International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

A Survey on Deduplication Scheme in Cloud Storage

Deepa .D[1], Revathi .M[2]

Department of Computer Science and Engineering, Kingston Engineering College, India

Abstract: ***Due to less maintenance enterprises and organizations outsource data storage to third-party cloud providers. Due to increase in data volume there is a need of data management in cloud storages. Deduplication is a technique which eliminates duplicate copies of same data. To make data management scalable, deduplication has been a well-known technique to reduce storage space and upload bandwidth in cloud storage. So we are going to build a deduplication mechanism which makes use of a convergent encryption scheme for data confidentiality. We also enhance our deduplication mechanism by solving the key management problem cause by generating an enormous number of keys with the increasing number of users in the convergent encryption using the Ramp secret sharing scheme.***

Keywords: deduplication, convergent, tag, encryption, hash

1. Introduction

Nowadays large organisations, enterprise and individual users are used to store large amount of data in the third party server. The cloud that provides many storage services such as Dropbox, Mozy, and Memoral. To the efficient storage that services use the deduplication scheme, this technology improves 80% efficient use of the storage space. Before upload the data to the server user need to encrypt the data and then upload their data into the cloud. Each user encrypts the data with their own encryption algorithm, in these identical data copies that produce the different cipher text, this make the deduplication process impossible. Convergent encryption is mainly used for the deduplication process.

Convergent encryption is applying the hash function to the data copies and that key is used to encrypt the data and the upload into the cloud. The key that is used to encrypt the data is called convergent key. The use retains the key and uploads the encrypted data to the cloud. Using convergent key data is decrypted by the user. This makes the user data confidential and also improves the deduplication process.

The encryption process can be done in two ways file level and block level. File level encryption is whole file encrypted once this encryption reduce the deduplication efficiencies because if any small change occur means it produce the different cipher text, so we go for block level, in this whole file is split into small chunks and each chunk is encrypted separately and upload the encrypted file into the cloud. Suppose another user upload the same file with small changes means that chunk only stored in the cloud instead of entire file. This block level duplication increase the efficiency of deduplication process.

2. Existing Techniques

The cloud environment is a large open distributed system. It is important to preserve the data, as well as privacy of the users. Existing techniques in cloud storage are ***Redundant Array of Cloud Storage (RACS):*** This RACS is used to stored data over multiple vendors. This is act as a proxy and performs the operation between the client and server. This method is simple and easy to work with. The drawback is single proxy cloud easily become bottleneck. ***Secure Overlay Service (SOS):*** This SOS is mainly used to solve the distributed denial of service, the idea provide by this solution is very complicated. It is unclear about the optimal solution. ***Vanish****:* This technique of storage is all data become unreliable after some specific period of time for the security purpose. The data will be deleted after the particular time period with the knowledge of the owner who created the data. This technique is more expensive and it requires large Distributed Hash Table (DHT). ***FADE:*** This FADE is secure overlay in cloud storage with assured deletion. The data owner can be sure of the deleted file. Only the deleted part of the file is considered not the accessing data. This method is more complex to implement.

3. Proposed Technique

Our proposed technique is deduplication scheme. This technique eliminates the content of same of data and stored once. If the cloud service providers implements this technique they can reduce the storage space and the uploading bandwidth. This method reduces the cost of the storage and increase the security of the data.

Attacks possible in deduplication: Predicting File*:* In this attack suppose the attacker wants to find out whether user1 possesses a file, File A. He will upload a copy of file A if the file gets uploaded this will indicate that the file is not possessed by the user1 otherwise the attacker easily finds out the file posses by the user.

Creating secret channel: In this attack the attacker install malicious software on the users A machine. This malicious software creates the side channel between the user and the attacker. Then the attacker easily finds out the users information. ***The content distribution attack:*** In this attack the attacker distribute the some file in the storage system to check whether hash value is matched with the user A file, if it is matched then that file can be easily compromised.

