International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 6, June 2017)

Implementation on Secured Hybrid Cloud Storage Sevice Provider with Deduplication

Dinesh Shinde[1], Amit Dangi[2]

MTech CSE, IITM Bhopal, M.P., India

Asst. Professor CSE, IITM Bhopal, M.p., India

***Abstract*-- Data deduplication is one of important data compression techniques for eliminating duplicate copies of repeating data, and has been widely used in cloud storage to reduce the amount of storage space and save bandwidth. To protect the confidentiality of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing. To better protect data security, this paper makes the first attempt to formally address the problem of authorized data deduplication. Different from traditional deduplication systems, the differential privileges of users are further considered in duplicate check besides the data itself. We also present several new deduplication constructions supporting authorized duplicate check in a hybrid cloud architecture. Security analysis demonstrates that our scheme is secure in terms of the definitions specified in the proposed security model. As a proof of concept, we implement a prototype of our proposed authorized duplicate check scheme and conduct testbed experiments using our prototype. We show that our proposed authorized duplicate check scheme incurs minimal overhead compared to normal operations.**

***Keywords*-- Deduplication authorized duplicate check, confidentiality, hybrid cloud.**

I. INTRODUCTION

It eliminates duplicate copies of the same file. Deduplication can also take place at the block level, which eliminates duplicate blocks of data that occur in non-identical files.

Although data deduplication brings a lot of benefits, security and privacy concerns arise as users' sensitive data are susceptible to both insider and outsider attacks. Traditional encryption, while providing data confidentiality, is incompatible with data deduplication. Specifically, traditional encryption requires different users to encrypt their data with their own keys. Thus, identical data copies of different users will lead to different cipher texts, making deduplication impossible. Convergent encryption has been proposed to enforce data confidentiality while making deduplication feasible. It encrypts/decrypts a data copy with a convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users retain the keys and send the ciphertext to the cloud.

II. GOALS AND OBJECTIVES

1. Unforged ability of file token/duplicatecheck token. Unauthorized users without appropriate privileges or fileshould be prevented from getting or generating the file tokens for duplicate check of any file stored atthe SCSP. The users are not allowed to collude with the public cloud server to break the unforged ability of filetokens. In our system, the SCSP is honest but curious and wil honestly perform the duplicate check uponreceiving the duplicate request from users. The duplicate check token of users should be issued from theprivate cloud server in our scheme.

2. In distinguishability of file token/duplicatecheck token. It requires that any user without querying the privatecloud server for some file token, he cannot get any useful information from the token, which includes the fileinformation or the privilege information.
3. Data Confidentiality. Unauthorized users without appropriate privileges or files, including the SCSP and theprivate cloud server, should be prevented from access to the underlying plaintext stored at SCSP. In anotherword, the goal of the adversary is to retrieve and recover the files that do not belong to them. In our system,compared t.o the previous definition of data confidentiality based on convergent encryption,a higher level confidentiality is defined and achieved.

Objective

1. To improved integrity
2. To increase the storage utilization
3. To remove the duplicate copies of data and improve the reliability.
4. To improve the security

III. Literature Survey

Cloud storage service providers such as Drop box, Mozy, and others perform deduplication to save space by only storing one copy of each file uploaded. Should clients conventionally encrypt their files, however, savings are lost. Message-locked encryption (the most prominent manifestation of which is convergent encryption) resolves this tension. However it is inherently subject to brute-force attacks that can recover files falling into a known set. We propose an architecture that provides secure deduplicated storage resisting brute-force attacks, and realize it in a system called DupLESS. In DupLESS, clients encrypt under message-based keys obtained from a key-server via an oblivious PRF protocol.

It enables clients to store encrypted data with an existing service, have the service perform deduplication on their behalf, and yet achieves strong confidentiality guarantees. We show that encryption for deduplicated storage can achieve performance and space savings close to that of using the storage service with plaintext data.We studied the problem of providing secure outsourced storage that both supports deduplication and resists brute-force attacks. We design a system, DupLESS that combines a CE-type base MLE scheme with the ability to obtain message-derived keys with the help of a key server (KS) shared amongst a group of clients. The clients interact with the KS by a protocol for oblivious PRFs, ensuring that the KS can cryptographically mix in secret material to the per-message keys while learning nothing about files stored by clients. These mechanisms ensure that DupLESS provides strong security against external attacks which compromise the SS and communication channels (nothing is leaked beyond file lengths, equality, and access patterns), and that the security of DupLESS gracefully degrades in the face of comprised systems. Should a client be compromised, learning the plaintext underlying another client's ciphertext requires mounting an online bruteforce attacks (which can be slowed by a rate-limited KS). Should the KS be compromised, the attacker must still attempt an offline brute-force attack, matching the guarantees of traditional MLE schemes. The substantial increase in security comes at a modest price in terms of performance, and a small increase in storage requirements relative to the base system. The low performance overhead results in part from optimizing the client-to-KS OPRF protocol, and also from ensuring DupLESS uses a low number of interactions with the SS. We show that DupLESS is easy to deploy: it can work transparently on top of any SS implementing a simple storage interface, as shown by our prototype for Dropbox and Google Drive.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 6, June 2017)

