ISSN:0975-9646

Result on Storage and Security Management on Private Cloud

Shraddha S. Sadavarte [#1]

ME CSE (Student), MSS's CET, Jalna, Dr. BAM University

Abstract—private cloud used for storage of important data of any organization we need to manage space of private cloud so deduplication technique is mostly used. In this technique for faster retrieval of a file we need to maintain chunk id of a file and metadata so we used indexing technique which helps in faster retrieval of data. And for complete security of cloud we provide security to cloud so cloud becomes perfectly manages and secured by hackers.

Index Terms: indexing, Security, Deduplication, Symmetric, Asymmetric, Public Key, Private Key.

I. INTRODUCTION

Private cloud is set up for any organization in that storage is storage is set up with the storage resources of unused commodity machines within the organization. This storage is delivered as service to users so storage space is limited and we need to use it well .so deduplication [3] technique is used in this technique file is splited in chunks and by comparing chunks of different files duplicates are calculated in this we need to maintain chunk id of each chunk & chunk index also need to maintain metadata of file for achieving faster retrieval of a data and improves search time.

In private cloud is secured still security issue may occur by the inner members of organization [8]. There may risk of data hacking by the employee of an organization in private cloud there is issue of securing data from other organization so for solving storage space issue and security management we have to work on different security mechanism and indexing technique.

II LITERATURE REVIEW

Chun-Ho Ng' work Live DFS [14] carried out deduplication of VM image in the Open-Source cloud. Live DFS saves 40% of storage space. It has limited scalability.

SAM [15] is a semantic-aware multi-tiered source de-duplication framework .it is time consuming. SAM proposed to combine global file-level deduplication with local chunk-level deduplication method

siLo[9], gives both locality and similarity in backup streams which helps to achieve a near-exact deduplication but at a RAM cost that is much lower than locality-only or similarity only based methods. It addresses only intra-node challenge of single deduplication server.

EMC Data Domains global deduplication Array [10], IBM's ProtecTier [11], and Sepation's S2100-ES2 [12], these are good in small clusters. But fails at thousands of nodes in cloud datacenters due to shortcomings in terms of cluster-wide deduplication ratio,

HYDRAstor [13] performs deduplication at a large-chunk size without data sharing among the nodes, and distributes data at the chunk level using distributed hash table (DHT).

III SYSTEM DEVELOPMENT

In private cloud we work on deduplication technique for efficient data storage & in this technique for fast retrieval of data we used indexing technique [1] and provide security to cloud by using following method and algorithms.

For finding duplicates we used extreme binning[4] .In that hierichacial index is used which is divided in two parts its primary index holds representative Chunk IDs Secondary index holds the unique chunks of files. It gives faster identification of duplicates and solve system overhead problem.

Scalable hybrid hash cluster used to store chunk entries in SSD [2] it avoids disk I/O bottleneck problem.

Efficient B+ tree indexing [5]. It maintains a large number of chunk IDs & gives faster retrieval of data and speed up the searching.so it reduces search time.

SHHC which uses sophisticated device SSD to maintain the chunk index. Because optimized private cloud storage is built using commodity machines, so it's not possible to maintain index in such devices [2].

RSA Algorithm is used for providing communication security to private cloud .In this client sends its data on cloud on encrypted form and it is decrypted only on cloud so no one can hack the communication between client & cloud. [6]

MD5 Algorithm is used for checking the integrity & correctness of a data by calculating checksum of chunk and & hole file It checks that the correct file is stored on cloud or not. In retrieval process of a file client got correct file from cloud by checking each chunk checksum with its previous checksum So user got correct file from cloud [6].

AES Algorithm is used for providing security by using encryption technique to most critical data of client. So no one can be able to hack the content of a file [6]

IV. RESULTS

A. Space complexity

This graph is calculated on basis of utilization of storage space before performing dedupliaction technique it requires more storage space and after performing deduplication technique the storage space required is less

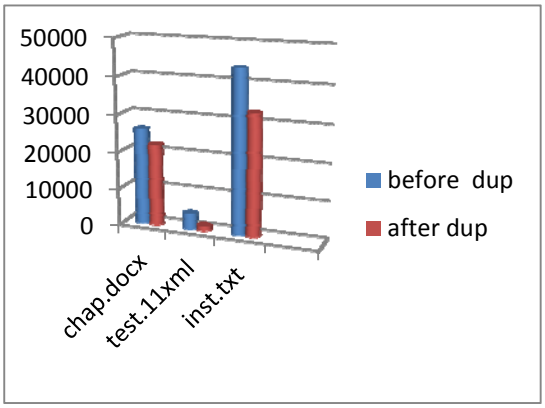

Fig 1. Storage space utilization before deduplication technique and after deduplication technique

