

DCT Image Compression and Secure Deduplication with Efficient Convergent Key Management

Soofiya MS¹, Sreetha V Kumar²

¹M. Tech Student, Marian Engineering College, Trivandrum, Kerala, India

²Assistant Professor, Dept. of CSE, Marian Engineering College, Trivandrum, Kerala, India

Abstract: *As our use and reliance of computer grows day by day we need an efficient way to manage our large amount of data. Sometimes a person has to use large amount of images like webpages or catalogs need to use some techniques for image compression. This is because the amount of space required to hold large amount of image can be prohibitively large in terms of cost. Fortunately we have several method of image compression available today. Discrete Cosine Transform separates images into parts of different frequencies. In a step called quantization in DCT the less important frequencies are discarded. Then for decompression process only most important frequencies are considered to recover the image. Management of large amount of data has become one of the greatest problem in today's cloud storage services. For data management a technique called Deduplication is introduced. Deduplication avoids duplicate copies being stored in the cloud. Convergent encryption is an efficient way to enforce deduplication. But the problem in making convergent encryption possible is the management of large number of keys. To this end for the management of keys we propose a new technique called Dekey. We implement Dekey using visual cryptography. This paper proposes Dekey for convergent key management, DCT image compression for minimizing storage space and for improving security and deduplication for avoiding redundant copies.*

Keywords: DCT image compression, Deduplication, Dekey, convergent encryption, visual cryptography

1. Introduction

By entering the digital age the world has to handle a large amount of information. We have to store, retrieve, analyze and process digital information in an efficient way in order to put in practical use. Compressing an image is different from compressing raw binary data. One among many techniques that comes under image processing is image compression. Image compression has got various applications and also plays an important role in efficient transmission and storage of images. The image compression aims at reducing redundancy in image data to store or transmit only a minimal number of samples. And from this we can reconstruct a good accession of the original image in accordance with human visual perception. Fortunately have several method of image compression available today. DCT image compression comes under two main categories. Lossless and lossy image compression. JPEG is a widely adopted technique that centers around Discrete Cosine Transform. Discrete Cosine Transform works by separating images into parts of different frequencies. During a step in DCT called quantization the less important frequencies are discarded. Then for decompression process only most important frequencies are considered to recover the image.

Data deduplication is a well-known technique for management of data by eliminating the redundant copies being stored. In deduplication only one physical copy is kept and other copies are referred to this physical copy. Although convergent encryption is used for deduplication the main challenge is the efficient management of keys with increase in number of users. Each party that outsources their data to cloud encrypt the data with different keys for sake of confidentiality producing different cipher text even for same data thus making deduplication impossible [7]. Convergent encryption provides an option for data confidentiality and reliability while making Deduplication possible. For

implementing secure storage additional to deduplication the data is embedded in an image. The image is then compressed using the DCT image compression technique [4].

This paper makes the following contribution:

- A new construction Dekey [1] is proposed for efficient and reliable convergent key management through convergent key deduplication and secret sharing.
- Security analysis demonstrates Dekey is secure in terms of definition specified in proposed security model.
- We implement Dekey using visual cryptographic method that enables the key management to adopt to different confidentiality levels.
- DCT image compression is used to reduce the upload bandwidth.

2. Related Works

As the world moves to digital storage for archival purposes, there is an increasing demand for systems that can provide secure data storage in a cost-effective manner. By identifying common chunks of data both within and between files and storing them only once, deduplication can yield cost savings by increasing the utility of a given amount of storage. Unfortunately, deduplication exploits identical content, while encryption attempts to make all content appear random; the same content encrypted with two different keys results in very different ciphertext. Thus, combining the space efficiency of deduplication with the secrecy aspects of encryption is problematic [9]. To protect the confidentiality of outsourced data, various cryptographic solutions have been proposed in the literature. Their idea builds on traditional (symmetric) encryption, in which each user encrypts data with an independent secret key. Some studies propose to use threshold secret sharing to maintain

