of rounds is determined on the key size. A 128-bit key size requires ten rounds, a 192-bit key size requires 12 rounds, and a 256-bit key size requires 14 rounds. Each of these rounds requires a round key, but because the method only accepts one key, this key must be expanded to obtain keys for all rounds, including round 0.Symmetric key symmetric block cipher

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 9, Issue 1 , January 2022

- 128-bit data, 128/192/256-bit keys
- Stronger and faster than Triple-DES
- Provide full specification and design details
- Software implementable in C and Java

Data Encryption Standard

The Data Encryption Standard (DES) is a block cypher technique that takes plain text in 64-bit blocks and turns it to ciphertext using 48-bit keys. It is a symmetric key algorithm, which means that the same key is used to encrypt and decrypt data.

- In the first step, the 64-bit plain text block is handed over to an initial Permutation (IP) function.
- The initial permutation is performed on plain text.
- Next, the initial permutation (IP) produces two halves of the permuted block; says Left Plain Text (LPT) and Right Plain Text (RPT).
- Now each LPT and RPT go through 16 rounds of the encryption process.
- In the end, LPT and RPT are rejoined and a Final Permutation (FP) is performed on the combined block
- The result of this process produces 64-bit ciphertext.

V. CONCLUSION

We investigated the topic of providing secure outsourced storage that allows deduplication while also resisting bruteforce attacks. DupLESS is a system that combines a CE-type base MLE scheme with the ability to receive messagederived keys via a key server (KS) shared by a group of clients. Clients communicate with the KS using a protocol for oblivious PRFs, ensuring that the KS can cryptographically mix in secret material to per-message keys while learning nothing about files saved by clients. These procedures ensure that DupLESS provides robust security against external assaults that compromise the SS and communication channels (nothing is disclosed beyond file lengths, equality, and access patterns), and that the security of DupLESS gently degrades in the face of compromised systems. If a client is compromised, knowing the plaintext underlying another client's cipher text requires conducting online brute force attacks (which can be slowed by a rate-limited KS). If the KS is compromised, the attacker must still conduct an offline bruteforce assault, which matches the assurances of standard MLE methods. The significant boost in security comes at a low cost in terms of performance and a little increase in storage requirements over the base system. The low performance overhead is due in part to optimizing the client-to-KS OPRF protocol, as well as ensuring DupLESS uses a low amount of interactions with the SS. We demonstrate that DupLESS is straightforward to deploy: it can work transparently on top of any SS that implements a simple storage interface, as demonstrated by our prototypes for Drop box and Google Drive.

VI. REFERENCES

[1] Bitcasa, inifinite storage. http://www.bitcasa.com/.
[2] Ciphertite data backup. http://www.ciphertite. com/.
[3] Dropbox, a file-storage and sharing service. http://www. dropbox.com/.
[4] Dupless source code. http://cseweb.ucsd.edu/ users/skeelvee/dupless.
[5] The Flud backup system. http://flud.org/wiki/ Architecture.
[6] GNUnet, a framework for secure peer-to-peer networking. https://gnunet.org/.
[7] Google Drive. http://drive.google.com.
[8] ADYA, A., BOLOSKY, W., CASTRO, M., CERMAK, G., CHAIKEN, R., DOUCEUR, J., HOWELL, J., LORCH, J., THEIMER, M., AND WATTENHOFER, R. Farsite: ACM SIGOPS Operating Systems Review 36, Federated, available, and reliable storage for an incompletely trusted environment., SI (2002), 1–14.
[9] AMAZON. Amazon Elastic Block Store (EBS). http:// aws.amazon.com/ebs.
[10] AMAZON. Amazon Elastic Compute Cloud (EC2). http:// aws.amazon.com/ec2.
[11] AMAZON. Amazon Simple Storage Service (Amazon S3). http://aws.amazon.com/s3.
[12] ANDERSON, P., AND ZHANG, L. Fast and secure laptop backups with encrypted de-duplication. In Proc. of USENIX LISA (2010), 1-10.