IV. Existing System Approch

From the above writing review we have inferred that current information deduplication frameworks, the private cloudare included as an intermediary to permit information proprietor/clients to safely perform copy check with differentialbenefits. Such design is functional and has pulled in much consideration from specialists. The information proprietorsjust outsource their information stockpiling by using open cloud while the information operation is overseen in privatecloud.We display a propelled plan to bolster more grounded security by scrambling the document with differentialbenefit keys. Along these lines, the clients without comparing benefits can't perform the copy check. Moreover, suchunapproved clients can't decode the figure message even connive with the S-CSP.

V. Algorhythm Used

1) Convergent Encryption

Convergent encryption provides data confidentiality in deduplication. A user (or data owner) derives aconvergent keyfrom each original data copy and encrypts the data copy with the convergent key. In addition, the user also derives a tagfor the data copy, such that the tag will be used to detect duplicates. Here, we assume that the tag correctness propertyholds, i.e., if two data copies are the same, then their tags are the same. To detect duplicates, the user first sends the tagto the server side to check if the identical copy has been already stored. Note that both the convergent key and the tagare independently derived and the tag cannot be used to deduce the convergent key and compromise dataconfidentiality. Both the encrypted data copy and its corresponding tag will be stored on the server side.

A convergent encryption scheme can be defined with four primitive functions:

1. KeyGenCE(M)!K is the key generation algorithm that maps a data copy M to a convergent key K.
2. EncCE(K, M)!C is the symmetric encryption algorithm that takes both the convergent key K and the data copy M as inputs and then outputs a ciphertextC;
3. DecCE(K, C)!M is the decryption algorithm that takes both the ciphertextC and the conv ergent key K as inputs and then outputs the original data copy M; and
4. TagGen(M)!T (M) is the tag generation algorithm that maps the original data copy M and outputs a tag T(M).

2) Proof Of Ownership

The notion of proof of ownership(POW) enables users to prove their ownership of data copies to the storage server. Specifically, POW is implemented as an interactive algorithm (denoted by POW). The verifier derives a short value $\phi(M)$ from a data copy M. To prove the ownership of the data copy M, the properneeds to send $\phi$to the verifier such that $\phi = \phi(M)$.

PSEUDO CODE

Step1: Calculate the two convergent key values

Step2: Compare the two keys and files get accessed.

Step3: Apply de-duplication to eradicate the duplicate values.

Step4: Ifany other than the duplicates it will be checked once again and make the data unique.

Step5: That data will be unique and also more confidential the authorized can access and data is stored.

VI. System Architecture

1. Secret Sharing Scheme:

Secret sharing scheme performs two operations namely Share and Recover. The secret is divided and shared by usingShare. With enough shares, the secret can be extracted and recovered with the algorithm of Recover.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 6, June 2017)

The input to thismodule is file. It performs dividing of file into fixed size blocks or shares. These blocks are then encoded and allocatedon cloud server at different nodes. When user request for file these blocks are decrypted and by combining these blocksfile is given to user.

2. Tag Generation:

In this tag similarity is considered a kind of semantic relationship between tags, measured by means of relative cooccurrence between tags, known as J. coefficient. The input to this block is file blocks. This module assigns tags toeach block for duplication check. The output of this module is blocks with tag assigned.

3. Convergent Encryption Module

Traditional encryption, while providing data confidentiality, is incompatible with data deduplication. Specifically,traditional encryption requires different users to encrypt their data with their own keys. Thus, identical data copies ofdifferent users will lead to different cipher texts, making deduplication impossible. Convergent encryption has beenproposed to enforce data confidentiality while making deduplication feasible. It encrypts/ decrypts a data copy with aconvergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After keygeneration and data encryption, users retain the keys and send the cipher text to the cloud. Since the encryptionoperation is deterministic and is derived from the data content, identical data copies will generate the same convergentkey and hence the same cipher text.

VII. Implementation Result

We implement a prototype of the proposed authorized deduplication system, in which we model three entities asseparate C++programs.

A Client program is used to model the data users to carry out the file upload process. A PrivateServer program is used to model the private cloud which manages the private keys and handles the file tokencomputation. A Storage Server program is used to model the S-CSP which stores and deduplicated files implementation of the Client provides the following function calls to support token generation and deduplication along the file upload process.