Solution for the attacks: Solution for those attacks is given by encrypt the file before uploading, performing target based deduplication. Randomization, Gateway based

deduplication, proof of Ownership. In this way the deduplication is performed then we can store 80% storage space and the efficient uploading bandwidth.

4. Convergent Encryption

The solution to those attacks is before uploading the data into the server encrypt the data then upload. If the different user applies the different encryption techniques that will generate the different cipher text. If the same data produces the different cipher text means it reduces the deduplication process. For this convergent encryption is implemented to produce the same ciphertexts for the same data. Convergent encryption is applying the hash function for the data and produces the key that is known as convergent key. This convergent key is used to encrypt the data. User retains the key and uploads the data to the server. While downloading the user decrypt the data using the convergent key.

The following architecture explain the procedure to upload the file in the cloud and block creation, encrypted data are stored in the cloud server.

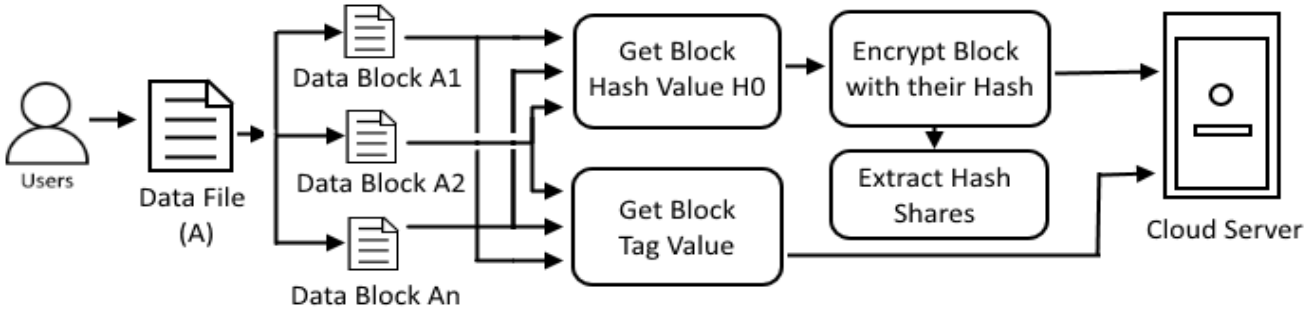

Figure4.1: File uploading

If the data is just applied for the hash function means there is a lots of chance to attacker to predict the file, for that this hash algorithm that take all first letter of the file and produce the key and that key is used to produce the encryption, that encrypted file is uploaded in the cloud server.

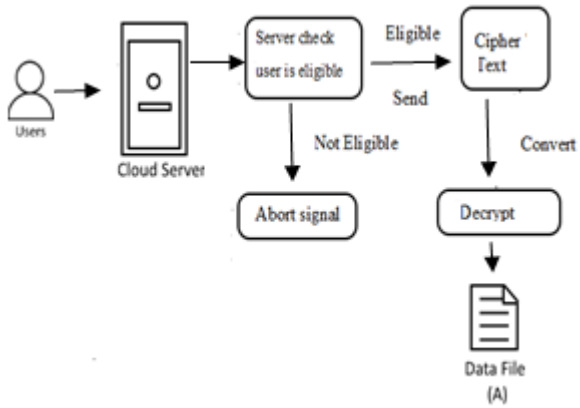

Figure 4.2: File Downloading

To download a file from the cloud the above architecture is implemented. The user first sends the request to the server with the particular file name. Then the server will check that the particular user is authorized or not for that merkle tree is executed in this the server will ask for a super logarithmic number of leaves chosen at random. Then the client also execute the same and give the random number if both the number is same then the user is identified as a authorised otherwise the server will give the abort signal. This merkle tree provides the security for the unauthorised access. These are the file upload and the downloading process.

