ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

DE-DUPLICATION WITH ENHANCED RELIABILITY AND SECURE AUDITING IN CLOUD

Vinayak Borwal*, Akshay Janrao
Computer Science, AISSM's Institute of Information Technology, India
Computer Science, AISSM's Institute of Information Technology, India

DOI: 10.5281/zenodo.48320

ABSTRACT

As the distributed computing innovation creates amid the most recent decade, outsourcing information to cloud administration for capacity turns into an alluring pattern, which benefits in saving endeavors on overwhelming information upkeep and administration, In any case, subsequent to the outsourced distributed storage is not completely reliable, it raises security worries on the best way to acknowledge information de-duplication in cloud while accomplishing honesty inspecting. In this work, we concentrate on the issue of respectability reviewing and secure de-duplication on cloud information. Specifically, going for accomplishing both information respectability and de-duplication in cloud, we propose two protected frameworks, in particular Sec-Cloud and Sec-Cloud+. Sec-Cloud presents a reviewing substance with upkeep of a Map Reduce cloud, which helps customers produce information labels before transferring and in addition review the honesty of information having been put away in cloud. Contrasted and past work, the calculation by client in Sec-Cloud is incredibly decreased amid the file transferring and examining stages. Sec-Cloud+ is planned persuaded by the way that clients constantly need to scramble their information before transferring, and empowers honesty evaluating and secure de-duplication on encoded information. Energy consumption and internet bandwidth reduced by using de-duplication because it can reduce large amount of data and also speed of data de-duplication. By eliminating the duplicate data we can serve internet bandwidth and also cost of storage gets reduced, improve the quality of searching and also helps to reduce the heavy load on the remote server. Hash value is going to use for each value then it is checked whether there is redundant value is present for the files which are uploaded on to the cloud. There are two types of duplication first is file level, in file level duplication hash value of two files is checked and if value is same that means content present in the files are same and only one file will get stored on to the cloud and next is block level, file is divided into multiple blocks and hash value of each block is checked for examining de-duplication.

KEYWORDS: Reliability, Secure De-duplication, Integrity Auditing, Distributed File System, Cloud Storage.

INTRODUCTION

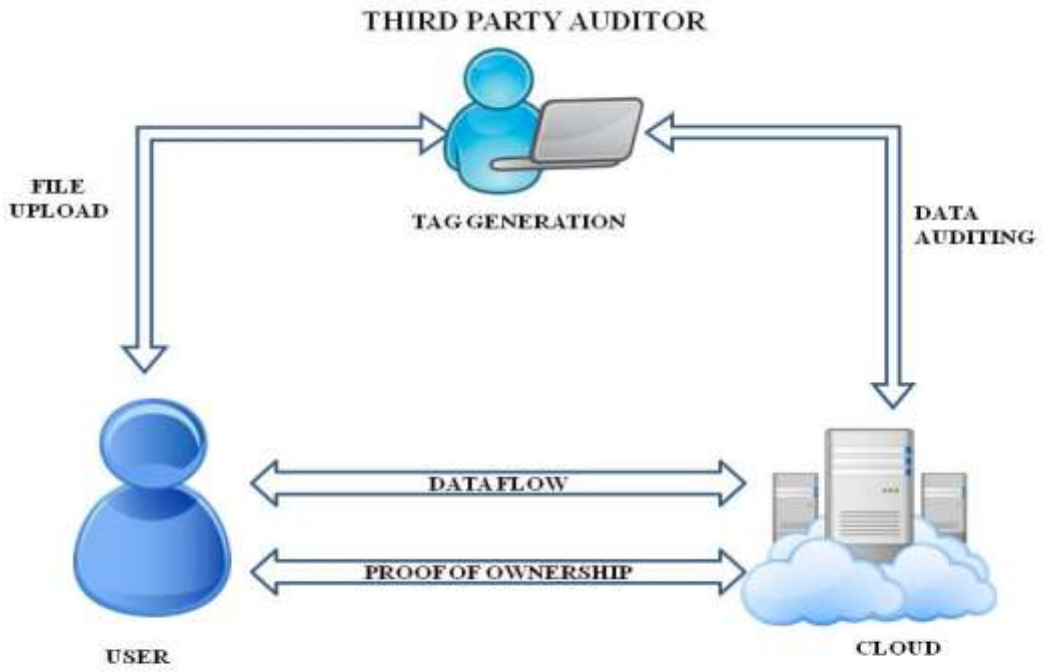
Distributed storage is a model of organized venture stockpiling where information is put away in virtualized pools of capacity which are by and large facilitated by third gatherings. Distributed storage gives customers benefits, going from expense sparing and simplified comfort, to versatility opportunities and adaptable administration. These extraordinary components pull in more clients to use and capacity their own information to the distributed storage: as per the examination report, the volume of information in cloud is required to accomplish 40 trillion gigabytes in 2020. Despite the fact that distributed storage framework has been broadly embraced, it neglects to oblige some vital rising needs, for example, the capacities of examining respectability of cloud files by cloud customers and identifying copied files by cloud servers. We outline both issues beneath. The first issue is respectability examining. The cloud server has the capacity diminish customers from the overwhelming weight of capacity administration and support. The most contrast of distributed storage from conventional in-house stockpiling is that the information is exchanged by means of Internet and put away in an unverifiable space, not under control of the customers by any stretch of the imagination, which unavoidably raises customer's extraordinary worries on the trustworthiness of their information. Cloud storage is huge and available for use in internet for public accessible storage called as DaaS (Data Storage as a Service). Cloud backup and restoration is slower and cost is more because bandwidth of internet

