Secure Image Deduplication in Cloud Storage

Han Gang[1], Hongyang Yan[2(✉)], and Lingling Xu[3]

[1] Admission Office of the Graduate School, Jinan University, Guangzhou, China
[2] School of Mathematics and Information Science, Guangzhou University, Guangzhou, China
Hyang.Yan@foxmail.com
[3] School of Computer Science and Engineering, South China University of Technology, Guangzhou, China
18826463520@163.com

Abstract. With the great development of cloud computing in recent years, the explosive increasing of image data, the mass of information storage, and the application demands for high availability of data, network backup is facing an unprecedented challenge. Image deduplication technology is proposed to reduce the storage space and costs. To protect the confidentiality of the image, the notion of convergent encryption has been proposed. In the deduplication system, the image will be encrypted/decrypted with a convergent encryption key which is derived by computing the hash value of the image content. It means that identical image copies will generate the same ciphertext, which used to check the duplicate image copy. Security analysis makes sure that this system is secure.

Keywords: Cloud computing · Image deduplication · Cloud storage · Security

1 Introduction

With the great development of cloud computing in recent years, the application of information and communication in the Internet has drew more and more attentions. Cloud computing [1] is a new computing paradigm with the dynamic extension ability, through the Internet to on-demand and extensible way of obtaining computing resources and services. It attracts much concern because of its unique technique and the emerging business computing model from the academic and industry.

However, with the explosive increasing of data, the mass of information storage, and the application demands for high availability of data, network backup is facing an unprecedented challenge. On the one hand, human society produce the data information from the Internet. On the other hand, we get the information from the daily production and kinds of scientific experiments (e.g. scientific computing and simulation, flight dynamics, a nuclear blast simulation, space exploration, and medical image data.). The growth of data information produced each day is to the impressive degree. According to the resent analysis

I. Khalil et al. (Eds.): ICT-EurAsia 2015 and CONFENIS 2015, LNCS 9357, pp. 243–251, 2015.
DOI: 10.1007/978-3-319-24315-3_25

report of IDC (International Data Corporation), the whole world produced 281 EB data in 2007, it is corresponded to everyone in the world owns 45 GB data. The world produced the amount of data will be close to 1800 EB, it is ten times than the amount of data in 2006 [2]. And the volume of data in the world is expected to reach 40 trillion GB in 2020 [3].

For above situation, data deduplication technology is proposed recently. Data deduplication technology [4] is a lossless data compression technology, mainly based on the principle of repeated data will be delete. This technology could reduce the cost of data transmission and storage [5]. Especially the image files in the social network, in most cases a celebrity public a message, it will be forwarded more than one thousand times soon. And popular images are also repeated many times. If such store operations occur every time, it certainly will cause waste of storage space. So simple to increase storage capacity does not solve the problem. Image deduplication have to be applied to the social network.

To protect the confidentiality of the image, the notion of convergent encryption [6] has been proposed. In the deduplication system, the image will be encrypted/decrypted with a convergent encryption key which is derived by computing the hash value of the image content [6–8]. It means that identical image copies will generate the same ciphertext, which allows the cloud storage server perform deduplication on the ciphertexts. Furthermore, image user make use of attribute-based encryption scheme to share images with friends by setting the access privileges.

In the rest of the paper is organized as follows. We introduce related work about deduplication in Sect. 2. Some preliminary works are introduced in Sect. 3. The architecture of image deduplication cloud storage system including security analysis will be described in Sect. 4. Finally, we include this paper in Sect. 5.

2 Related Work

There have been a number of deduplication technologies proposed recently. Most researchers focuss on text deduplication like [9]. They proposed a scheme to address the key management in deduplication system. There are different ways according techniques.

Technique based on the file-level deduplication is to delete the same file to reduce the data storage capacity, save storage space. It uses a hash function for each file to compute a hash value. Any two files with the same hash value is considered to be the same file. For example, SIS [10], FarSite [11], EMC Center [12] systems use this method.

Technique based on the block-level deduplication is to delete the same data block to reduce storage space [13]. This method is to divide a file into some data blocks [14], and uses hash functions compute the hash value, which be named as block fingerprint. Any two data block with the same block fingerprint are defined duplicate data block [15].

Based on the deduplication delete time, deduplication technology could divided to on-line deduplication [16] and post-processing deduplication [17].

On-line deduplication is to delete the duplicate data before storing, the storage service always stores a unique data copy. Post6-processing deduplication needs additional storage buffer to realize delete repeated data.

