ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016

Data Deduplication with Attribute Based Encryption Using Multiple Servers on Cloud

Thorat Surekha Sampatrao[1], Prof. S.R. Lahane [2]
PG Scholar, Dept.Of Computer Engineering, GES's R. H. Sapat COE, Nashik, Maharashtra, India[1]
Assistant Professor, Dept.Of Computer Engineering, GES's R. H. Sapat COE, Nashik, Maharashtra, India[2]

ABSTRACT— Data deduplication technique is used to reduce the storage space requirement of the organizations. Data de-duplication is used to remove the multiple copies of same data. By deduplication we save only unique copy of the data and replace all other copies with a pointer which points to the original data file. The proposed system performs the deduplication in two levels i.e. file level and block level. In file level deduplication check the token is generated for the file and check on the storage for the same token, if the token is existed in the public cloud then instead of uploading the file the user gets the file pointer of the saved file for their reference and use. When the file level deduplication shows the result as file is unique then it go for the block level deduplication check. File level deduplication is already done but it has a drawback that it is useful only when both the files are unique. In proposed system block level deduplication is used to solve this issue, it divides the file into the blocks and then perform the deduplication check.

KEYWORDS—Deduplication, file level deduplication, block level deduplication token, reliability, secret sharing.

I. INTRODUCTION

Cloud computing offers infinite virtualized resources across the network for the users, and hide the platform and implementation details. Now a day the cloud service providers provide highly available storage space as well as the parallel computing resources at very low cost. As the cloud computing is growing rapidly, the more amount of data is stored in the cloud and many users can share it with specified privileges, which shows the access rights of the data. The data on the cloud is constantly growing and it's very hard to maintain that data for the cloud storage providers. Recently the deduplication of data provides a great solution to this problem, it offers scalability in cloud computing.

The deduplication is a technique which is used to reduce the duplicate data copies on storage. It is a data compression technique used to eliminate the redundant data on the cloud and improve the storage utilization. Deduplication saves only one copy of the file having the same content of data like other files and all the other files refer to that file for the data content. It means only one physical copy of data is available on the cloud and all the others are pointers which point towards the original file. Deduplication is done at file level means check for the whole data content of the file and eliminates the duplicate files or at block level means check for the same chunk of data content in non- identical files to avoid the duplicate blocks of data. Data deduplication has a various benefits but the privacy and security is a immense challenge as the insider and outsider attacks can spoil the sensitive data of the users. Users normally use the encryption or decryption techniques to provide the security to their data but the conventional encryption techniques are not provide deduplication. In old encryption techniques users encrypt the data with their own key that's why the same file gives the different cipher texts, so it is not possible to do the deduplication.To resolve this problem a hashing technique is used to calculate the hash value which allows deduplication for the data. Convergent key is computed using the cryptographic hash value of the file. Key generation and data encryption provides the data key to the data user and accumulate the cipher-text to the cloud. As

ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016

it uses deterministic operation which is derived from the data content same files will generate the same convergent key and therefore the cipher-text is also same. To control the unauthorized access, a safe proof of ownership protocol is required, which provides a proof that the user also possesses the file, if the duplicate is found. After this, instead of uploading the same file again on the server it provides a pointer of the same file to that consequent user. By using the pointer the user can download the required encrypted file from the server, which can be later on decrypted by their convergent keys. Thus, deduplication on cipher-text is made possible by using the convergent encryption method and to prevent the unauthorized users to access the file proof of ownership is used.

II. LITERATURE SURVEY

A. *Secure Deduplication*

Now days secure data deduplication has been very popular in cloud computing from investigate community. For the cloud storage deduplication system Yuan et al. [2] proposed an integrity check method which reduces the storage size of the tags. Bellare et al. [3] explained by converting the predictable message into unpredictable message we can protect the data confidentiality, which helps to improve the security and confidentiality of the data deduplication. To generate the file tag for replacement check, he introduced the third party called key server. Stanek et al. [4] provides a new encryption structure which provides differential security for popular or commonly used data and unpopular data which means rarely used data. For most commonly used data which is usually not much sensitive, traditional conventional encryption technique is used. Another two layered encryption schemes with sturdy security while supporting deduplication is proposed for seldom used data. Li et al. [5] distributes the keys across multiple servers after encrypting the files which tackles the key management issue in block-level deduplication.