is lower than local area network it is the main challenge is cloud backup service. In private cloud space hardware resources are limited, there is need of data utilization optimally so it can accommodate maximum data. In public cloud it is very easy to set up and cost of set up is minimum and it meet to the scalability needs. It may be free service and charge service as per usage of storage space. In the comparison of block level, data may be fixed sized or variable sized, the process of dividing data into bytes is called chunking and data block is called chunk. In the fixed size data chunking size of data bytes may be 4kb, 8kb, 16kb, 32kb.In variable size data blocks may present in variable size and it is better than fixed size because in the fixed size memory gets wasted and in variable size block it will occupy only needed space. There are various algorithms were used to design system which satisfy the user requirement. Here we are going to use SHA-1 data tag generation, RSA for encrypt and decrypt data. It has a advantage that it makes use of only one key rather than generating two key for encryption and decryption used in AES algorithm. By using this integrity and reliability is maintained between the data.

PROPOSED SYSTEM

We determine that our proposed Sec-Cloud framework has accomplished both integrity auditing and file de-duplication. Be that as it may, it can't keep the cloud servers from knowing the substance of files having been put away. In other words, the functionalities of integrity auditing and secure de-duplication are just forced on plain files. In this area, we propose Sec-Cloud+, which takes into account integrity auditing and de-duplication on scrambled files. Framework Model Compared with Sec-Cloud, our proposed Sec-Cloud+ involves an extra trusted element, to be specific key server, which is in charge of assigning customers with mystery key (according to the file content) for encrypting files. This construction modeling is in line with the late work. However, our work is distinguished with the past work by allowing for integrity auditing on encoded data. Sec-Cloud+ takes after the same three protocols (i.e., the file uploading protocol, the integrity auditing protocol and the proof of ownership protocol) as with Sec-Cloud. The main distinction is the file uploading protocol in Sec-Cloud+ involves an extra stage for correspondence between cloud customer and key server. That is, the customer needs to speak with the key server to get the merged key for encrypting the uploading file before the phase in Sec-Cloud.

Figure:



Architecture

There are two important concepts in Sec-Cloud+.