Based on the deduplication delete place, it can be divided to client deduplication [18] and service deduplication [19]. Client deduplication is before transferring the data copy to cloud server, user check and delete duplicate data. Service deduplication is performing duplicate data check and delete with service's resource in cloud server.

However, multi-media data like images, videos are larger than text. So image deduplication is becoming more important. Researchers have pay attention to this field like [20]. We have to handle the images before uploading them to server, a general way is watermarking [21,22]. Compression technique save the space of cloud storage in some way, but deduplication will address this problem from the root.

3 Preliminaries

3.1 Bilinear Mapping

Definition 1. Let G_1, G_2 be two cyclic groups of the number q(q is a prime number.), g is a generating element of G_1, a bilinear map is a map $e : G_1 \times G_1 \rightarrow G_2$ which satisfies the following three properties:

- Bilinear: $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$ for all $a, b \in \mathbb{Z}_p$ and $g_1, g_2 \in G_1$.
- Non-degenerate: $e(g, g) \neq 1$
- computable: $e(g_1, g_2)$ can be computed effectively with an algorithm for all $g_1, g_2 \in G_1$. so e is an efficient bilinear mapping from G_1 to G_2.

3.2 Access Structure

Let P=$\{P_1, P_2, \cdots, P_n\}$ be a set of parties. A collection $\mathbb{A} \subseteq 2^P$ is monotone. If $\forall B, C$, if $B \in \mathbb{A}$ and $B \subseteq C$, then $C \in \mathbb{A}$. An access structure(respectively, monotone-access-structure) is a collection(respectively,monotone collection)$\mathbb{A} \subseteq 2^P \setminus \phi$. The set in $\mathbb{A}$ are called the authorized sets, and the sets not in $\mathbb{A}$ are called the unauthorized sets. In this context, the attributes decide the role of the parties. So the authorized sets of attributes are included in $\mathbb{A}$.

3.3 Convergent Encryption

Convergent encryption [6,23] provides image confidentiality in deduplication. Because it uses the image content to compute encryption Hash value as the image encryption key. It makes sure that the key is directly related to the image content. The encryption key will not be leak under the condition of no leaking of the content of the image. And at the same time, because of the one-way operation of hash function, the image content will not be leaked when the key is

leaked. Above all, it also can ensure the ciphertext is only related to the image content, but has nothing to do with the user.

In addition, we have to compute a tag to support deduplication for the image and use it to detect duplicate copy in the cloud storage server. If two image copies are the same, then their tags are the same. The user first sends the tag to the cloud storage server to check if the image copy has been already stored. We can not guess the convergent key in terms of the tag because they are derived independently. In a general way, the convergent encryption scheme has four primitive functions:

- $KeyGen_{CE}(M) \rightarrow K_M$ This algorithm computes a convergent key K_M which maps an image copy M to a key.
- $Encrypt_{CE}(K_M, M) \rightarrow C$ This algorithm uses symmetric encryption algorithm outputs a ciphertext C, with taking both the convergent key K_M and the image copy M as inputs.
- $TagGen_{CE}(M) \rightarrow T(M)$ This is the tag generation algorithm that maps the image copy M to a tag $T(M)$. We make $TagGen_{CE}$ to generate a tag from the corresponding ciphertext as index, by using $T(M) = TagGen_{CE}(C)$, where $C = Encrypt_{CE}(K_M, M)$.
- $Decrypt_{CE}(K_M, C) \rightarrow M$ This is the decryption algorithm which outputs the original image M, with taking both the convergent key K_M and the ciphertext C as inputs.

3.4 KP-ABE Scheme

This scheme is used to encrypt the K_M, which computed from image content called convergent key. As the same time we delete the duplicate copy, the image owner wants some other friends access this image file. In key-policy ABE(KP-ABE) scheme [24,25], the access policy is embedded into the decryption key. The image owner signs the K_M ciphertexts with a set of attributes, when a user wants to access the image, the cloud storage server judges the user's attribute and decides which type of ciphertexts the key can decrypt.

We show the KP-ABE scheme by the following four polynomial algorithms.