B. *Convergent Encryption*

Convergent encryption provides deduplication with data privacy. Bellare et al. provides this primitive as message locked encryption, and determine its application in secure outsourced storage with space-efficiency. Xu et al. without taking into account issues of the key-management and block-level deduplication, it provides a secure convergent encryption for efficient encryption. There are also several implementations of different convergent encryption variants for secure deduplication.

C. *Proof of Ownership*

Halevi et al. proposed the concept of proofs of ownership (PoW) for deduplication systems. In this a user can use the file without uploading it on the cloud server. Various PoW constructions based on the Merkle-Hash Tree are proposed to facilitate client-side deduplication, which include the bounded leakage setting. Pietro and Sorniotti proposed another competent PoW scheme by choosing the projection of a file onto some arbitrarily selected bit-positions as the file proof. Recently, Ng et al. extended PoW for encrypted files, but they do not address how to minimize the key management overhead.

III.EXISTING SYSTEM

A number of deduplication systems have been proposed based on various deduplication strategies such as client-side or server-side deduplications, file-level or block-level deduplications. With the advent of cloud storage, data deduplication techniques become more attractive and critical for the management of ever-increasing volumes of data in cloud storage services which motivates enterprises and organizations to outsource data storage to third-party cloud providers, as evidenced by many real-life case studies . According to the analysis report of IDC, the volume of data in the world is expected to reach 40 trillion gigabytes in 2020. Today's commercial cloud storage services, such as Dropbox, GoogleDrive and Mozy, have been applying deduplication to save the network bandwidth and the storage cost with client-side deduplication.

ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016

Disadvantages:

- Duplication technique can save the storage space for the cloud storage service providers, it reduce the reliability of the system.
- Most of the previous deduplication systems have only been considered in a single-server setting.
- The traditional deduplication methods cannot be directly extended and applied in distributed and multi-server systems.

IV.PROPOSED SYSTEM

The proposed system work on both the methods, file level deduplication as well as block level deduplication. It first does the file level deduplication check and then work on block level deduplication check. The system divides in three levels Data User, Public Cloud and Private Cloud. The deduplication architecture to check the authority of the user in proposed system is same as the existing system as shown in fig. 1.

The proposed system model involves four parties:

- Data owner: Who outsources its encrypted data and encrypted keyword-index to the cloud;
- A cloud: It offers storage services and can conduct keyword search operations on behalf of the data users.
- Data user: Who repossess the data owner's encrypted data according to required keyword (i.e. keyword search).
- A trusted authority: Which provides credentials to the data owners or users for validation? In the proposed system the Client provides the following function calls to support deduplication and token generation along with the file upload process.
- FileTag (File) the process computes SHA-1 hash of the File as the File Tag.
- TokenReq (Tag, UserID) with the User ID and the File Tag the process requests for File Token generation to the Private Server or cloud.
- DupCheckReq (Token) by sending the file token received from the private server the process requests the Storage Server for the Duplicate Check of the File.
- ShareTokenReq (Tag, Priv.) the process requests to the Private Server or Cloud to produce the Share File Token with Target Sharing Privilege Set and the File Tag.
- FileEncrypt (File) the process encrypts the File in cipher block chaining (CBC) mode, using 256-bit of the AES algorithm with Convergent Encryption, where the convergent key is from SHA- 256 Hashing of the file.
- FileUploadReq(FileID, File, Token) if the file is Unique the process uploads the File to the Storage Server and the updates of the File Token is stored. In the proposed system the for the token generation the Private Server includes a corresponding request handler and maintains a key storage with Hash Map.
- TokenGen(Tag, UserID) the process first loads the associated privilege keys of the user and generate token with HMAC-SHA-1.

  In the proposed system the following handlers are used to maintains a record between existing files and associated token with Hash Map at Storage Server to offer deduplication and storage of data.
- DupCheck(Token) - It searches in the Token Map for the duplicate file; and
- FileStore(FileID, File, Token) It updates the Mapping when a new file is stored on the disk.

  All the above functions are used for the file level deduplication check. For block level deduplication check one more function is required at client level.
- Divide File(File, Block Size) It divides the file into the block and performs the duplication check at each block level.

