Journal of Computer and System Sciences 80 (2014) 1549–1562

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

EFADS: Efficient, flexible and anonymous data sharing protocol for cloud computing with proxy re-encryption

Guiyi Wei [a], Rongxing Lu [b], Jun Shao [a,*]

[a] College of Computer and Information Engineering, Zhejiang Gongshang University, 310018, PR China
[b] School of Electrical and Electronics Engineering, Nanyang Technological University, 639798, Singapore

ARTICLE INFO

Article history:
Received 28 January 2013
Received in revised form 20 August 2013
Accepted 10 April 2014
Available online 23 April 2014

Keywords:
Flexible sharing
Cloud computing
Proxy re-encryption
Anonymity

ABSTRACT

The concept of cloud computing has emerged as the next generation of computing infrastructure to reduce the costs associated with the management of hardware and software resources. It is vital to its success that cloud computing is featured efficient, flexible and secure characteristics. In this paper, we propose an efficient and anonymous data sharing protocol with flexible sharing style, named EFADS, for outsourcing data onto the cloud. Through formal security analysis, we demonstrate that EFADS provides data confidentiality and data sharer's anonymity without requiring any fully-trusted party. From experimental results, we show that EFADS is more efficient than existing competing approaches. Furthermore, the proxy re-encryption scheme we propose in this paper may be independent of interests, i.e., compared to those previously reported proxy re-encryption schemes, the proposed scheme is the first pairing-free, anonymous and unidirectional proxy re-encryption scheme in the standard model.

1. Introduction

Cloud computing, as it can provide many flexible and efficient services, e.g., storage and computing services, has received considerable attention in recent years. However, the paradigm of cloud computing also brings many new challenges when the sensitive data are outsourced to the semi-trusted cloud server. In order to preserve the confidentiality of the data, files should be encrypted before being uploaded to the server. When a data holder wants to share his/her data to some sharer, he/she can send the ciphertext, denoted as K, of the data's corresponding decryption key(s) to the sharer, where K is computed by the sharer's public key. After that, the sharer can first obtain the decryption key(s) from the ciphertext K by using his/her private key, and then decrypt the corresponding data downloaded from the cloud server. Obviously, this method is simple and efficient. However, the sharing situation would be more complex in practice. The data held by the data holder may not be original one generated by himself/herself, but received from others with an encryption form under the data holder's public key.[1] In this case, the above sharing method cannot work any more, since the decryption key becomes the data holder's private key, which should be kept secret from others except the data holder. To avoid revealing his/her private key to the sharer, the data holder may firstly use the private key to obtain the data, and then encrypt the data by

* Corresponding author at: Room 420, 18# Xuezheng Street, SCIE building, Zhejiang Gongshang University, Hangzhou, Zhejiang Province, 310018, PR China. Fax: +86 571 28008303.
E-mail addresses: weigy@zjgsu.edu.cn (G. Wei), rxlu@ntu.edu.sg (R. Lu), chn.junshao@gmail.com (J. Shao).

[1] To improve the efficiency of encryption, the data is usually encrypted by a hybrid method. In particular, the data is firstly encrypted by a symmetric key encryption with a key, and then the key is encrypted by using the data holder's public key.