- Setup(1^n) $\longrightarrow$ (parameters,msk): The probabilistic polynomial time(PPT) algorithm takes a security parameter n as input. It outputs the public parameters and the master secret key(msk) which is known only to the trusted the cloud storage server.
- Encrypt(m, parameters, μ)$\longrightarrow$ c: The PPT encryption algorithm takes as a input with a message m, the public parameters and a set of attributes mu. It outputs the ciphertext c.
- KeyGen(parameters, msk, $\mathbb{A}$) $\longrightarrow$ SK_w: The PPT key generation algorithm takes as a input with the public parameters, the master secret key and an access structure $\mathbb{A}$. It outputs the decryption key $D_{\mathbb{A}}$.
- Decrypt(parameters, c, $D_{\mathbb{A}}$) $\longrightarrow$ m or $\perp$: The Decryption algorithm takes as a input with c, the public parameters and the decryption key. It outputs the message m if $\mu \in \mathbb{A}$ or else it outputs an error message.

Here we note that the convergent key k_M is seen as the message m in this paper.

3.5 Proof of Ownership

Proof of ownership [26] is a protocol to be used to prove the user indeed has the image to the cloud storage server. This is to solve the problem of using a small hash value as a proxy for the whole image in client side deduplication. In order to describe the proof of ownership in details, we suppose a prover (i.e. a user) and a verifier (i.e. the cloud storage server). The verifier derives a short value $\phi(M)$ from an image copy M. And the prover needs to send ϕ' and run a proof algorithm to prove the ownership of the image copy M. It is passed if and only if $\phi' = \phi(M)$.

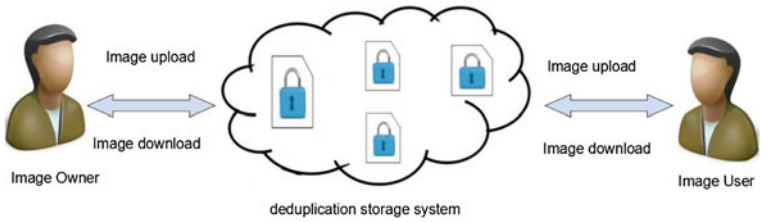

Fig. 1. Deduplication cloud storage system

4 Architecture of Image Deduplication System

4.1 System Participants

In this paper, we consider a deduplication cloud system consisting of image owner, image user, cloud service provider. The image is assumed to be encrypted by the image owner before uploading to the cloud storage server. We assume the authorization between the image owner and users is appropriately done with some authentication and key-issuing protocols. After uploading the encrypted image to the cloud server, image users who are authorized could access the encrypted image. In more details, an authorized image user send a request to the cloud storage server, the server will verify the proof of ownership. The image user needs to send ϕ' and run a proof algorithm to prove the ownership of the image copy M. It is passed if and only if $\phi' = \phi(M)$. It is passed if and only if $\phi' = \phi(M)$.

– **Image Owner.** The image owner is an entity that send the image to the cloud service to storage, share and access again. In order to protect the image content, the owner have to encrypt the image before uploading to the cloud.

In a client side image deduplication system, only the first image owner could store in the cloud. If it is not the first one, then the storage server will tell the owner this image is duplicate. So there is only one image copy in the cloud storage.

- **Image User.** The image user is an entity that has privileges to access the same image by passing the proof of ownership in the deduplication cloud system. And image user also includes the friends of image owner who shared the image resource in the cloud storage.
- **Deduplication Cloud Service Provider.** The entity of deduplication cloud storage server provides the image storage service for the image owners and users. Moreover, the cloud storage server will also play the role of performing duplicate image before users upload their images. The users couldn't upload the image again if there is an identical content image stored in the cloud storage server, and then they will get the privileges of accessing the same image by using the proof of ownership.

4.2 Deduplication Cloud System

Figure 1 shows the participants of deduplication cloud system and the specific work process. It goes as follows:

- **System Setup:** Define the security parameter 1^{λ} and initialize the convergent encryption scheme. We assume that there are N encrypted images $C = (C_{M_1}, C_{M_2}, \cdots, C_{M_N})$ stored in the cloud server by a user. Then we could compute $K_M = H_0(M)$ and $C_M = Enc_{CE}(K_M, M)$. The user also could compute a tag $T_M = H(C)$ for duplicate check.
- **Image Upload:** Before uploading an image M, the user interacts with the cloud server and use the tag to check if there is any duplicate copy stored in the cloud storage server. The image tag will be computed $T_M = H(C)$ to check the duplicate image. If the image is the first time to upload, then the cloud storage server will receive the image ciphertext. At the same time, image owner could set the attributes to control access privileges.
  - If there is a duplicate copy founded in the storage server, the user will be asked to verify the proof of ownership, if the user pass, then he will be assigned a pointer, which allows him to access the image. In details, the image user needs to send ϕ' and run a proof algorithm to prove the ownership of the image copy M. It is passed if and only if $\phi' = \phi(M)$. It is passed if and only if $\phi' = \phi(M)$. By using proof of ownership, users have privileges to access the same image.
  - Otherwise, if there is no duplicate images in the storage server, the user computes the encrypted image $C_M = Enc_{CE}(k_M, M)$ with the convergent key $K_M = H_0(M)$, and uploads C_M to the cloud server. The user also encrypts the convergent key K_M with attributes for setting the access privileges. He will get the $C_{K_M} = Enc(sk, K_M)$ also be uploaded to the cloud server.