V. METHODOLOGY USED

ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016

1. ***Attribute Based Encryption***

Attribute-based encryption (ABE) for one to-many encryption in which cipher-texts are encrypted for those who are able to fulfill certain requirements. The most suitable variant for fine-grained access control in the cloud is called cipher text-policy, in which cipher-texts are associated with access policies, determined by the encryptor and attributes describe the user, according to attributes are embedded in the users' secret keys. A cipher-text can be decrypted by someone if and only if, his attributes satisfy the access structure given in the cipher-text, thus data sharing is possible without prior knowledge of who will be the receiver preserving the flexibility of the cloud even after encryption.

2. ***Secret Sharing Algorithm with Lagrange Interpolation***

Secret sharing algorithm refers to method which is used for distributing a secret among a group of uses, each user is allocated a share of the secret. The secret can be reconstructed only when a enough number, of possibly different types, of shares are collective together; individual shares are of no use on their own.

Lagrange Interpolation method is used to achieve perfect security.

$$p_{(x)} = \sum_{i=0}^{t-1} y_i \prod_{\substack{0<j<t-1 \\ j\neq i}} \frac{x - x_j}{x_i - x_j}$$

For retrieving the data according to Lagrange Interpolation user must combine more or equal predefined number of shears.

3. ***MD5 Algorithm:***

Widely used cryptographic hash function Used to verify data integrity . In the proposed system MD5 algorithm is used to calculate hash value of data chunks. Used to avoid file level or block level collision attack. And also used for Sub file hashing or whole file hashing.

4. ***Working Methodology***

This system is divided in to two sections one is upload file and another is download file.

Methodology for File Upload:

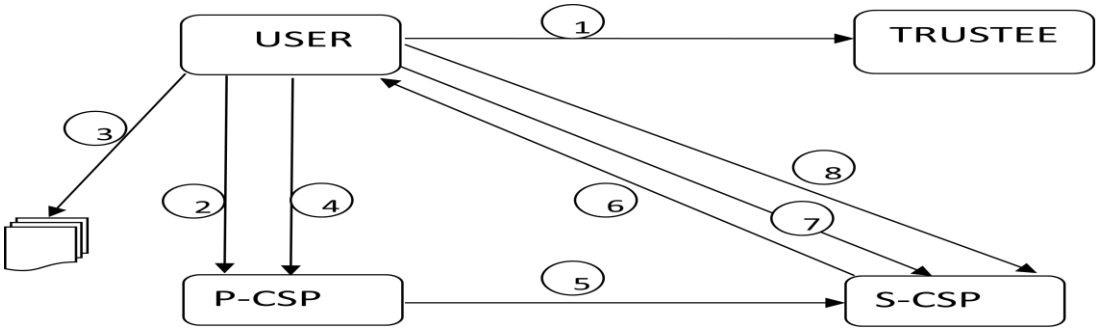


Fig. 1 System Architecture for file uploading

1) New User should register first.
2) User should Login to enter into the system.
3) User selects the file to upload.
4) File tag generation using Private cloud thus private cloud generate token.
5) Public cloud server checks If token exist or not.
6) If token exists then it will return pointer.
7) If token does not exist then cloud server return signal to upload the file on the server.
8) Upload the file on the server by dividing in to blocks.

ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016

Methodology for File Download:

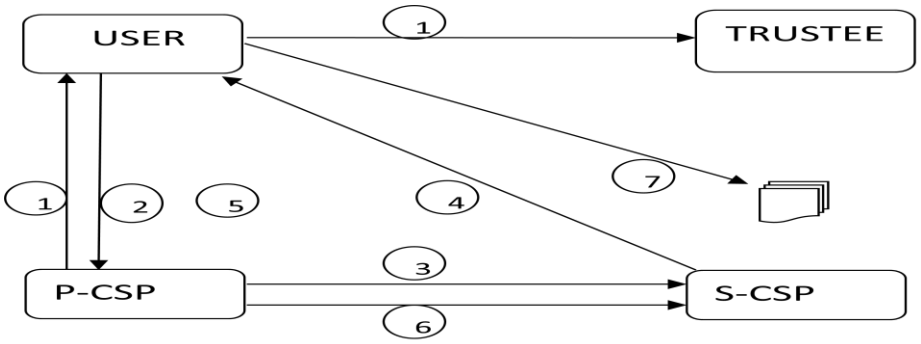


Fig. 2 System Architecture for file downloading

1) Public cloud return the identification token T to the user.

