Future Generation Computer Systems 62 (2016) 85–91

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Cloud data integrity checking with an identity-based auditing mechanism from RSA

Yong Yu [a], Liang Xue [a], Man Ho Au [b,*], Willy Susilo [d], Jianbing Ni [a], Yafang Zhang [a], Athanasios V. Vasilakos [c], Jian Shen [e]

[a] Big Data Research Center, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China

[b] Department of Computing, The Hong Kong Polytechnic University, Hong Kong

[c] Lulea University of Technology, Sweden

[d] Center for Computer and Information Security Research, School of Computing and Information Technology, University of Wollongong, Wollongong, NSW 2522, Australia

[e] School of Computer and Software at Nanjing University of Information Science and Technology, Nanjing, China

HIGHLIGHTS

- We formalize the security requirement for identity-based cloud data integrity auditing mechanism.
- We provide a concrete construction of identity-based cloud data integrity checking protocol.
- We prove that our construction is secure under the well-known RSA assumption.

ARTICLE INFO

Article history:
Received 30 May 2015
Received in revised form
15 January 2016
Accepted 13 February 2016
Available online 3 March 2016

Keywords:
Cloud storage
Data integrity
Identity-based signature
Provable security

ABSTRACT

Cloud data auditing is extremely essential for securing cloud storage since it enables cloud users to verify the integrity of their outsourced data efficiently. The computation overheads on both the cloud server and the verifier can be significantly reduced by making use of data auditing because there is no necessity to retrieve the entire file but rather just use a spot checking technique. A number of cloud data auditing schemes have been proposed recently, but a majority of the proposals are based on Public Key Infrastructure (PKI). There are some drawbacks in these protocols: (1) It is mandatory to verify the validity of public key certificates before using any public key, which makes the verifier incur expensive computation cost. (2) Complex certificate management makes the whole protocol inefficient. To address the key management issues in cloud data auditing, in this paper, we propose ID-CDIC, an identity-based cloud data integrity checking protocol which can eliminate the complex certificate management in traditional cloud data integrity checking protocols. The proposed concrete construction from RSA signature can support variable-sized file blocks and public auditing. In addition, we provide a formal security model for ID-CDIC and prove the security of our construction under the RSA assumption with large public exponents in the random oracle model. We demonstrate the performance of our proposal by developing a prototype of the protocol. Implementation results show that the proposed ID-CDIC protocol is very practical and adoptable in real life.

1. Introduction

Cloud computing is a type of large-scale distributed computing paradigms which has become a driving force for Information and Communications Technology over the past several years, due to its innovative and promising vision. It provides the possibility of improving IT systems management and is changing the way in which hardware and software are designed, purchased, and utilized. There are a range of applications that can be delivered to users via cloud computing models, from content management to specialist applications for activities. Cloud computing provides nearly "infinite" and "ubiquitous" information service for cloud users. Among all the services provided by cloud computing, cloud

* Corresponding author.
E-mail address: csallen@comp.polyu.edu.hk (M.H. Au).

http://dx.doi.org/10.1016/j.future.2016.02.003
0167-739X/

storage is one of the most important services which allows cloud users to migrate their data from local storage systems to the cloud. Cloud storage service brings significant benefits to data owners, say, (1) reducing cloud users' burden of storage management and equipment maintenance, (2) avoiding investing a large amount of hardware and software, (3) enabling the data access independent of geographical position, (4) accessing data at anytime and from anywhere. Because cloud storage offers scalable pay as you go and location independent storage services for cloud users, it has become a quick profit growth point in cloud computing. According to a survey in [1], 79% of organizations prefer to choose IT outsourcing service.

However, cloud storage does trigger some new security threats to data owners. A number of cloud users would not like to use cloud storage due to some serious security worries. A primary concern of cloud users is the integrity of their outsourced files. There are a few factors that might lead to data corruption. First, cloud service providers are not fully trusted. As a result, for monetary reason, the cloud service provider might delete the data that are rarely or have not been accessed so that it can save the space for storing other files for charging extra expenses. Second, the stored data could be corrupted due to cloud server's failure, management errors or adversary attacks. However, in order to maintain a good reputation, cloud service provider may deliberately hide data loss events. In cloud storage, data integrity and leakage have become a primary concern of cloud users [2]. Recently, a series of cloud storage security incidents aggravate cloud users' worries. Take Amazon's cloud crash disaster as an example.[1] The Amazon's huge EC2 cloud services crash in 2011 permanently destroyed some data of cloud users. The data loss was apparently small relative to the total data stored, but anyone who runs a website can immediately understand how terrifying a prospect any data loss is. Sometimes it is insufficient to detect data corruption when accessing the data because it might be too late to recover the corrupted data. Therefore, it is becoming necessary for cloud users to frequently check if their outsourced data remain intact [3].