5. Problems In Converget Encryption

There is a chance to vulnerability activity that can be performed in convergent encryption. Potential malicious cloud providers can perform offline dictionary attacks and discover predicable files. This explains why a strategy is needed to enforce security while retaining benefits offered by deduplication and convergent encryption.

6. Related Work

Convergent encryption that is particularly used to achieve both deduplication and data confidentiality, but several well known weaknesses is available to predict the file using dictionary attacks. In dictionary attack the hacker try all possible related tag value to get the required file. For this problem we introduce the new hashing technique, this hash function that take all first letter of the file and then produce the convergent key, that key is used to encrypt the data and produce the cipher text. If the encryption can be done in this way there is a less chance to dictionary attack no one can predict the file.

7. Example

Using the gathered information so far, able to estimate the cost of backing up all the machines we have to Amazon S3 over a six-month period. The initial upload of 1.94 TB data will cost $434 without data de-duplication. With data de-duplication, only a total of $370 ($164 for upload) is needed, saving $355 or 49.0% of the initial upload cost. Monthly estimated bills are presented this table.

Table: Cost Estimation

	Conventional		**De-Duplicted**	
Months	Storage	Cost	Storage	Cost
1	1.94 TB	$725	1.37 TB	$370
2	2.35 TB	$445	1.66 TB	$284
3	2.76 TB	$507	1.95 TB	$328
4	3.18 TB	$569	2.25 TB	$372
5	3.59 TB	$630	2.54 TB	$415
6	4.00 TB	$692	2.83 TB	$459
Total		**$3,568**		**$2,228**

Another example shows that the storage space for the cloud service providers is compared with the existing storage with the deduplication storage. More storage space is reduced in the deduplication scheme.

Number of files in thousands	Storage space in GB	
	Normal storage	Deduplication storage
3	50	0.2
6	100	0.3
9	150	0.4
12.5	200	0.5
13	250	0.6
15	300	0.6
19	350	0.7
25	400	0.7
30	450	0.8
35	500	0.8
40	550	0.8
50	600	0.9
60	700	1
70	800	1

8. Conclusion

We propose an efficient and reliable convergent encryption scheme for secure deduplication. RSSS applies deduplication among convergent keys and distributes convergent key shares across multiple key servers, while preserving semantic security of convergent keys and confidentiality of outsourced data. We also explained detailed about secure file upload and downloading process.

References

[1] M. Bellare, S. Keelveedhi, and T. Ristenpart, ‘‘Message-Locked Encryption and Secure Deduplication,’’ in Proc. IACR CryptologyePrint Archive, 2012, pp. 296-3122012:631.
[2] 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 Proc. ICDCS, 2002, pp. 617-624.
[3] M.W. Storer, K. Greenan, D.D.E. Long, and E.L. Miller, ‘‘Secure Data Deduplication,’’ in Proc. StorageSS, 2008, pp. 1-10.
[4] A. Yun, C. Shi, and Y. Kim, ‘‘On Protecting Integrity and Confidentiality of Cryptographic File System for Outsourced Storage,’’ in Proc. ACM CCSW, Nov. 2009, pp. 67-76.
[5] 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 Proc. 3rd Int’l Workshop Security Cloud Comput., 2011, pp. 160-167.
[6] W.K. Ng, Y. Wen, and H. Zhu, ‘‘Private Data Deduplication Protocols in Cloud Storage,’’ in Proc. 27th Annu. ACM Symp. Appl. Comput., S. Ossowski and P. Lecca, Eds., 2012, pp. 441-446.
[7] D.T. Meyer and W.J. Bolosky, ‘‘A Study of Practical Deduplication,’’ in Proc. 9th USENIX Conf. FAST, 2011, pp. 1-13.
[8] D. Harnik, B. Pinkas, and A. Shulman-Peleg, ‘‘Side Channels in Cloud Services: Deduplication in Cloud Storage,’’ IEEE Security Privacy, vol. 8, no. 6, pp. 40-47, Nov./Dec. 2010