2) User Ask for file to download to Public cloud.

3) Public cloud checks the privileges of user.

4) If user has privileges it returns file info & decryption key to the user.

5) User sends file info and token to Private cloud.

6) Private cloud verifies the token and return file blocks to the user.

7) User decrypts the block and generates the original block.

5. *Mathematical Model:*

- Let D be the Data Deduplication System.
- Upload Process
- Input = {File F}
- H is the set of calculated hash values.
- H = h1, h2,h3,h4
- B = b1, b2,b3...
- where, h1 ← md5(F)
- h2 ← SHA(F)
- h3 ← md5(B)
- h4 ← Rabin (B)
- F = File, B= Block of file
- File deduplication ← Compare (h1, h2)
- Block deduplication ← Compare (h3, h4)
- File F = B1, B2,B3...
- B1 is set of various divided blocks.
- B1 =B11,B12,B13...
- Use (K,n)threshold scheme to generate share secrets
- where K < n , n = number of shares and k = minimum shares or threshold
- Determine n points by,

$$f(x) = a0 + a1x = a2x^2 + ...a_{k-1}^{k-1}$$

- Put x=1, 2,3...
- Generate Secret share(SS)
- Let CS = CS1, CS2, CS3...

ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016

- CS be the distributed cloud storage servers.
- CS1 ← SS1
- Store all secret shares over distributed Storage server.
- Output : Secret shares of whole file
- Download Process
- Input: Secret shares(SS)
- Determine Secret S
- To reconstruct the secret we are using following formula:
- For convenience, let yi denote P(xi). We can generate the coefficients of P using Lagrange Interpolation. Define,

$$p_{(x)} = \sum_{i=0}^{t-1} y_i \prod_{\substack{0<j<t-1 \\ j\neq i}} \frac{x - x_j}{x_i - x_j}$$

- Block B SS1, SS2, SS3...
- Generate complete file F
- F ← B1,B2,B3...

Output: Complete File F

6. *Security Analysis*

- Type-I Attack

This type of attack the attacker tries to convince the S-CSPs with some supporting information to get the content of the file stored at S-CSPs. To get one piece of share stored in a S-CSP, the user needs to perform a correct PoW protocol for the consequent share stored at the S-CSP. In this way, if the attacker wants to get the *k*-th piece of a share he does not own, he has to induce the *k*-th SCSP by correctly running a PoW protocol. However, the user cannot get the supplementary or secondary value used to perform PoW if he does not own the file.Hence, based on the security of PoW, the security against a Type-I attack is easily derived.

- Type-II Attack

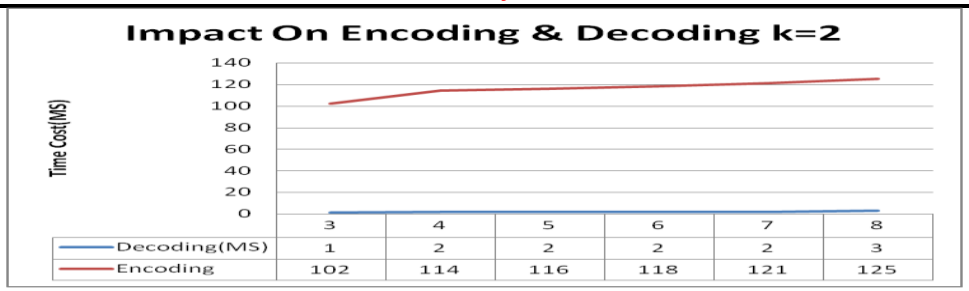
The data is processed before being outsourced to cloud servers. A secure secret sharing scheme has been applied to split each file into pieces, where each block is distributedely stored in a SCSP. Because the original secret sharing scheme is semantically secure, the data can not be recovered from pieces of shares that are less than a predefined threshold number. This means the confidentiality of the data stored at the S-CSPs is guaranteed even if some S-CSPs collude. In the secret sharing scheme, no information will be disclose even if any *r* of *n* shares collude. Thus, the data in our scheme remains secure even if any *r* S-CSPs collude.

