JOURNAL OF INFORMATION SECURITY AND APPLICATIONS 27-28 (2016) 54–64

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/jisa

Secure image deduplication through image compression

Fatema Rashid, Ali Miri, Isaac Woungang *

Department of Computer Science, Ryerson University, Toronto, Ontario, Canada

ARTICLE INFO

Article history:
Available online 5 January 2016

Keywords:
Image deduplication
Cloud storage provider
Data security
SPIHT image compression algorithm
Image hashing
Partial encryption

ABSTRACT

Cloud technology has empowered its users with remarkable expediency in terms of unlimited storage, accessibility, and availability of data. It has also contributed to the tremendous growth of digital data. To encounter this frenziedly growth, data deduplication has become a central approach for Cloud Storage Providers (CSPs) since it allows them to remove the identical data from their storages successfully. Currently, images are among the most common shared types of data found on cloud storages, and they are a good candidate for deduplication. In this paper, a novel compression scheme is proposed that achieves a secure deduplication of images in the cloud storages. Its design consists of embedding a partial encryption and a unique image hashing into the Set Partitioning In Hierarchical Trees (SPIHT) compression algorithm. The partial encryption scheme is meant to ensure the security of the proposed scheme against a semi honest CSP whereas the image hashing scheme is meant for classifying the identical compressed and encrypted images so that deduplication can be performed on them, resulting to a secure deduplication strategy with no extra computational overhead incurred for image encryption, hashing and deduplication. Experimental results and security analysis are provided to validate the stated goals.

1. Introduction

With the advent of cloud computing and its digital storage services, the growth of digital content has become irrepressible at both the enterprise and individual levels. According to the EMC Digital Universe Study (EMC, 2014), the global data supply reached 2.8 trillion GB in 2012, but just 0.5% of it was used for various kind of analysis purposes. The same study has also revealed that volumes of data are projected to reach about 5247 GB per person by 2020. Due to this explosive growth of digital data, there is a clear demand from CSPs for more cost effective use of their storage and network bandwidth for data transfer. In the recent years, data deduplication (ComputerWorld, 2013a) has been advocated as a promising and effective technique to bank the digital space by removing the duplicated data from the data centers or clouds.

Data deduplication can be performed at two levels: single user and cross-user levels. At the single user level, only the duplicate data saved by that specific user are deduplicated. On the other hand, at the cross user level, the data belonging to one user are matched with the data of all other users in order to identify any duplicates. Even though cross-user data deduplication often generates higher deduplication ratios compared to single user data deduplication, it has been reported to be more attractive to CSPs in terms of storage cost (Stanek et al., 2013). Independent of the considered deduplication level, it should be stressed that data deduplication can be performed either at the client side or at the server side. In the server side deduplication, the users are allowed to upload their

* Corresponding author. Department of Computer Science, Ryerson University, Toronto, Ontario, Canada. Tel.: 416 979 5000; fax: 416 979 5064.
E-mail addresses: fatema.rashid@ryerson.ca (F. Rashid), Ali.Miri@ryerson.ca (A. Miri), iwoungan@ryerson.ca (I. Woungang).
http://dx.doi.org/10.1016/j.jisa.2015.11.003

data in the cloud, and thereafter, the CSP will identify the similar data in its cloud storage. On the other hand, in the client side deduplication option, the similar data are identified at the client side without sending it entirely to the cloud. The server side deduplication approach generally incurs less computational cost (but high bandwidth requirements) compared to the client side deduplication option.

In view of the above discussed approaches for data deduplication and considering the advantages offered by CSPs to corporate and private users in terms of outsourcing their data, the security of data has become an increasingly prominent requirement from both the CSP and user perspectives. The users of the cloud storage services loose the control of their data once these are uploaded in the cloud since they no longer have a physical access to them. Therefore, one of the main security challenges is how can these users ensure that their data are kept safely in the cloud storage without tempering, modification or threats from malicious users. This security concern is even more pronounced when cross user deduplication is considered since only one unique copy of the redundant data will be allowed to be kept in the cloud storage and all its duplicates will be removed.

