Towards an Image Encryption Scheme with Content-Based Image Retrieval Properties

Bernardo Ferreira, João Rodrigues, João Leitão, and Henrique Domingos

Nova University of Lisbon / NOVA-LINCS
Faculdade de Ciências e Tecnologia, 2829-516 Caparica, Portugal
bernardof@acm.org,jm.rodrigues@campus.fct.unl.pt
{jc.leitao,hj}@fct.unl.pt

Abstract. In this paper we introduce a new secure cryptographic scheme, named IES-CBIR, specifically designed for images and their outsourced storage and retrieval in large private image repositories. Our solution enables both encrypted storage and querying using Content Based Image Retrieval (CBIR), while preserving privacy. We have implemented a prototype system around the proposed scheme, and experimentally analyzed its performance when compared to similar proposals for privacy-preserving image retrieval. Our results show that IES-CBIR allows more efficient operations than existing proposals, both in terms of time and space overheads, while enabling less restrictive application scenarios.

Keywords: Data and Computation Outsourcing, Encrypted Data Processing, Privacy-Preserving Content-based Image Retrieval (CBIR)

1 Introduction

The amount of visual data being generated and shared by Internet users is growing everyday. The requirements for storing and sharing such large amounts of image data has been a key factor for the growth of data outsourcing solutions such as Cloud Computing and Storage services (e.g. Instagram, Flickr, etc.) [1]. Furthermore, in these services the ability to efficiently retrieve relevant fractions of the outsourced data comes of utter importance for usability.

Although data outsourcing seems like the perfect solution for supporting large scale image storage and retrieval systems, it actually raises new issues for users' privacy. On one hand, recent news have proven that privacy isn't preserved by outsourced storage providers [2,3]. On the other hand, honest yet curious or malicious system administrators working for the providers have access to all data on disk and memory in the hosting machines [4,5]. Finally, external hackers can exploit software vulnerabilities to gain unauthorized access to servers and their stored data.

The traditional solution to solve these issues and enforce users' privacy has been to outsource encrypted data and run computations on the users' devices, after it's transfer and decryption [6]. However, such approach limits it's own applicability, specially regarding online applications managing very large repositories. A promising approach is to perform operations on encrypted data, directly on the server side. Nonetheless, existing solutions are still of theoretical interest only (e.g. Fully Homomorphic Encryption [7]), or have complexity and scalability issues that limit their wide adoption [8,9], particularly for supporting private image retrieval over large-scale repositories.

To address these challenges we introduce IES-CBIR (Image Encryption Scheme with Content-Based Image Retrieval properties), a cryptographic scheme proposal that supports outsourcing of private storage and search / retrieval of images in the encrypted domain. Key to the design of this scheme is the observation that in images, color information can be separated from texture information, enabling the use of different encryption techniques with different properties for protecting each of these features. Following this observation, and considering that texture is usually more relevant than color for object recognition, in IES-CBIR we make the following tradeoff: we choose to protect image contents first, by encrypting texture information with probabilistic encryption; then we somewhat relax the security of the remaining features, by using deterministic encryption on image color information.

2 Related Work

The related work on privacy-preserving image retrieval can be divided in two classes: Searchable Symmetric Encryption (SSE) based approaches [9] and partially-homomorphic approaches [8].

SSE approaches force clients, before outsourcing a repository, to generate an index referencing it's images, and then separately encrypting both images and index and uploading them to the outsourced servers. Image encryption is typically performed using conventional symmetric cryptography. To allow privacy-preserving search over it, the index is encrypted with a symmetric-key scheme displaying some form of homomorphic property, such as determinism [10] or order-preservation [11]. An example of such work is [9], where different approaches are proposed to extract and encrypt indexing structures from image repositories, while preserving the ability to perform CBIR based on color features. Unfortunately, these SSE-based approaches present significative limitations: clients have ad-

ditional computational overhead, as they have to process images locally and extract their indexing structures; additional bandwidth consumption is required to move both the repositories and their indexes to the outsourced servers; and when a user adds images to his repository, he may be required to download image feature vectors and recompute their index, which exacerbates the previous drawbacks.