So our propose system helps in utilization of storage space as compare to other

- Before deduplication- space required at the time of simple storage without use of deduplication technique
- After deduplication- space required by using deduplication technique.
- Graph shows that our propose system performs better than existing before deduplication technique
- Graph chap.docx.test.xml,inst.txt are different file name each have different size and in this deduplication is performs so we get reduction in storage space after deduplicaton of data

B. ***Throughput***

1) AES Algorithm: We calculate throughput and encryption performance of an algorithm according to different types of files with different size. Throughput is compared with encryption time, maximum throughput shows better performance of an algorithm our graph shows our results performs better as compared to others because algorithm gives higher throughput

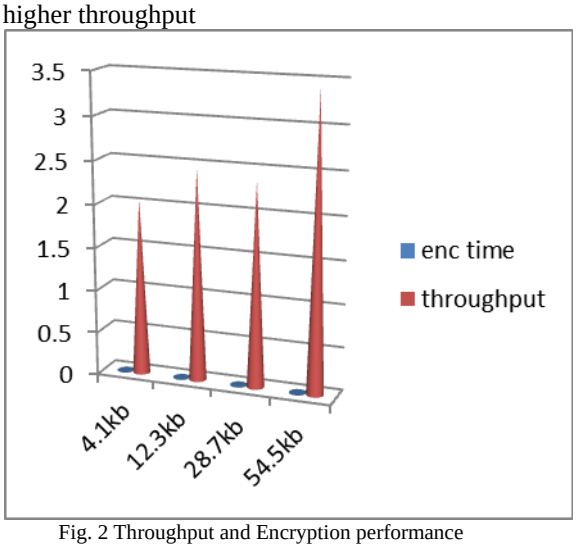

Fig. 2 Throughput and Encryption performance

2). RSA Algorithm: We calculate throughput and encryption performance of an algorithm according to different types of files with different size. Throughput is compared with encryption time, maximum throughput shows better performance of an algorithm our graph shows our results performs better as compared to others because algorithm gives higher throughput

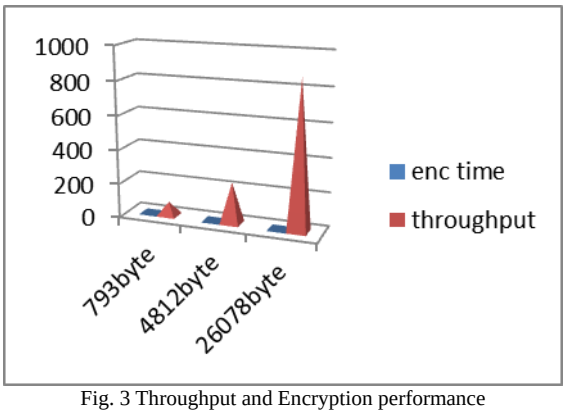

Fig. 3 Throughput and Encryption performance

C. ***Time Complaxcity***

TABLE I Performance analysis of B+ tree

This helps in organization of chunks assemble start time for storage of file assemble end time is time required for retrieval of a file

Sr. No	File size	Assemble start time(MS)	Assemble end time(MS)
1	793	5684	5724
2	26078	5308	4968
3	4211	6829	6909
4	4812	4379	4449
5	8632	2159	2339
6	43695	1210	1720

TABLE II Performance analysis of detect duplicates

It shows how much time requires for finding duplicates

File size (byte)	Total chunk	Duplicate Chunk	Start time (MS)	End time (MS)	Total storage time(MS)
143	4	4	1853	1863	10
26078	26	26	5740	5830	90
14447	15	0	4951	5051	100
7396	8	8	2693	2713	20

TABLEIIIPerformance analysis of Md5

File size (bytes)	Test String Name	Hash code	Hash time(ms)
26078	Chapter01. Docx	5f99e01b933531ca5db3b 438c48ffded	0
143	AA. rar	04960f7d18960e3483729 49e44baaa752	0
7376	123.jpg	247050306d37115f6d7cc 17d301ad949	0
14447	Abstract. Txt	Fa1bd9f53ecfb74f2f3c2b dc94bc60a	0

TABLE IV Performance Analysis of Retrieval Time

File size(kb)	File type	Sequential search(sec)	Retrieval Time(MS)
10	Odt	1	40
30	Rar	6	150
50	Txt	4	130

Retrieval time with all security algorithm work & deduplication or with indexing technique work this final proposed system result is calculated by taking combination of all b+ tree and deduplication results work and it is compared with sequential work [1] so we find that index performs better than sequential

1). Performance Analysis of Retrieval Time: In this graph retrieval time of a file is compared with sequential work so we find that index performs better than sequential [1]