1.FileTag (File) - It computes SHA-1 hash of the File as File Tag;

2.TokenReq (Tag, UserID) It requests the Private Server for File Token generation with the File TandUser ID;

3.DupCheckReq (Token) It requests the Storage Server for Duplicate Check of the File by sending the file token receivedfrom private server;

4.ShareTokenReq (Tag, {Priv.}) It requests the Private Server to generate the Share File Token with the File Tag andTarget Sharing Privilege Set;

5.FileEncrypt (File) It encrypts the File with Convergent Encryption using 256bit AES algorithm in cipher block chaining

6.(CBC) mode, where the convergent key is from SHA-256 Hashing of the file;

7.FileUploadReq (FileID, File, Token) It uploads the File Data to the Storage Server if the file is Uni que and updates the

8.File Token stored. Our implementation of the Private Server includes corresponding request handler s for the tokengeneration and maintains a key storage with Hash Map.

9.TokenGen (Tag, UserID) It loads the associated privilege keys of the user and generate the token with HMAC-SHA-1

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 6, June 2017)

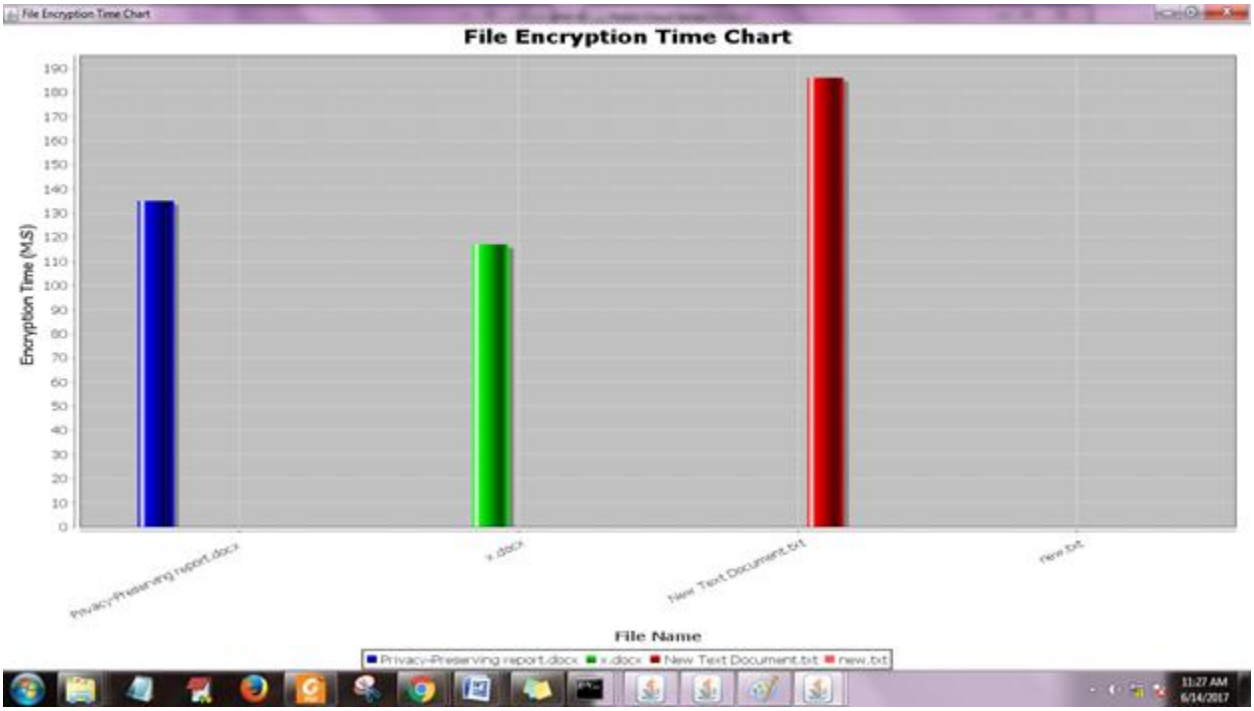
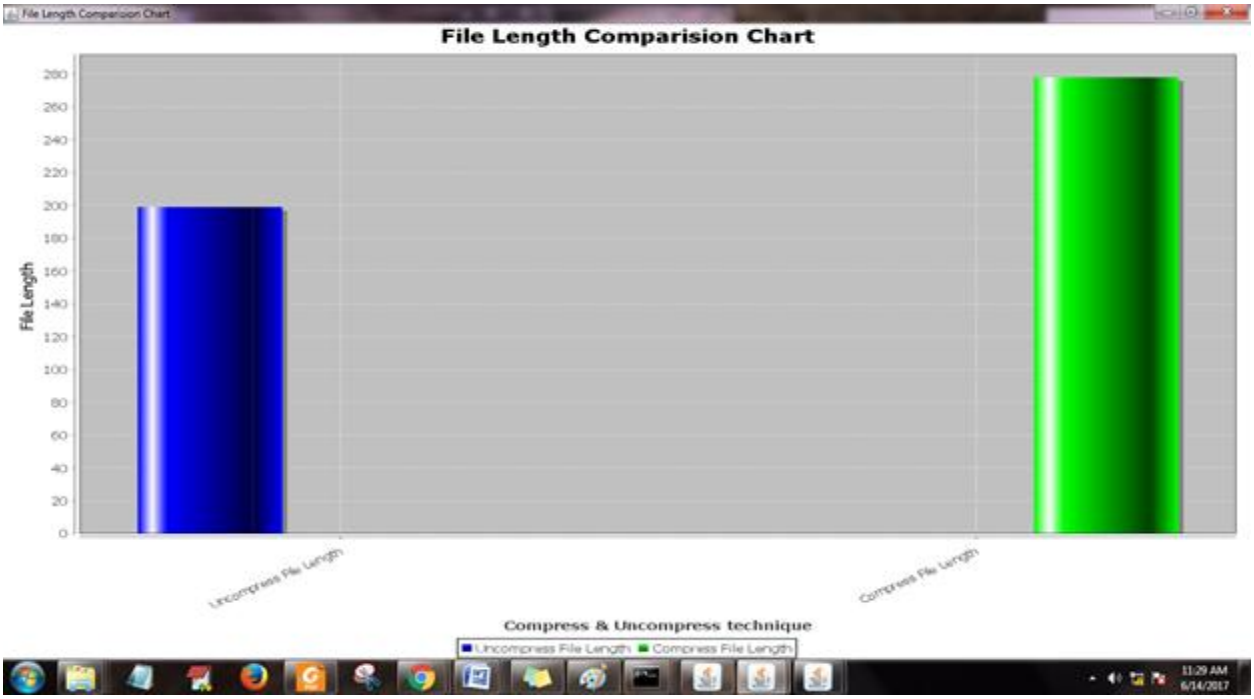