The alternatives to SSE that can be found in the literature are based on public-key partially-homomorphic encryption (PKHE) schemes such as Paillier [12] or ElGammal [13]. In these approaches clients encrypt images in a way that outsourcing servers can perform all image indexing and querying operations directly over the encrypted data, avoiding many of the practical issues of SSE-based solutions. Unfortunately, PKHE presents much higher time and space complexities when compared with SSE schemes. For instance, in [8] the authors design a powerful CBIR algorithm for the encrypted domain (based on SIFT [8]) by resorting to the Paillier cryptosystem [12], a PKHE which enables additions on the encrypted domain. However, their approach results in big ciphertext expansion and slow encryption and decryption times (as we will experimentally prove in our evaluation section 6).

With our proposal of IES-CBIR, we try to address the limitations of previous works, by aiming at a balance between cryptographic complexity and client's processing overhead in the presence of large, dynamically changing repositories.

3 System and Adversary Models

System Model The envisioned system model where IES-CBIR would be applied (Figure 1) considers 3 main entities: *Owners*, the entities that own images and outsource them to an externally managed repository, by encrypting and sending them to a third-party's infrastructure (the Store); *Store*, the entity responsible for storing images for the Owners and performing computations over them on their behalf, including feature vector extraction and indexing [14]; and the *Users*, which are entities authorized by one or more Owners to perform search operations over their repositories. Although being able to perform search operations, Users must still explicitly request an Owner for individual access to images that might interest them (i.e. returned by the Store in reply to their queries).

Owners have a trapdoor/search key per repository, which are shared with trusted Users to grant them search privileges over it; and multiple decryption/access keys (one per stored image) which are kept secret by

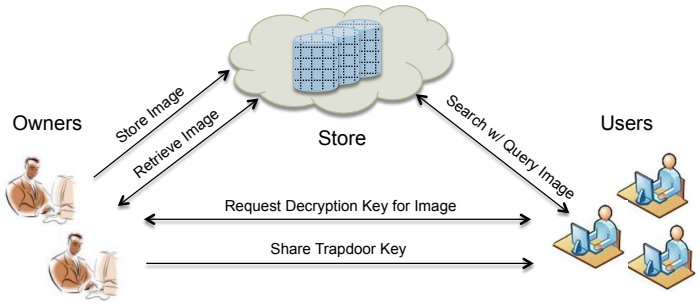

Fig. 1. System Model

the Owner, and only shared with a User if he wants to grant full access to a particular image. Key distribution between Owners and Users can be done by different mechanisms orthogonal to this paper.

Adversary Model In this work we aim at protecting the privacy of the Owner's images and User's query images. Attacks on privacy may come from curious or malicious administrators managing the Store's infrastructure, malicious Users that deviate from their expected behavior, or external Internet hackers. Malicious Store administrators may have access to all data stored on disk (at the Store), in RAM, and passing through the network. External malicious entities are assumed to only be able to access data passing through the network, or in the particular case of malicious Users, being able to issue search request over an Owner's repository (with or without access to the correct trapdoor key). Leveraging their capacities towards the system, the described adversaries will try to break any secure protocols or cryptographic schemes to gain access to the image contents. In this model, the Trusted Computing-Base is reduced to the Owners and Users personal infrastructures, and the Store's infrastructure and communication channels are considered as *not trusted*.

4 IES-CBIR Proposal

In this section we present our proposal of an Image Encryption Scheme with CBIR capabilities (IES-CBIR). We remind the reader that our scheme leverages the key observation that in images, color information can be separated from texture information and hence different levels of security can be applied when protecting each. Following this observation, and considering that texture is usually more relevant than color for object recognition, in IES-CBIR we protect image texture with probabilistic encryption and color information with deterministic encryption. Hence,

privacy-preserving CBIR based on color can be performed on outsourced servers (the Store), without intervention of clients (Owners and Users), while fully protecting image texture. IES-CBIR is composed of four main algorithms, which we present in the next paragraphs.

Key Generation IES-CBIR works with two different types of cryptographic keys, which are generated by two different algorithms: search/trapdoor keys (tk) and access/decryption keys (dk). Trapdoor keys tk are used to give search privileges to a group of Users, and one is generated for each Owner. Decryption keys dk are unique to each encrypted image stored in an Owner's repository, and are only shared with (trusted) Users to accept explicit access requests to individual images. Both keys are required in the encryption and decryption of images. However, to generate trapdoors for searching an Owner's repository, only tk is required. The key tk is generated by making a random permutation (through a Pseudo-Random Number Generator function (PRNG) parameterized with a random seed) of all values in the range [0..100] (represents all values in HSV color space). In contrast, key dk is generated by requesting a 128-bit key from a Symmetric-Key Generator function, that will be used as a cryptographic seed for the probabilistic encryption counterpart of IES-CBIR.