1. Data De-duplication
2. Integrity Auditing

Data de-duplication is the unique technique for the redundant operations like backup, which stores repeatedly copying and storing the same data block for multiple times. Elimination of redundant data and replaced by a pointer to the unique data block. Eliminating redundant data can significantly shrink data requirements and lower storage costs. De-duplication also improve bandwidth efficiency and also save processing power.

Types of de-duplication:

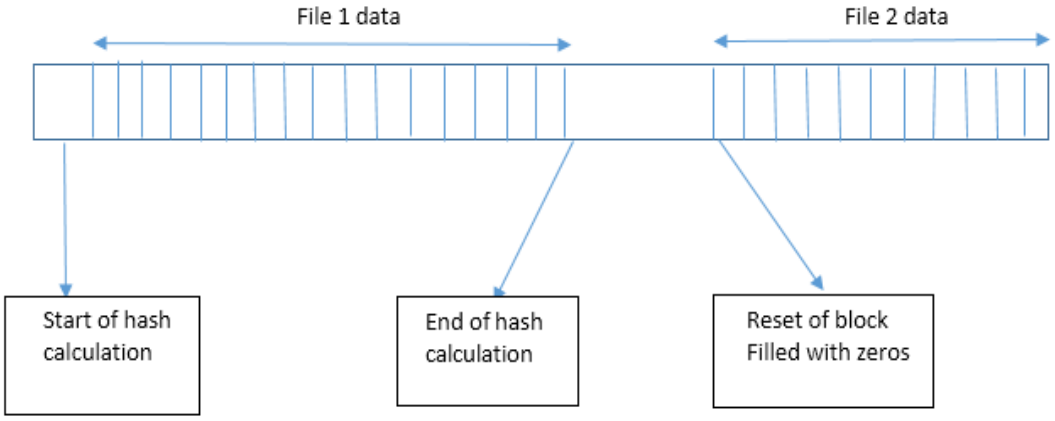
File Level de-duplication

A file is a data content file when examining the data of duplication, and it is basically uses the hash value of the file as an identifier. If two or more files have the same hash value, they are assumed to have the same data content and only one of these files will be stored. When every new file is transferred from the user we just need to check the hash value first in order to make sure only unique data file is being stored.

Block level de-duplication

In this strategy, each file is divided into number of blocks with flexible size based on the size of data file. For each block computes a hash value for examing the duplicated blocks of that particular data file. For computing the hash value SHA -1 is used to compute values for particular block. After doing all types of de-duplication the data is transferred to the cloud server and tag generated by the algorithm is also transferred to the cloud. Suppose, when a file is transferred from the user, the particular data file is first encrypted at the user end and then it transferred to the third party auditor which is act as a Tag generator. Based on the SHA-1 algorithm, the tag is generated for each block of the data file or for each of the file. The file and tag of the file is transferred to the cloud server. Suppose the tag is already in the cloud in that case the tag and file is cannot get transferred to the cloud server and POW (Proof of Ownership) protocol is given to the user by the cloud server.

Figure:



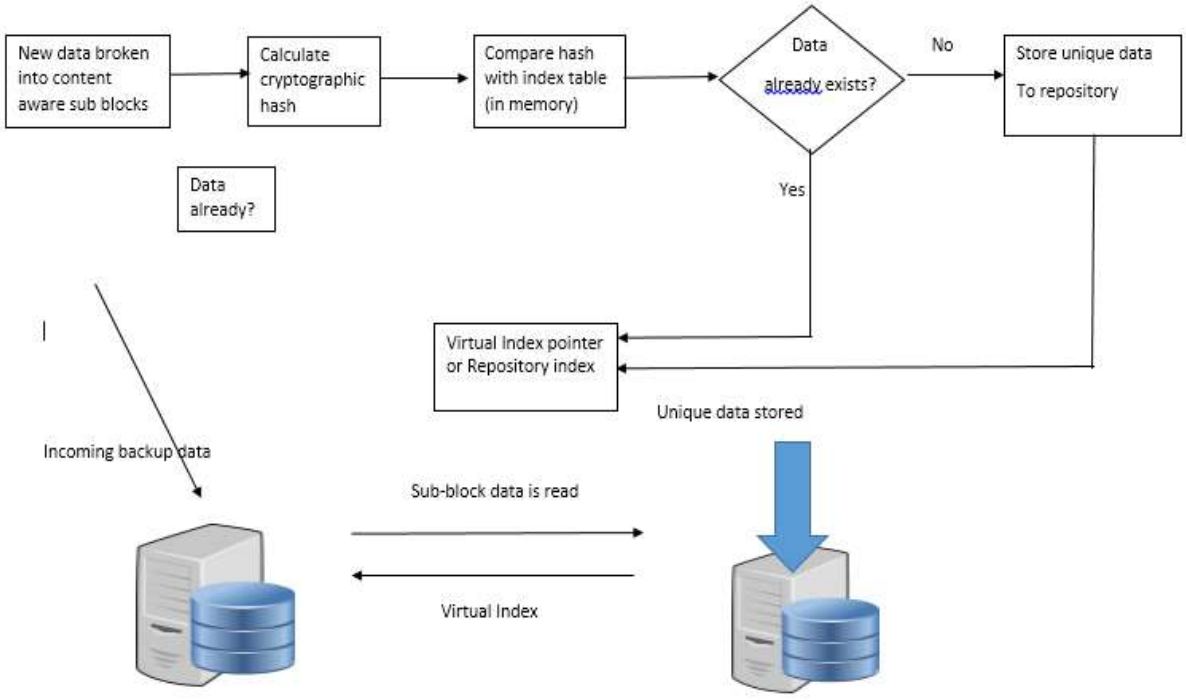
SHA-1

SHA – 1 is a cryptographic hash function designed by the National Security Agency (NSA).SHA – 1 is the most widely used of the SHA hash function and so efforts are underway to develop improved alternatives. SHA-1 produces a 160-bit message based on principles and those used by Ronald L. Rivest of MIT in the design of the MD4 and MD5 message digest algorithms. Due to the block and structure of the algorithms and the absence of additional steps, all SHA functions are vulnerable to length- extension and partial- message collision attacks. Keyed hash – SHA (message || key) or SHA (key || message) and recalculating the hash without knowing the key. The simplest improvement to avoid the attacks is to hash the key twice- SHA' (message) = SHA (SHA(0'|| message)) (0' – zero block, length is equal to size of hash function.)

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

INTEGRITY AUDITING

For assuring the definition of provable data possession (PDP) the cloud servers possesses the targeted files without retrieving or downloading the whole data. PDP is a proof protocol for sampling a particular set of blocks and asking the server to prove that the existing block possesses the blocks in the cloud server and the verifier only maintains the metadata for performing the integrity checking. Another proof protocol is Proof of Retrievability (POR) protocol which assures the cloud servers possesses the target files. But it also guarantees for the retrievability or full recovery of the particular data file. The improved POR model is constructed by manipulating the classic Merkle hash tree construction for block tag authentication and proper retrievability. The third party auditor can also act as a verifier. It transferred or executes a challenge call to the cloud server for checking the particular is stored or not at a particular nodes.



RSA

RSA is an asymmetric algorithm used by to encrypt and decrypt messages. Asymmetric indicates or explained that there are two different keys. This is also called public key cryptography, because one of them can be given to everyone. The other key must be kept private. It is basically based on the fact that finding the factors of an integer is hard .RSA stands for Ron Rivest Adi Shamir and Leonard Adleman, who first described it in 1978. A user of RSA creates and then publishes the product of two large prime numbers, along with an auxiliary or hash value, as their public key. The prime factors must be kept secret. Anyone can use the public key to encrypt a message. If the public key is large enough, only with detailed knowledge of the prime factors can decode the message.