VI. RESULTS

1) Impact on Encoding/Decoding time: case 1(Base Paper) case 1: r = 1, k = 2, and 3 ≤ n ≤ 8

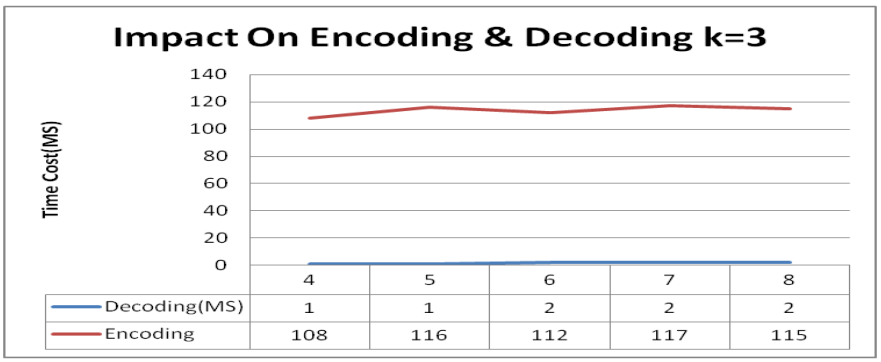
Number of Servers	Encoding(Ms)	Decoding(MS)
3	102	1
4	114	2
5	116	2
6	118	2
7	121	2
8	125	3

ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016



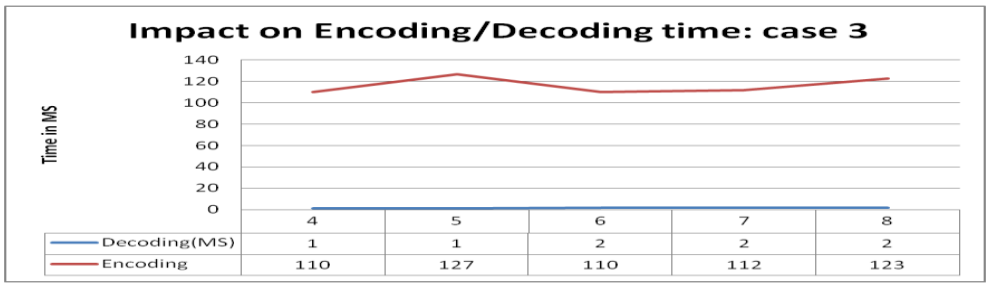
2) Impact on Encoding/Decoding time: case 2(Base Paper) case 2: r = 1, k = 3 and $4 \leq n \leq 8$

Number of SCP	Decoding(MS)	Encoding
4	1	108
5	1	116
6	2	112
7	2	117
8	2	115



3) Impact on Encoding/Decoding time: case3(Base Paper) Case3: r=2, k=3 and 4<n<8

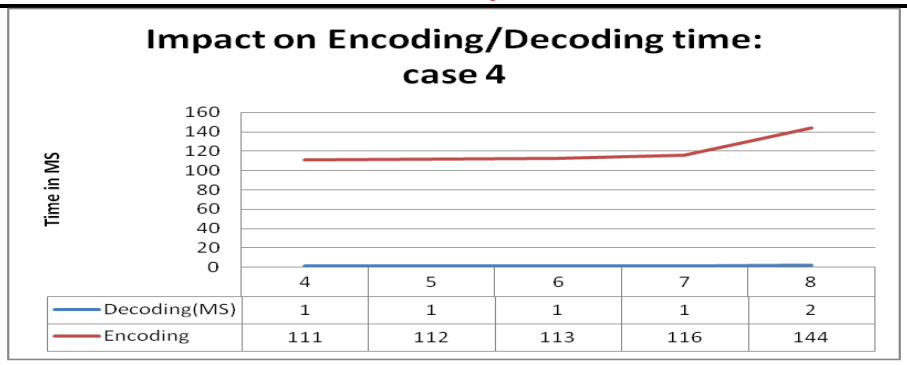
Decoding(MS)	Encoding	Number of Scp
1	110	4
1	127	5
2	110	6
2	112	7
2	123	8



4) Impact on Encoding/Decoding time: case4(Base Paper) Case 4 : r=2 k=4 and 5<n<8