Encryption Image encryption in IES-CBIR is achieved through two steps: i) pixel color value encryption and ii) pixel rows and columns position shifting. The goal of the first is to protect image color contents, by deterministically replacing each pixel color value in each color channel. This is achieved by mapping each pixel color value to it's encrypted counterpart, according to key tk. To further protect image contents, we rely on a second probabilistic step: random pixel rows and columns shifting. This step consists in instantiating a PRNG function with a previously generated decryption key dk as cryptographic seed. Then, for each pixel column, we request from the PRNG a new random value v between 1 and the image height and do a shift on that pixel column of v positions. The procedure is then repeated for the rows (with random values ranging between 1 and the image width).

Decryption The decryption algorithm applies the different steps of encryption through the opposing order: rows shifting to their original positions, columns shifting, and then pixel color decryption. Note however that the random values must be generated in the same order as they were in encryption.

Trapdoor Generation This algorithm generates trapdoors that Users use to search over image repositories. Trapdoor generation requires a

query image (Q) as input, as well as a trapdoor key tk. Given a trapdoor key tk, the algorithm operates in a similar fashion to the encryption algorithm, where the decryption key dk is substituted by a User's randomly generated key.

5 CBIR in the Encrypted Domain

IES-CBIR enables content-based image retrieval tasks (for color features) to be performed on the encrypted domain without modification from their plaintext counterparts. As such, upon receiving new images for storage, the Store can process them, extract their feature vectors [14] and index them. Feature extraction consists in processing an image and extracting a reduced set of feature vectors that describe it. In this proposal we focus on color features in the HSV color model and their representation as color histograms. As such, for each encrypted image and each HSV color channel, the Store can build a color histogram representing it (yielding a total of 3 histograms per image).

After processing an Owner's encrypted images, the Store can receive search requests (trapdoors) from authorized Users. Upon receiving a trapdoor, the Store also extracts its feature vectors (the same way it did for the Owner's images) and finds the k most similar images in the repository by comparing their features vectors through histogram intersection [14]. After receiving these ranked results, Users can explicitly request full access to an image by asking its Owner for the decryption key.

6 Experimental Evaluation

In this section we perform an initial experimental evaluation to access the performance of a system prototype leveraging IES-CBIR, as well as similar systems leveraging competing alternatives found in the literature. In particular we compare the performance of our prototype with that of: (i) a system leveraging the Paillier cryptosystem described in [8] (labeled *PKHE*), and (ii) the two SSE-based solutions originally proposed in [9] (labelled *SSE OPE* and *SSE MinHash*). To conduct this experiment we measured the time taken by the *Store Image* operation with two distinct workloads: (i) One where 1000 JPEG images are simultaneously used to populate an outsourced repository, and another where after the initial upload of 1000 images, the Owner uploads another 10 different groups of 100 images per group. This last workload allow us to show hidden overheads of updating repositories in some alternatives described in the literature.

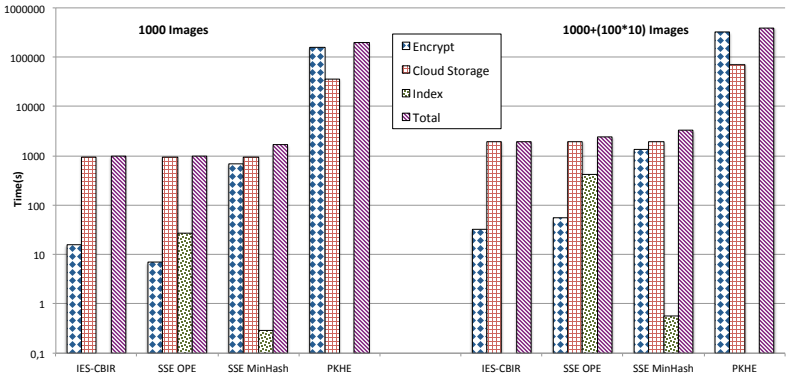

Fig. 2. Performance for the *Store Image* operation

These results present the performance from the perspective of users (i.e. we don't consider the computation in the outsourced infrastructure, as this can easily be scaled). In the experiments, client computations (i.e., Owners and Users) were performed on a Linux-Ubuntu Desktop with a Intel Core i3-2130 3.4Ghz processor, running a OpenJDK 7 JVM with 2 GB of reserved memory, located in Lisbon, while the Store processing was carried on a Amazon EC2 Medium instance (Ireland datacenter) with a limited Internet connection of 30 Mb/s download and 3 Mb/s upload between clients and store.