Volume 5 Issue 6, June 2016

www.ijsr.net

Licensed Under Creative Commons Attribution CC BY

the robustness of key management. However, the above studies do not consider deduplication. In traditional encryption different users use their own secret keys for encryption resulting in different cipher texts thus making deduplication impossible [2]. Convergent encryption ensures data privacy and enables deduplication. Bellare et al. [3] formalize this primitive as message-locked encryption, and explore its application in space-efficient secure outsourced storage. In the baseline approach the convergent encryption is based on a layered approach. In this the user key is encrypted by using a master key and it is stored along with the data in cloud. Baseline approach [6] suffers critical problems. It is inefficient in the aspect that it generates enormous number of keys with increasing number of users.

Message-Locked Encryption (MLE) [3] [so named because the message is locked, as it were, under itself] with the goal of providing an encryption primitive that provably enables secure deduplication. The key generation algorithm of an MLE scheme K maps a message M to a key K . The encryption algorithm E takes input the key K and a message M and produces a ciphertext C . The decryption algorithm D allows recovery of M from C given the key K . The tagging algorithm T maps the ciphertext C to a tag T used by the server to detect duplicates. (Tag correctness requires that tags corresponding to messages M_1, M_2 are likely to be the same iff M_1, M_2 are the same.) . All algorithms may depend on a parameter P but the latter is public and common to all parties including the adversary, and thus is not a key. Any MLE scheme enables deduplication of ciphertexts. CE is captured by our syntax as the MLE scheme that lets $K = H(M)$, $C = E(K, M)$ and tag $T = H(C)$. MLE is trivially achieved by letting the key K equal the message M . (Set $C = T = "$ to the empty string and have decryption simply return the key.) This degenerate solution is however useless for deduplication since the client stores as K the entire file and no storage savings result. We rule it out by requiring that keys be shorter than messages, ideally keys are of a fixed, short length.

Fortunately there are a lot of method for image compression. JPEG compression is a widely used form of lossy image compression that centers around the Discrete Cosine Transform [4]. Due to the increasing requirements for transmission of images in computer, mobile environments, the research in the field of image compression has increased significantly. Image compression plays a crucial role in digital image processing, it is also very important for efficient transmission and storage of images. When we compute the number of bits per image resulting from typical sampling rates and quantization methods, we find that Image compression is needed. Therefore development of efficient techniques for image compression has become necessary[8]. Three closely connected components form a typical lossy image compression system, they are (a) Source Encoder (b) Quantizer and (c) Entropy Encoder. (a) Source Encoder (or Linear Transformer) It is aimed at decorrelating the input signal by transforming its representation in which the set of data values is sparse, thereby compacting the information content of the signal into smaller number of coefficients. a variety of linear transforms have been developed such as Discrete Cosine Transform (DCT), Discrete wavelet Transform (DWT), Discrete Fourier Transform (DFT). (b)

Quantizer: A quantizer aims at reducing the number of bits needed to store transformed coefficients by reducing the precision of those values. Quantization performs on each individual coefficient i.e. Scalar Quantization (SQ) or it performs on a group of coefficients together i.e. Vector Quantization (VQ). (c) Entropy Coding Entropy encoding removes redundancy by removing repeated bit patterns in the output of the Quantizer. the most common entropy coders are the Huffman Coding, Arithmetic Coding, Run Length Encoding (RLE) and Lempel-Ziv (LZ) algorithm.

3. Proposed System

A. System model

First we develop a data outsourcing model by using Dekey. It consists of three entities. User, Storage- cloud Service Provider (S-CSP), Key management Cloud Service Provider (KMCSPP).

User: it is an entity who wish to outsource the data to the cloud and retrieve it later. To save upload bandwidth the user only upload unique data.

S-CSP: Provides the data outsourcing service and stores the data on behalf of the user.

KMCSPP: Provide key management by maintaining the convergent keys.

In this we specify two types of deduplication. File level deduplication that identifies duplicate files and block level deduplication in which the file is divided into fixed size blocks and avoid uploading of redundant blocks[5].