– **Image Retrieve:** Supposing that a user wants to download an image M. He first sends a request and the image names to the cloud storage server. When the cloud storage server receive the request and the image name, it will check whether the user is eligible to download the files. If pass, the cloud server returns the ciphertext C_M and C_{K_M} to the user. The user decrypts and gets the key K_M by using sk which stored locally. If the user's attributes match the owner setting, then the cloud storage server will send the corresponding sk. With the convergent encryption key, the user could recover the original images. If failed, the cloud storage server will send an abort signal to user to explain the download failure.

4.3 Security Analysis

In this section, we present the security analysis for the deduplication cloud system.

– Confidentiality: The image user stored in the cloud will not be read because the image have to be encrypted to $C_M = Enc_{CE}(K_M, M)$ with the convergent key $K_M = H_0(M)$. Therefore, we couldn't get the content of the image which stored in the cloud from a ciphertext.
– Privacy protection: Because it uses the image content to compute encryption Hash value as the image encryption key. It makes sure that the key is directly related to the image content, it will not be leak under the condition of no leaking of the contents of the image. And at the same time, because of the one-way operation of hash function, the image content will not be leaked when the key is leaked. Above all, it also can ensure the ciphertext is only related to the image content, but has nothing to do with the user. Therefore, it can protect the privacy of users as more as possible.
– Completeness: We suppose that if the images have been successfully uploaded to the cloud server, the image owner can retrieve them from the cloud storage server and decrypt the ciphertext by using the correct convergent encryption key. Furthermore, a user who has the same image wants to upload to the cloud server, will perform the proof of ownership and get the privilege to access the stored image.

5 Conclusion

In this paper, we propose the image deduplication cloud storage system. To protect the confidentiality of sensitive image content, the convergent encryption has been used while supporting image deduplication. Owner could download the ciphertext again and retrieve the image with secret key, as the same time, image owner makes use of attribute-based encryption scheme to share images with friends by setting the access privileges. A user who has the same image copy could get the privilege to access the ciphertext by passing the proof of ownership and delete his duplicate copy. If a user's attributes match the owner's

setting access control, then he also could download the images. Security analysis makes sure that this system is secure in confidentiality, privacy protection and completeness.

Acknowledgement. This paper is supported by Fundamental Research Funds for the Central Universities (South China University of Technology) (No. 2014ZM0032), the Guangzhou Zhujiang Science and Technology Future Fellow Fund (Grant No. 2012J2200094), and Distinguished Young Scholars Fund of Department of Education (No. Yq2013126), Guangdong Province.