Figure 2 summarizes the obtained results for each system, in terms of time required for sub-operations (*Encryption*, *Indexing* and *Cloud Storage*) and as a whole (*Total*). Results show that *IES-CBIR* offers overall better performance when compared with the remaining competing alternatives. More specifically, in terms of client processing *IES-CBIR* only requires encryption, while also presenting one of the highest cryptographic throughput. Alternatives either rely on indexing operations (*SSE OPE*) or slower cryptography (*SSE MinHash* and *PKHE*).

Furthermore, when adding additional images to the repository, our solution is the one that offers overall best performance, specially when compared with *SSE OPE*. This happens because with IES-CBIR, the Owner isn't required to fetch image feature vectors, re-indexing them, and re-uploading the newly generated indexes to the Store. Finally, we also observe that IES-CBIR offers similar ciphertext expansion as *SSE OPE* and *MinHash* (a ratio of $1,00005\%$ compared to $1,379\%$ and $1,003\%$, respectively), and much lower than *PKHE* (a ratio of 37%). This is further exacerbated by the data upload time to the Store in the different systems.

7 Conclusions

In this paper we have presented a proposal for a new cryptographic scheme, named IES-CBIR, which supports private outsourcing of storage and search/retrieval of images in the encrypted domain. Key to the design of IES-CBIR is the observation that in images, color information can be separated from texture information, enabling the use of different encryption techniques with different properties for each one. Leveraging IES-CBIR, we designed a secure system model focused on dynamic and distributed image outsourcing. We experimentally validated the performance of our proposal by comparing an early system prototype leveraging IES-CBIR with the relevant related works. The results obtained show that our proposal is as efficient as the related works, promises better scalability and demands lower computational overhead from clients.

References

1. Global Web Index: Instagram tops the list of social network growth. http://blog.globalwebindex.net/instagram-tops-list-of-growth (2013)
2. Rushe, D.: Google: don't expect privacy when sending to Gmail. The Guardian. `http://tinyurl.com/kjga34x` (2013)
3. Greenwald, G., MacAskill, E.: NSA Prism program taps in to user data of Apple, Google and others. The Guardian. `http://tinyurl.com/oea3g8t` (2013)
4. Chen, A.: GCreep: Google Engineer Stalked Teens, Spied on Chats. Gawker. `http://gawker.com/5637234` (2010)
5. Halderman, J., Schoen, S.: Lest we remember: cold-boot attacks on encryption keys. Commun. ACM **52**(5) (2009) 91–98
6. Mahajan, P., et al.: Depot: Cloud Storage with Minimal Trust. ACM Trans. Comput. Syst. **29**(4) (December 2011) 1–38
7. Gentry, C., et al.: Homomorphic evaluation of the AES circuit. In: Adv. Cryptology–CRYPTO 2012. Springer (2012) 850–867
8. Hsu, C.Y., et al.: Image Feature Extraction in Encrypted Domain With Privacy-Preserving SIFT. IEEE Trans. Image Process. **21**(11) (2012) 4593–4607
9. Lu, W., et al.: Enabling Search over Encrypted Multimedia Databases. In: IS&T/SPIE Electron. Imaging, ISOP (February 2009) 725418–725418–11
10. Song, D.X., et al.: Practical techniques for searches on encrypted data. In: Proc. of IEEE S&P, IEEE (2000) 44–55
11. Agrawal, R., et al.: Order preserving encryption for numeric data. In: Proc. SIGMOD, ACM (2004) 563–574
12. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: EUROCRYPT'99. (1999) 223–238
13. ElGamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. In: Adv. Cryptol., Springer (1985) 10–18
14. Swain, M.J., Ballard, D.H.: Color indexing. Int. J. Comput. Vis. **7**(1) (1991) 11–32