4. Pseudo Code

A. File Upload

To upload a file F first file level deduplication is to be performed.

Step 1: On input file F , the user computes and sends the block tags $T(B_i) = \text{TagGen}_{CE}(B_i)$ and file tag (TagGen_{CE}) to the S-CSP.

Step 2: Upon receiving, the S-CSP checks whether there exists the same tag on the S-CSP. If so, the S-CSP replies the user with a response "file duplicate," or "no file duplicate" otherwise.

Step 3: Upon receiving results for a block B_i returned from KM-CSPs, if it is a valid pointer, the user stores it locally; otherwise the user computes the secret shares K_1, K_2 ; then sends the share K_i and $T_j(B_i) = \text{TagGen}_{CE}(B_i, j)$ to the j -th KM-CSP for $j = 1; 2; \dots n$

Step 4: Upon receiving K_{ij} and $T_j(B_i)$, the j -th KM-CSP stores them and sends back the pointer for K_{ij} to the user for future access.

Step 5: Embed the data in an image.

Step 6: Stores the data to the cloud.

B. File Download

To decrypt the data first the user has to obtain the key shares from KMCSPP and recover the image from S-CSP and extract it. After extracting it decrypt the cipher text using the obtained key.

C. Image Upload and Download

To upload an image DCT image compression is to be applied.

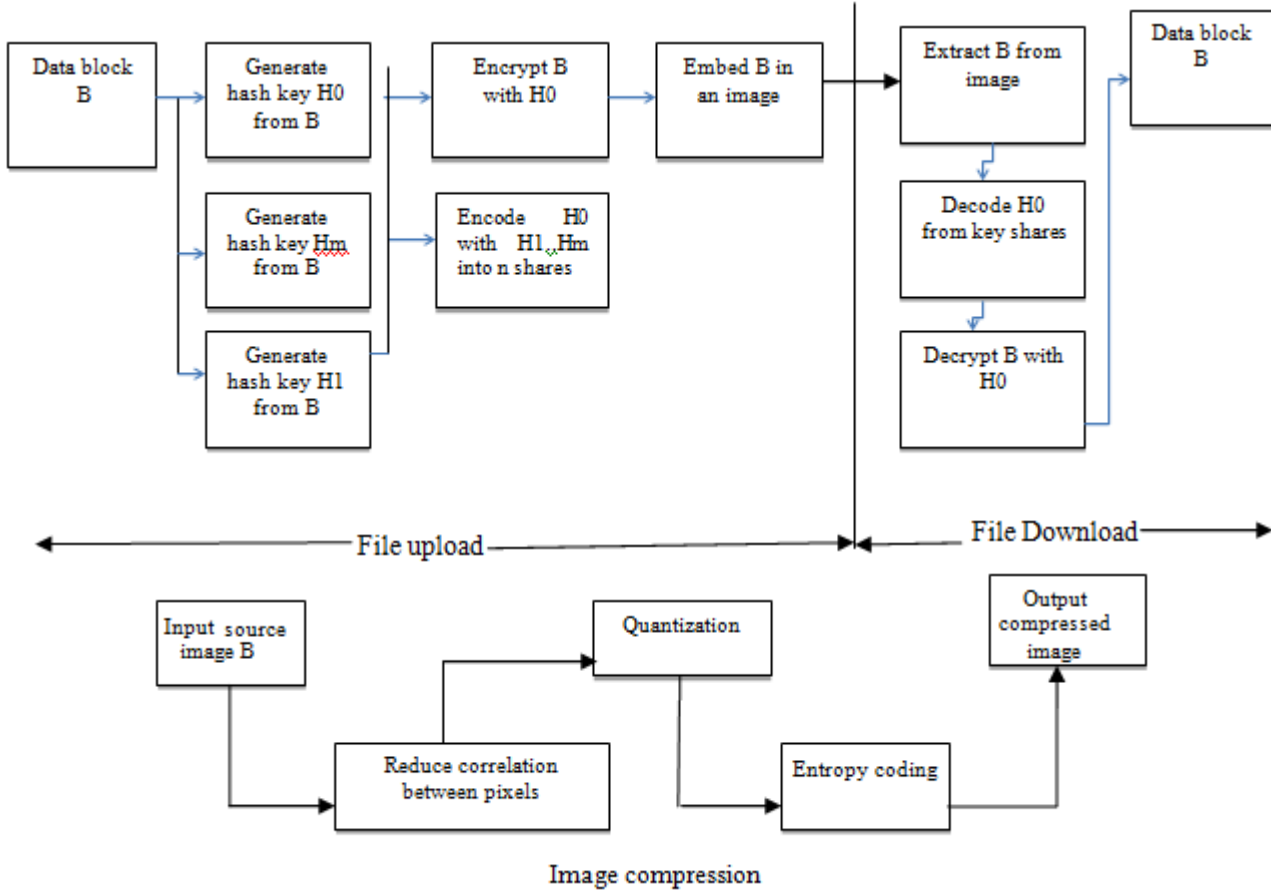
Step 1: image is broken into 8x8 blocks of pixels