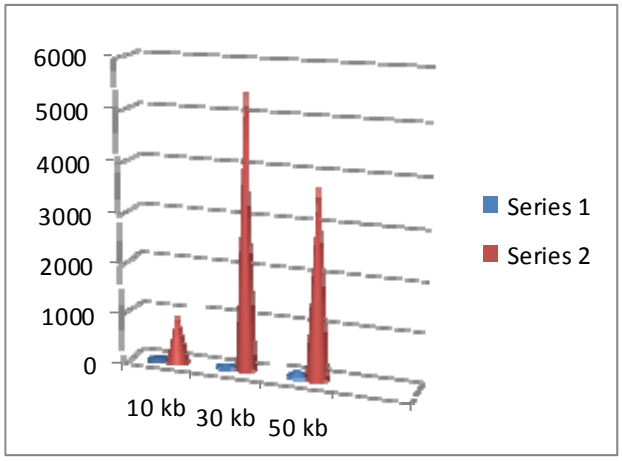

Fig 4 Graph for retrieval time of a file

This graph shows that index performs better than sequential search

This table shows the time required for performing Encryption & Decryption of a different size of files with different types

TABLE V Table for Encryption performance of AES algorithm

File size (kb)	Encryption time(MS)	Decryption time(MS)
4.1	2	1
12.3	5	3
28.7	12	10
54.5	16	14
40	13	11

This table shows the time required for performing Encryption & Decryption of a different size of files with different types

TABLE VI Table for Encryption performance of RSA algorithm

File size (bytes)	Encryption time(MS)	Decryption time(MS)
793	20	10
26078	20	20
4211	20	10

V. CONCLUSION

For faster retrieval of data we used indexing technique with deduplication technique for utilization of storage space and provide data security to cloud so it becomes perfectly managed and secured. And gives faster retrieval of data improves search time reduces storage space and data is secured by using communication security, data security, and integrity checking mechanism

REFERENCES

[1] Prabavathy; M Subha Devi; Chitra Babu "Multi-index technique for metadata management in private cloud storage "In IEEE International Conference on Recent Trends in Information Technology (ICRTIT), pp 84 - 89, 2013

[2] L. Xu, J. Hu, S. Mkandawire and H. Jiang. "SHHC: A Scalable Hybrid Hash Cluster for Cloud Backup Services in Data Centers. "In IEEE 31St International conference on Distributed Computing Systems Workshops (ICDCSW), pp. 61-65, 2011.

[3] W. Zeng, Y. Zhao, K. Ou and W. Song. "Research on cloud storage architecture and key technologies. "In Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human, pp. 1044-1048. ACM, 2009.

[4] D. Bhagwat, K. Eshghi, D. D. E. Long and M. Lillibridge. "Extreme Binning: Scalable, parallel deduplication for chunk-based file backup". IN IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, pp. 1 - 9, 2009.

[5] T. T. Thwel and N. L. Thein. "An Efficient Indexing Mechanism for Data Deduplication." In IEEE International Conference on the Current Trends in Information Technology (CTIT), pp. 1-5, 2009.

[6] Shraddha sadavarte, Bapusaheb B. Bhusare, "cloud computing storage security server", International Journal of engineering Research in Computer science engineering (IJERCSE) June 19, 2016 vol 3 issue 6.

[7] C. Policroniades and I. Pratt. "Alternatives for detecting redundancy in storage systems data." In Proceedings of the USENIX Annual Technical Conference, pp. 73-86, 2004.

[8] Text Book: Cryptography and network security, Principles and practices by William Stalling, Retrieved on 8 December 2006

[9]. Xia, W., Jiang, H., Feng, D., Hua Y.: Silo: a Similarity-locality based Near-exact Deduplication Scheme with Low RAM Overhead and High Throughput. In: Proc. of USENIX ATC (2011)

[10] EMC Data Domain Global Deduplication Array.http://www.datadomain.com/products/global-deduplication-array

[11] IBM Protect TIER Deduplication Gateway.http://www03.ibm.com/systems/storage/tape/ts7650g/index.html

[12].SEPATONS21OES2.http://www.sepaton.com/products/SEPATON_ES2.html

[13] Dubnicki, C., Gryz, L., Heldt, L., Kaczmarczyk, M., Kilian, W., Strzelczak, P., Szczepkowski, J., Ungureanu, C., Welnicki, M.: HYDRAstor: a Scalable Secondary Storage. In: Proc. of USENIX FAST (2009)

[14] Y. Fu, H. Jiang, N. Xiao and L. Tian. "Aa-dedupe: An applicationaware source deduplication approach for cloud backup services in the personal computing environment." In IEEE International Conference on Cluster Computing, pp. 112-120, 2011.

[15] C. Policroniades and I. Pratt. "Alternatives for detecting redundancy in storage systems data." In Proceedings of the USENIX Annual Technical Conference, pp. 73-86, 2004.

Shraddha S. Sadavarte Completed B.E. from Anuradha Engineering College Chikhli, Maharashtra. Doing PG in M.E. (CSE) from MSS's CET Jalna, Maharashtra, India