References

1. Mell, P., Grance, T.: The NIST Definition of Cloud, vol. 53, issue 6, p. 50 (2009)
2. Gantz, J.F., Ghute, C., Manfrediz, A., Minton, S., Reinsel, D., Schilchting, W., Toncheva, A.: The diverse and exploding digital universe: an updated forecast of world wide information growth through 2010. IDC white paper, pp. 2–16, March 2008
3. Gantz, J., Reinsel, D.: The digital universe in 2020: big data, bigger digital shadows, and biggest growth in the far east, December 2012. http://www.emc.com/collateral/analyst-reports/idc-the-digital-universe-in-2020.pdf
4. Asaro, T., Biggar, H.: Data De-duplication and Disk-to-Disk Backup Systems: Technical and Business Considerations, pp. 2–15. The Enterprise Strategy Group, July 2007
5. Tolia, N.: Using content addressable techniques to optimize client-server system. Doctoral thesis (2007)
6. Douceur, J.R., Adya, A., Bolosky, W.J., Simon, D., Theimer, M.: Reclaiming space from duplicate files in a serverless distributed file system. In: ICDCS, pp. 617–624 (2002)
7. Sabzevar, A.P., Sousa, J.P.: Authentication, authorisation and auditing for ubiquitous computing: a survey and vision. In: IJSSC, ser (2014). doi:10.1504/IJSSC.2011.039107
8. Yuriyama, M., Kushida, T.: Integrated cloud computing environment with it resources and sensor devices. In: IJSSC, ser (2014). doi:10.1504/IJSSC.2011.040342
9. Li, J., Chen, X., Li, M., Lee, P.P.C., Lou, W.: Secure deduplication with efficient and reliable convergent key management. IEEE Trans. Parallel Distrib. Syst. **25**, 1615–1625 (2013)
10. Bolosky, W.J., Corbin, S., Goebel, D.: Single instance storage in windows 2000. In: Proceedings of the 4th USENIX Windows Systems Symposium, pp. 13–24. Seattle, WA, USA (2000)
11. Adya, A., Bolosky, W.J., Castro, M.: Federated, available, and reliable storage for an incompletely trusted environment. In: Proceedings of the 5th Symposium on Operating Systems Design and Implementation, pp. 1–14. Boston, MA, USA (2002)
12. EMC Centera: Content Addressed Storage System. EMC CORPORATION (2003)
13. Policroniades, C., Pratt, L.: Alternative for detecting redundancy in storage systems data. In: Proceedings of the 2004 USENIX Annual Technical Conference, pp. 73–86. Boston, MA, USA (2004)
14. Rabin, M.O.: Fingerprinting by random polynomials. Technical report, Center for Research in Computing Technology, Harvard University (1981)

15. Henson, V.: An analysis of compare-by-hash. In: Proceedings of The 9th Workshop on Hot Topics in Operating Systems, pp. 13–18. Lihue, Hawaii, USA (2003)
16. Ungureanu, C., Atkin, B., Aranya, A.: A high-throughput file system for the HYDRAstor content-addressable storage system. In: Proceedings of the 8th USENIX Conference on File and Storage Technologies, pp. 225–238. San Jose, CA, USA (2010)
17. Clements, A.T., Ahmad, I., Vilayannur, M.: Decentralized deduplication in SAN cluster file systems. In: Proceedings of the 2009 USENIX Annual Technical Conference, pp. 101–114. San Diego, CA, USA (2009)
18. Fu, Y., Jiang, H., Xiao, N.: AA-Dedupe: an application-aware source deduplication approach for cloud backup services in the personal computing environment. In: Proceedings of the 2011 IEEE International Conference on Cluster Computing, pp. 112–120. Austin, TX, USA (2011)
19. Tan, Y., Feng, D., Zhou, G.: DAM: a data ownership-aware multi-layered deduplication scheme. In: Proceedings of the 6th IEEE International Conference on Networking. Architecture and Storage (NAS2010), pp. 403–411. Macau, China (2006)
20. Li, X., Li, J., Huang, F.: A secure cloud storage system supporting privacy-preserving fuzzy deduplication. Soft Computing (2015). doi:10.1007/s00500-015-1596-6
21. Pizzolante, R., Carpentieri, B., Castiglione, A.: A secure low complexity approach for compression and transmission of 3-D medical images. In: Broadband and Wireless Computing, Communication and Applications (BWCCA), pp. 387–392 (2013)
22. Pizzolante R., Castiglione A., Carpentieri B., De Santis A., Castiglione A.: Protection of microscopy images through digital watermarking techniques. In: Intelligent Networking and Collaborative Systems (INCoS), pp. 65–72 (2014)
23. Bellare, M., Keelveedhi, S., Ristenpart, T.: Message-locked encryption and secure deduplication. In: Proceedings of IACR Cryptology ePrint Archive, pp. 296–312 (2012)
24. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute based encryption for fine-grained access conrol of encrypted data. In: ACM conference on Computer and Communications Security, pp. 99–112 (2006)
25. Ostrovsky, R., Sahai, A., Waters, B.: Attribute-based encryption with non-monotonic access structures. In: the 14th ACM Conference on Computer and Communications Security, pp. 195–203 (2007)
26. Halevi, S., Harnik, D., Pinkas, B., Shulman-Peleg, A.: Proofs of ownership in remote storage systems. In: Chen, Y., Danezis, G., Shmatikov, V. (eds.) Proceedings of ACM Conference on Computer and Communication Security, pp. 491–500 (2011)