Fig.01 Time Breakdown for Different File Size

There is also used to reduce upload & download file by compressing its size using lossless compression algorithm. To Secure Data or Provide Confidentiality at the time of store on cloud whenever user download it from public cloud it is downloaded in the form of compressed file to reduce its size & also protect from virus & malware attack. The following chart shows the original file & compressed file result.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 6, June 2017)



As a proof of concept, we implemented a prototype of our proposed authorized duplicate check scheme and conduct test bed experiments on our prototype. We showed that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.Compression is useful because it reduces resources required to store and transmit data. Computational resources are consumed in the compression process and, usually, in the reversal of the process (decompression). Data compression is subject to a space–time complexity trade-off. In signal processing, data compression, source coding, or bit-rate reduction involves encoding information using fewer bits than the original representation. Compression can be either lossy or lossless. Lossless compression reduces bits by identifying and eliminating statistical redundancy. No information is lost in lossless compression. We showed that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.

VIII. Future Work

Currently this system is Standalone application where we create dummy cloud to verify the data on private cloud & public cloud. Also check the authentication of user & administrator but in future we also implement this technique on real time cloud server which is available in market for eg, Amazon Real time cloud server. By paying their money they allowed to give space on their server to upload & download data also user can use their actual email id to check authentication of user to provide their private keys & token for encryption to their private account.

REFERENCES

[1] S. Plank LihaoXu Optimizing Cauchy ReedSolomon Codes for FaultTolerant Network Storage Applications The 5th IEEE InternationalSymposium on Network Computing and Applications (IEEE NCA06), Cambridge, MA, July, 2006

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

[3] M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Serveraided encryption for deduplicated storage. In USENIX Security Symposium, 2013.

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

[5] M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. J. Cryptology, 22(1):1–61, 2009.

[6] M. Bellare and A. Palacio. Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In CRYPTO, pages 162–177, 2002.

[7] 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.

[8] 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, pages 617–624, 2002.

[9] D. Ferraiolo and R. Kuhn. Role-based access controls. In 15th NIST-NCSC National Computer Security Conf., 1992.

[10] S. Plank LihaoXu Optimizing Cauchy ReedSolomon Codes for FaultTolerant Network Storage Applications The 5th IEEE InternationalSymposium on Network Computing and Applications (IEEE NCA06), Cambridge, MA, July, 2006[10]Backialakshmi. N Manikandan. M SECURED AUTHORIZED DEDUPLICATION IN DISTRIBUTED SYSTEM IJIRST InternationalJournal for Innovative Research in Science and Technology— Volume 1 — Issue 9 — February 2015