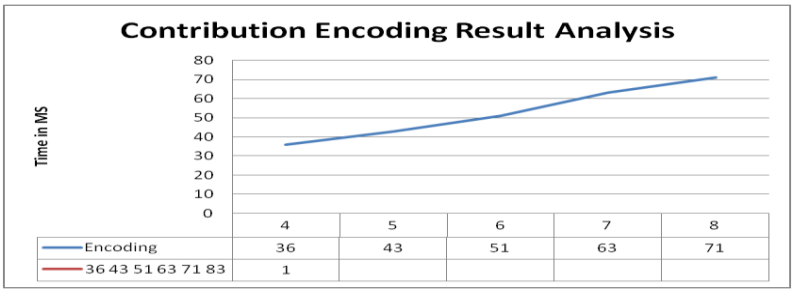
Number of Scp	Decoding(MS)	Encoding
4	1	111
5	1	112
6	1	113
7	1	116
8	2	144

ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016



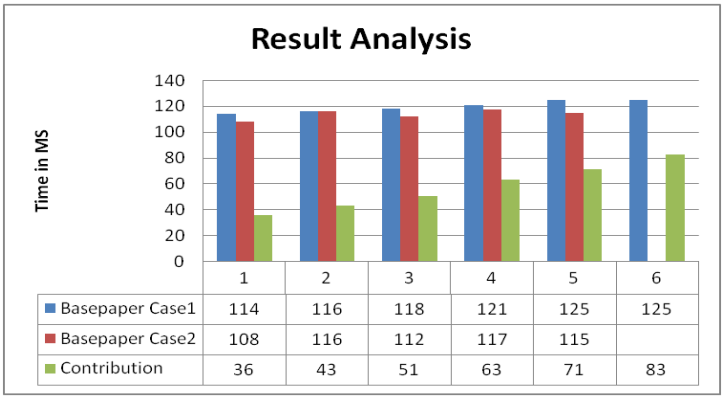
5) Contribution Encoding:

Encoding Time (Ms)	Number of Servers
36	4
43	5
51	6
63	7
71	8
83	9



6) Encoding Analysis

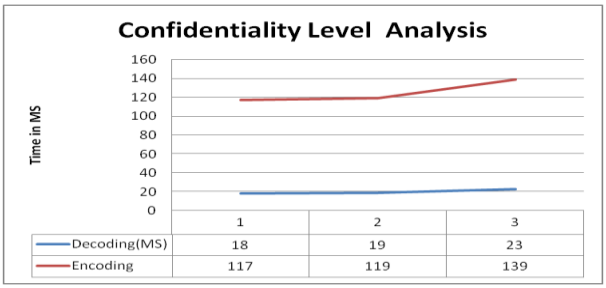
Case1 (r = 1, k = 2, and 3 ≤ n ≤ 8)	Case2 (r = 1, k = 3 and 4 ≤ n ≤ 8)	Contribution
114	108	36
116	116	43
118	112	51
121	117	63
125	115	71
125		83



ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016

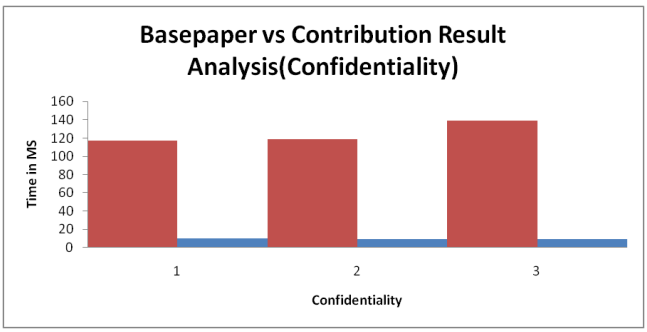
7) Impact on Encoding/Decoding time: case 1

