Shuang Tan, Yan Jia, 2014. NaEPASC: a novel and efficient public auditing scheme for cloud data. *Journal of Zhejiang University-SCIENCE C (Computers & Electronics)*, **15**(9):794-804. [doi:10.1631/jzus.C1400045]

NaEPASC: a novel and efficient public auditing scheme for cloud data

JZUS C (Comput & Electron)

Key words: Cloud storage, Public verification, Identity-based aggregate signature

Corresponding author: Shuang Tan
E-mail: taylorshuang@163.com

Introduction

- Cloud users no longer possess their data in a local data storage infrastructure, which would result in auditing for the integrity of outsourced data being a challenging problem, especially for users with constrained computing resources. Therefore, how to help users complete the verification of the integrity of the outsourced data has become a key issue.

- The existing mechanism can be used merely in such a situation: one key and one file.

- In this paper, an identity-based public verification protocol (NaEPASC) has been proposed to verify the users' outsourced data in the cloud without retrieving the original data file. The proposed scheme not only eliminates the burden on cloud users of tedious and possibly expensive auditing tasks, but also alleviates the users' fear of losing their keys.

IJCSC (Comput & Electron)

System model

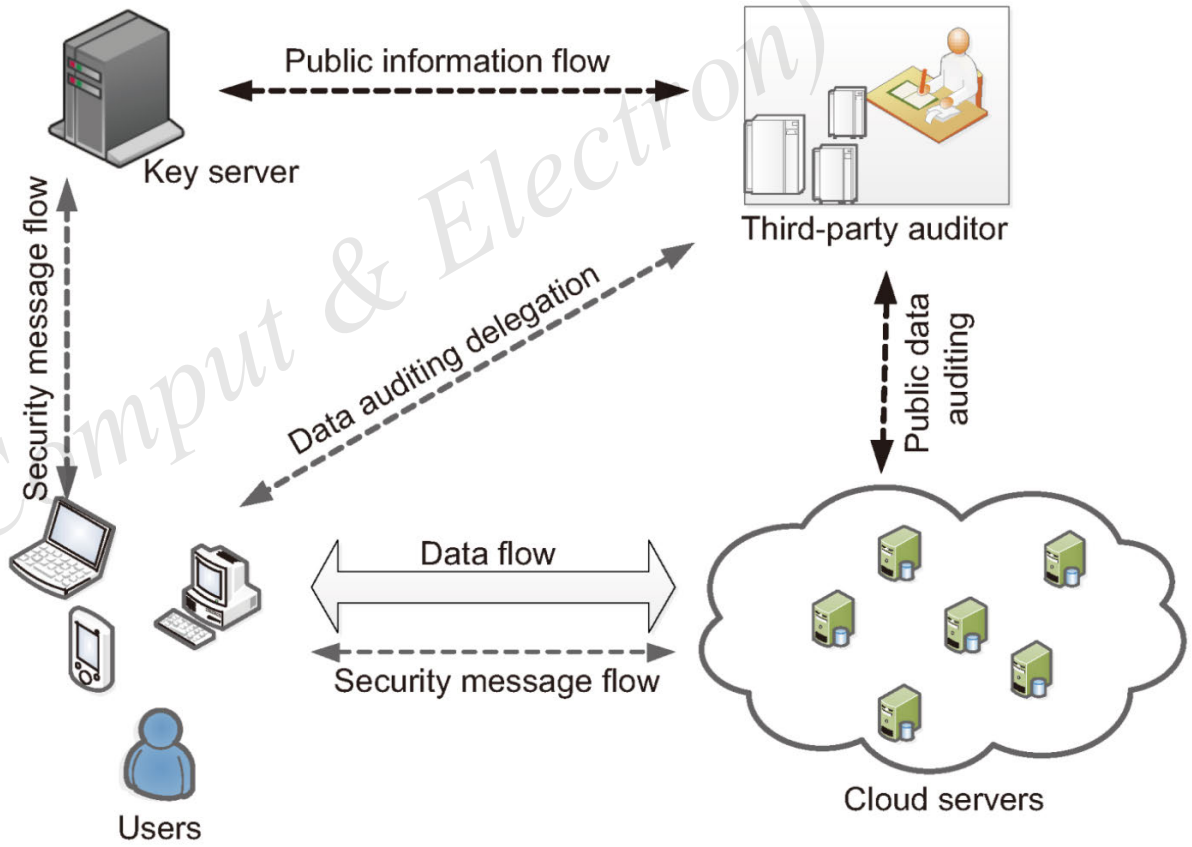


Fig. 1 The architecture of our cloud data storage model

JZUS C (Comput & Electron)

Design method

NaEPASC is usually executed in two phases: Setup and Challenge

Setup phase:

- The PKG initializes the public and secret parameters of the system by executing ***KeyGen***
- The cloud user with his/her public identity receives the secret key from the PKG, and pre-processes data file *F* using Sign to generate the verification metadata
- The cloud user stores data file *F* and the verification metadata at the cloud server, and deletes its local copies

IZUS C Comput & Electron

Challenge phase:

- The TPA first sends an auditing challenge request to the cloud server to make sure that the cloud server has truly stored their data files *F* at the time of the Challenge
- As the cloud server receives the request, it aggregates the proof of the stored data files *F* by executing ***GenProof***, and returns this proof to the TPA
- The TPA verifies this proof via running ***VerifyProof***

JZUS C (Comput & Electron)

Performance

Table 3 Performances under different numbers of sampled blocks c for high assurance checking schemes

Method	TPA computation time (ms)		Server computation time (ms)		Communication overhead (KB)	
	$c = 300$	$c = 460$	$c = 300$	$c = 460$	$c = 300$	$c = 460$
Wang *et al.* (2013)	636.0	963.9	619.4	947.5	4.24	6.43
Shacham and Waters (2008; 2013)	627.9	955.7	615.3	943.4	4.14	6.33
NaEPASC	**14.3**	**14.6**	1230.7	1886.1	4.16	6.34

JZUS C (Comput & Electron)

Conclusions

- In this paper, the first identity-based auditing scheme (NaEPASC) is proposed to check the integrity of the data stored in the cloud.
- NaEPASC adopts an identity-based aggregate signature to construct real-time and homomorphic verifiable tags, and the TPA is able to audit the correctness on behalf of the users.
- The proposed scheme not only eliminates the burden on cloud users of tedious and possibly expensive auditing tasks, but also alleviates the users' fear of losing their keys.
- Experiment results show that the proposed scheme is efficient and secure in checking the integrity of the data stored in the cloud.

JZUSCE (Comput. & Electron.)