Step 2: Working from left to right top to bottom DCT is applied to each block

Step 3: Each block is compressed using quantization.

Step 4: the array of compressed blocks that constitute the image is stored in a drastically reduced space.

Step 5: When desired the image is reconstructed by decompression, a process that uses Inverse Discrete Cosine Transform



5. Performance Analysis.

The performance can be analyzed on the basis of security it provides. Dekey is designed to solve the key management problem in secure deduplication where the files have been encrypted by utilizing convergent encryption and then embedding in an image.

6. Conclusion and Future Work

We propose Dekey, an efficient and reliable convergent key management scheme for secure deduplication. Dekey 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 implement Dekey using visual cryptography and demonstrate that it incurs small encoding/decoding overhead compared to the network transmission overhead in the regular upload/download operations. To enforce more security data is embedded in an image. To reduce storage space DCT compression is used.

References

[1] Jin Li, Xiaofeng Chen, Mingqiang Li, Jingwei Li, Patrick P.C. Lee, and Wenjing Lou Secure

Deduplication with Efficient and Reliable Convergent Key Management IEEE transactions on parallel and distributed systems, vol. 25, no. 6, June 2014

[2] P. Anderson and L. Zhang, "Fast and Secure Laptop Backups with Encrypted De-Duplication," in Proc. USENIX LISA, 2010, pp. 8.

[3] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-Locked Encryption and Secure Deduplication," in Proc. IACR Cryptology ePrint Archive, 2012, pp. 296-3122012:631.

[4] Ken Cabeen and Peter Gent, "Image Compression And Discrete Cosine Transform", Math 45 college of REDWOODS. S. Kamara and K. Lauter, "Cryptographic Cloud Storage," in Proc. Financial Cryptography: Workshop Real-Protocols Standardization, 2010, pp. 136-149

[5] M. Mulazzani, S. Schrittwieser, M. Leithner, M. Huber, and E. Weippl, "Dark Clouds on the Horizon: Using Cloud Storage as Attack Vector and Online Slack Space," in Proc. USENIX Security, 2011, p. 5.

[6] M.W. Storer, K. Greenan, D.D.E. Long, and E.L. Miller, "Secure Data Deduplication," in Proc. StorageSS, 2008, pp. 1-10.

[7] A.M.Raid1, W.M.Khedr2, M. A. El-dosuky1 and Wesam Ahmed1 "Jpeg Image Compression Using Discrete Cosine Transform" - A Survey. International

Journal of Computer Science & Engineering Survey
(IJCSES) Vol.5, No.2, April 2014.

- [8] Mark W. Storer Kevin Greenan Darrell D. E. Long
Ethan L. Miller “Secure Data Deduplication” Storage
Systems Research Center University of California,
Santa Cruz.