Unfortunately, once cloud users store their data on remote cloud servers, they lose the de-facto control over their data. Consequently, traditional cloud data integrity checking methods such as digital signatures, hash functions are not feasible. To solve this problem, in 2007, Ateniese et al. [4,5] proposed the notion of Provable Data Possession (PDP) and formalized the security model for this primitive. They also presented two highly effective and provably secure PDP schemes, which enable the cloud user to efficiently verify the integrity of their outsourced data without accessing the entire file. Meanwhile, Juels and Kaliski [6] proposed the concept of Proof of Retrievability (PoR) which enables the cloud server to generate a correct proof that the cloud user can retrieve the remote file, and they put forward a sentinel-based construction using pseudo-random sampling technique. As a state-of-the-art of the PoR protocols, in 2008, Shacham and Waters [7,8] presented two efficient and compact PoR schemes, which use error-correcting code technology. The construction of the first scheme is based on BLS signature [9], which satisfies the publicly verifiable, and its security can be proved in the random oracle model. The second scheme is constructed by using pseudo-random functions (PRFs). It is secure in the standard model. In recent years, a number of cloud data integrity checking protocols [10–20] were proposed for catering to different requirements of the integrity verification of cloud users' data.

The aforementioned schemes are based on complex public key infrastructure, where each user's public key is certified with a public key certificate issued by the certificate authority. One has to verify the legitimacy of public key certificate before using its public key, thus the computation and communication cost of the cloud user is heavy. To fix this drawback, Zhao et al. [21] proposed the first identity-based public verification scheme based on the identity-based aggregate signature proposed by Gentry [22]. Their privacy preserving method follows the random masking technology due to Wang et al. [12]. In addition, Wang et al. [23] proposed the notion of identity-based distributed provable data possession for multicloud storage. But the verification algorithm of this scheme requires user's additional secret information, i.e., a part of user's private key, and as a result, this proposal is not purely identity-based. Besides, the scheme [23] is not sound since even if the cloud server deletes the entire file, the server is still able to produce a valid proof to deceive verifiers only by making use of the hash values of the challenged blocks. Quite recently, Yu et al. [24] described a generic construction of identity-based PDP, derived from identity-based signatures and traditional PDP protocols. Their concrete construction is based on the short signature due to Boneh et al. [9].

Our contribution. The contributions of this paper can be summarized as follows:

1. We propose a concrete ID-CDIC construction from RSA signature, which supports variable-sized file blocks and public auditing. The new protocol is different from the previous schemes in [21,23].
2. We prove the soundness of the new ID-CDIC protocol in the improved soundness model in [24]. The security of our construction is based on the RSA assumption with large public exponent in the random oracle model.
3. We show the efficiency of the proposal by developing a prototype implementation of the protocol.

Paper Organization: The rest of the paper is organized as follows. In Section 2, we review some preliminaries of the ID-CDIC protocol. In Section 3, we describe our ID-CDIC protocol and demonstrate its efficiency by implementing it. In Section 4, we provide security proofs of the proposed ID-CDIC protocol. We conclude the paper in Section 5.

2. Preliminaries

In this section, we review some basic knowledge of ID-CDIC protocols, including its system model, components and security model [24].

2.1. System model

Fig. 1 depicts the system model of the cloud data integrity checking with identity-based auditing mechanism. Four entities are involved in the system. (1) A user who possesses a great deal of data needed to be stored on cloud, can be an enterprise or an individual. (2) A cloud server who provides significant storage space to store cloud user's data. (3) A third party auditor (TPA) who has expertise and capacities that the client does not have, and the TPA can check the correctness of the data stored on cloud servers upon request. (4) A private key generator (PKG) who generates keys for users by using the user's identity information. Once the user stores data on the cloud, the user loses the direct control over their data. Besides, the cloud server is not fully-trusted. Therefore, it is necessary for the user to efficiently verify whether the outsourced data are intact or not.

The identity-based integrity checking mechanism consists of the following algorithms.

- Setup (1^k) is a probabilistic algorithm run by the PKG. It takes a security parameter k as input and outputs the system parameters *param*, the master secret key *msk* and the master public key *mpk*.

[1] http://www.businessinsider.com/amazon-lost-data-2011-4.