The RSA algorithm are as follows:

Choose two different large random prime numbers p and q.

Calculate $n = pq$ n is the modulus for the public key and the private keys

Calculate $\phi(n) = (p-1)(q-1)$

Choose an integer e such that $1 < e < \phi(n)$ and e is co-prime to $\phi(n)$ **i.e.** e and $\phi(n)$ share no factors other than 1; gcd $(e,\phi(n))$ =

e is released as the public key exponent.

Compute d to satisfy the congruence relation $de \equiv 1 \pmod{\phi(n)}$ **i.e.** $de = 1 + k\phi(n)$ for some integer k

d is kept as the private key exponent.

SUMMARY AND CONCLUSION

Duplicate data files can be removed and by using proof of ownership access for particular file will be given to the user. Hence reliability is maintained throughout the system. By using file level duplication and block level duplication data will be uploaded to the cloud. Data can be retrieved at faster rate and backup of data can be maintained by using distributed server. Secure+ cloud is used for forwarding data in encrypted for security and can be retrieved to the end user in original format by using decryption. De-duplication should effectively provide high deduce ratio with high throughput.it is efficient technique to optimize storage of data. Multiple user can share the same file and can be retrieved files successfully.

Thus we have conclude that data is securely retrieved by the user and cloud space and be utilize efficiently. De-duplication can be checked amongst the data and only one copy of data can be uploaded to the cloud if the data file is same. Data file gets encrypted before uploading to the cloud by using this, security is maintained. Data de-duplication provide high data transfer, lower data cost and higher disk IO rate. Application helps in maintenance of file data on the cloud server so that no duplicate files are stored in the cloud. Currently this technique for storage files has been tested for text format files only. In near future, it can be further extended to support other format files.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Communication of the ACM, vol. 53, no. 4, pp. 50–58, 2010.

[2] J. Yuan and S. Yu, "Secure and constant cost public cloud storage auditing with deduplication," in IEEE Conference on Communications and Network Security (CNS), 2013, pp. 145–153

[3] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in Proceedings of the 18th ACM Conference on Computer and Communications Security. ACM, 2011, pp.491–500.

[4] S. Keelveedhi, M. Bellare, and T. Ristenpart, "Dupless: Server- aided encryption for deduplicated storage," in Proceedings of the 22Nd USENIX Conference on Security, ser. SEC'13. Washington, D.C.: USENIX Association,2013,pp.179194.[Online].Available:https://www.usenix.org/conference/usenixsecurity13/technicalsessions/presentation/bellare

[5] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proceed- ings of the 14th ACM Conference on Computer and Communications Security, ser. CCS '07. New York, NY, USA: ACM, 2007, pp. 598– 609.

[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson, and D. Song, "Remote data checking using provable data possession," ACM Trans. Inf. Syst. Secur., vol. 14, no. 1, pp. 12:1–12:34, 2011.

[7] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks, ser. SecureComm '08. New York, NY, USA: ACM, 2008, pp. 9:1–9:10.

[8] C. Erway, A. K¨upc¸¨u, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in Proceedings of the 16th ACM Conference on Computer and Communications Security, ser. CCS '09. New York, NY, USA: ACM, 2009, pp. 213–222.

[9] F. Seb´e, J. Domingo-Ferrer, A. Martinez-Balleste, Y. Deswarte, and J.-J. Quisquater, "Efficient remote data possession checking in critical information infrastructures," IEEE Trans. on Knowl. And Data Eng., vol. 20, no. 8, pp. 1034–1038, 2008.

[10] H. Wang, "Proxy provable data possession in public clouds," IEEE Transactions on Services Computing, vol. 6, no. 4, pp. 551–559, 2013.

[11] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," IEEE Trans- actions on Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231–2244, 2012.

[12] H. Shacham and B. Waters, "Compact proofs of retrievability," in Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology, ser. ASIACRYPT '08. Springer Berlin Heidelberg, 2008, pp. 90–107.