http://dx.doi.org/10.1016/j.jcss.2014.04.021
0022-0000/

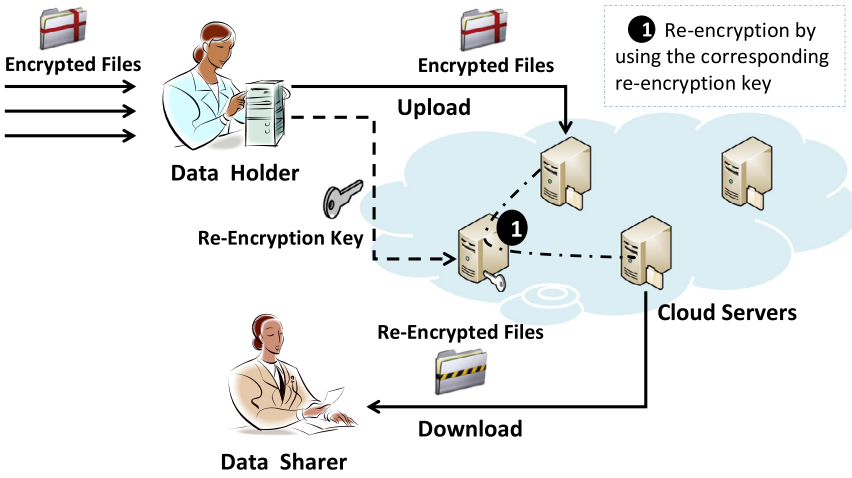

Fig. 1. Data sharing on cloud computing implemented by PRE.

using every sharer's public key. However, this method will bring a huge number of workload to the data holder. Hence, it is desirable to design a new sharing method on the could computing.

The concept of proxy re-encryption (PRE), proposed by Blaze et al. [8] at Eurocrypt 1998, could be a potential tool to solve the above problem. In such a scheme, a semi-trusted proxy with specific information (a.k.a., re-encryption key) can transform a ciphertext under Alice's (delegator's) public key into another ciphertext of the same plaintext under Bob's (delegatee's) public key. However, the proxy cannot decrypt any ciphertext under the public key of Alice or Bob. Conceptually, the data sharing on cloud computing without a trusted server implemented by PRE, as shown in Fig. 1, is as follows. Firstly, the data holder (delegator) generates re-encryption keys, and sends the keys to the re-encryption key pool. When the data sharer (delegatee) wants to retrieve the files, the sharer tells the cloud server which re-encryption key will be used. Secondly, the cloud server obtains the associated re-encryption key, and uses it to do the transformation. Thirdly, the cloud server sends the re-encrypted data to the sharer. Finally, the data sharer decrypts the encrypted data by using his/her private key.

It is easy to see that the PRE-based data sharing method allows the data holder not only to be free from the huge workload of decryption-then-encryption method, but also to keep his/her private key secret. Meanwhile, it also highly desires the user anonymity in the data sharing, i.e. the cloud server cannot extract a list of "the data is shared with whom". Because this list can hurt the secrecy of users' action, as well as the confidentiality of the content of files. In particular, the adversary may only focus on trying to decrypt encrypted files shared with a specific user. Note that the ciphertext in public key encryption may reveal the intended decryptor's identity.

The existing works [7,3,11,13,5,6,22] for the data sharing based on PRE or not, as discussed in Section 7, can only resolve the data confidentiality (some even require a fully-trusted party), and most of them does not support flexible sharing (i.e., the shared data is not the original one generated from the data holder, or the sharer list should be pre-decided) or user anonymity. In this paper, we address the problem of the flexible data sharing on semi-trusted servers with data confidentiality and user anonymity. Our proposed sharing protocol, named EFADS, has the following properties.

- Compared to previous PRE-based sharing protocols [5,6], EFADS provides the *data sharer's anonymity*. In particular, from the ciphertexts and proxy re-encryption keys stored in the cloud, the adversary (even corrupting the could server but not the data holder) cannot deduce the identities of the corresponding data sharers.
- Compared to previous data sharing protocols for cloud computing [22], EFADS provides *flexible data sharing style*. In particular, EFADS allows the data holder to share the data not only generated from himself/herself, but also received from others. Furthermore, it also allows the data holder to decide the sharer at any time.
- The proposed PRE scheme may be independent of interests. It is the first pairing-free, anonymous and unidirectional PRE scheme proven-secure in the standard model.

The remainder of this paper is organized as follows. In Section 2, we formalize the system model, security model, and identify our design goal. Then, we present the EFADS protocol in Section 4, followed by the security analysis and performance evaluation in Section 5 and Section 6, respectively. We also review some related works in Section 7. Finally, we draw our conclusions in Section 8.

2. Models and design goal

In this section, we formalize the system model, security model, and identify our design goal as well.