In an attempt to address these concerns, our objective has been to design a deduplication scheme where the users can encrypt their data and make it secure against the CSP, who is assumed to be semi-honest and who is supposed to act according to a prescribed protocol. For instance, the CSP is not expected to tamper the data stored in the cloud storage, but at the same time, he/she might be curious to identify the data in the storage, in order to gain some additional information. For this reason, the CSP cannot be completely trusted, and therefore the data must be encrypted. The CSP is trusted by the data owner to only return the requested encrypted files to the users upon request. The data and the encryption keys are both encrypted and kept secret from the CSP. It is also assumed that the user whose data are being deduplicated will not collude with other users, otherwise, the CSP will have access to all of the decrypted data. Based on these assumptions, some of our earlier works are described as follows.

In Rashid et al. (2012), a privacy-preserving scheme for securing the data deduplication in the cloud storage, where data are textual file formats, is proposed, with focus on cross-user client side deduplication. The proposed scheme consists of a deduplication algorithm — meant to divide a given textual file into smaller units, a combination of a secure hash function and block encryption algorithm — employed by the user to encrypt these units, and a asymmetric search encryption scheme — employed by the user to generate the index tree of hash values of these units, which then enables the CSP (assumed to be semi-honest) to search through the index and retrieve the requested units. This approach has the merit of allowing the CSPs to employ data deduplication techniques without the need for access to either the user plaintext or decryption key.

In Rashid et al. (2013a), a two level data deduplication scheme is introduced, which can be used in cloud storage by enterprises. At the enterprise level, cross-user data deduplication is performed and the data are outsourced to the cloud storage. At the CSP level (also assumed to be semi-honest), cross-enterprise data deduplication is performed by the CSP to remove the duplicates. The proposed scheme consists of (1) a secure indexing scheme — to index the data and its metadata in such a way that a complete data privacy is ensured against the semi-honest CSP; (2) a multi-user private keyword searchable encryption on the encrypted data while ensuring that the CSP does not have access to the searches and resulting files; and (3) a developed strategy to enable data sharing between the users based on existing metadata, indexing structures, and the searchable encryption scheme.

In Rashid et al. (2013b), a privacy-preserving scheme for assuring the data integrity in terms of proof of retrievability and proof of ownership in the context of cross-user client-side data deduplication for medium-sized and small-sized enterprises is proposed. Following the same trend, in Rashid et al. (2014), a scheme for secure image storage in the cloud is proposed, which consists of the Set Partitioning In Hierarchical Trees (SPIHT) image compression algorithm (SPIHT) — meant to classify the important and unimportant parts of the compressed data; a proof of retrieval (POR) scheme — probed by the user to the CSP (assumed to be semi-honest); and a proof of ownership (POW) scheme — probed by the CSP to the user. The POR scheme is meant to allow the users to ensure the security and privacy of their images residing in the cloud storage, by enabling them to identify their data and recover them completely using some error correcting codes even in case some minor corruptions have been injected into the data. On the other hand, the POW scheme is meant to enable the CSP to verify that the user of a requested file is indeed the true owner of that file.

Building upon the above works (Rashid et al., 2012, 2013a, 2013b), this paper also focuses on image deduplication since images constitute a huge portion of the digital data of the users, which are expected to be replicated in a huge number (ComputerWorld, 2013b). On the other hand, from a storage saving perspective, image deduplication is a desired requirement for both the end users and the CSPs. A novel SPIHT-based secure image deduplication scheme in the cloud is proposed, assuming that the CSP is also semi-honest. The proposed scheme consists of a partial encryption scheme involving the SPIHT algorithm — which ensures the security against the semi honest CSP in the sense that the compressed image resulting from the SPIHT algorithm is no more available in plaintext even to the CSP; and an image hashing scheme which also involves the SPIHT algorithm — which allows a classification of the identical compressed and encrypted images in such a way that deduplication can further be performed on these data. It should be noted that the use of the SPIHT algorithm is justified by the fact that in order to save storage space and bandwidth usage during transmission, images are often compressed enough before being uploaded to any storage.

To the best of our knowledge, the proposed scheme is the first attempt to explore image compression for the purpose of secured image data deduplication in the area of cloud storage services. It is also worth mentioning that in the proposed scheme, image compression is applied first, and then the resulting output is used for encryption and hashing purposes. This approach differs substantially from the previous works where image encryption or hashing is often applied first, and then the image is compressed for transmission purpose, which may lead to excessive computational overhead and more metadata to be stored.