Decoding(MS)	Encoding	Confidentiality
18	117	1
19	119	2
23	139	3



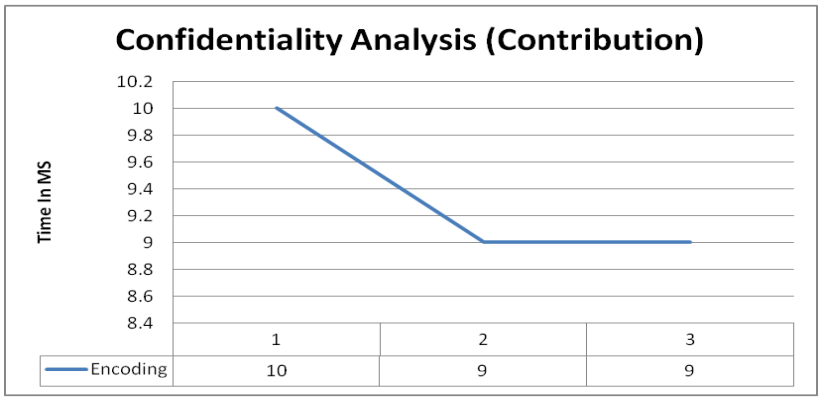
8) Base paper vs contribution result analysis (Confidentiality)

Encoding Contribution	Encoding Base Paper	Confidentiality
10	117	1
9	119	2
9	139	3



9) Confidentiality Analysis (Contribution)

Encoding	Confidentiality
10	1
9	2
9	3



ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 7, July 2016

VIII. CONCLUSION

In this paper, the distributed deduplication system is used to improve the reliability of data while achieving the confidentiality of the users' outsourced data without an encryption mechanism. The security of tag consistency and integrity were achieved. This system implemented using the Secret sharing scheme and demonstrated that it incurs small encoding/decoding overhead compared to the network transmission overhead in regular upload/download operations. The use of Lagrange Interpolation Scheme improves reliability and confidentiality of our system.

REFERENCES

[1] **S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, Proofs of ownership in remote storage systems, In Y. Chen, G. Danezis and V. Shmatikov, editors, ACM Conference on Computer and Communications Security, pages 491-500, ACM, 2011.**

[2] **J. Yuan and S. Yu., Secure and constant cost public cloud storage auditing with deduplication, IACR Cryptology ePrint Archive, 2013:149, 2013.**

[3] **M. Bellare, S. Keelveedhi, and T. Ristenpart, Dupless: Server- aided encryption for deduplicated storage, In USENIX Security Symposium, Harlow, 2013.**

[4] **J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl, A secure data deduplication scheme for cloud storage, In Technical Report 2013.**

[5] **J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, Secure deduplication with efficient and reliable convergent key management, In IEEE Transactions on Parallel and Distributed Systems 2013.**

[6] **J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer, Reclaiming space from duplicate files in a serverless distributed file system, In ICDCS, Harlow, pages 617-624, 2002.**

[7] **M. Bellare, S. Keelveedhi, and T. Ristenpart, Message-locked encryption and secure deduplication, In EUROCRYPT pages 296-312, 2013.**

[8] **J. Xu, E.-C. Chang, and J. Zhou, Weak leakage-resilient client- side deduplication of encrypted data in cloud storage, In ASIACCS, pages 195-206, 2013.**

[9] **P. Anderson and L. Zhang, Fast and secure laptop backups with encrypted de-duplication, In Proc. of USENIX LISA, 2010.**

[10] **A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S. Lui, A secure cloud backup system with assured deletion and version control, In 3rd International Workshop on Security in Cloud Computing, 2011.**

[11] **M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, Secure data deduplication, In Proc. of StorageSS, 2008.**

[12] **Z. Wilcox-OHearn and B. Warner., Tahoe: the least- authority filesystem, In Proc. of ACM StorageSS. 2008.**

[13] **R. D. Pietro and A. Sorniotti, Boosting efficiency and security in proof of ownership for deduplication,In H. Y. Youm and Y. Won, editors, ACM Symposium on Information, Computer and Communications Security, pages 81-82, ACM, 2012.**

[14] **W. K. Ng, Y. Wen, and H. Zhu., Private data deduplication protocols in cloud storage, In S. Ossowski and P. Lecca, editors, Proceedings of the 27th Annual ACM Symposium on Applied Computing, pages 441-446, ACM, 2012.**

[15] **S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider, Twin clouds: An architecture for secure cloud computing, In Workshop on Cryptography and Security in Clouds (WCSC 2011), 2011.**

[16] K. Zhang, X. Zhou, Y. Chen, X. Wang, and Y. Ruan, Sedic: privacyaware data intensive computing on hybrid clouds, In Proceedings of the 18th ACM conference on Computer and communications security, CCS11, pages 515-526, New York, NY, USA, 2011. ACM.