Faculty of Engineering and Information Sciences - Papers: Part A

Faculty of Engineering and Information Sciences

1-1-2016

Comments on "public integrity auditing for dynamic data sharing with multi-user modification"

Yong Yu
University of Electronic Science and Technology of China, yyong@uow.edu.au

Yannan Li
University of Electronic Science and Technology of China

Jianbing Ni
University of Electronic Science and Technology of China

Guomin Yang
University of Wollongong, gyang@uow.edu.au

Yi Mu
University of Wollongong, ymu@uow.edu.au

See next page for additional authors

Recommended Citation

Yu, Yong; Li, Yannan; Ni, Jianbing; Yang, Guomin; Mu, Yi; and Susilo, Willy, "Comments on "public integrity auditing for dynamic data sharing with multi-user modification"" (2016). *Faculty of Engineering and Information Sciences - Papers: Part A*. 5062.
https://ro.uow.edu.au/eispapers/5062

Comments on "public integrity auditing for dynamic data sharing with multi-user modification"

Abstract

Recently, a practical public integrity auditing scheme supporting multiuser data modification (IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, DOI 10.1109/TIFS.2015.2423264) was proposed. Although the protocol was claimed secure, in this paper, we show that the proposal fails to achieve soundness, the most essential property that an auditing scheme should provide. Specifically, we show that a cloud server can collude with a revoked user to deceive a third-party auditor (TPA) that a stored file keeps virgin even when the entire file has been deleted.

Disciplines

Engineering | Science and Technology Studies

Publication Details

Yu, Y., Li, Y., Ni, J., Yang, G., Mu, Y. & Susilo, W. (2016). Comments on "public integrity auditing for dynamic data sharing with multiuser modification". IEEE Transactions on Information Forensics and Security, 11 (3), 658-659.

Authors

Yong Yu, Yannan Li, Jianbing Ni, Guomin Yang, Yi Mu, and Willy Susilo

Comments on Public Integrity Auditing for Dynamic Data Sharing with Multi-user Modification

Yong Yu, Yannan Li, Jianbing Ni, Guomin Yang, Yi Mu and Willy Susilo

***Abstract*— Recently, a practical public integrity auditing scheme supporting multi-user data modification (IEEE Trans. on Information Forensics and Security, DOI 10.1109/TIFS.2015.2423264) was proposed. Although the protocol was claimed secure, in this paper, we show that the proposal fails to achieve *soundness*, the most essential property that an auditing scheme should provide. Specifically, we show that a cloud server can collude with a revoked user to deceive a third-party auditor (TPA) that a stored file keeps virgin even when the entire file has been deleted.**

Keywords: Cloud storage, data integrity, soundness

I. Introduction

Cloud storage enables cloud users to focus more on their core competencies by alleviating data owners' burden of local data storage and maintenance. However, data integrity becomes the biggest concern of cloud users because they lose physical control over their outsourced files. Ateniese et al. proposed the notion of provable data possession [1], or data auditing, to address this challenging problem. Considering many practical scenarios where all users sharing cloud data need to read and modify the data, very recently, Yuan and Yu [2] proposed a novel and efficient integrity auditing scheme supporting multi-user modification, public auditing, high error detection probability and efficient user revocation.

Contributions. We show the schemes in [2] fail to achieve the basic property of a secure auditing scheme – *soundness*. We demonstrate that in the scheme with basic user revocation, if a malicious cloud server colludes with a revoked user, the server is able to generate a valid proof of a challenge even if the entire file has been deleted. The same attack can also be applied to the multi-file auditing protocol and the protocol with advanced user revocation in [2].

II. Review

Some notations of the system are as follows. $H()$ denotes a one-way hash function and λ represents a security parameter. G, G_1 denote two multiplicative cyclic groups and $e : G \times G \rightarrow G_1$ is a bilinear map. q is the order of G and g, u are random generators of G. $f_{\vec{\alpha}}(x)$ denotes a polynomial with coefficients $\vec{\alpha} = \{\alpha_0, \cdots, \alpha_{s-1}\}$, where $\alpha_j \in Z_q^*$.

Suppose there are K users $\{u_k\}_{0 \leq k \leq K-1}$ in a group. The data owner, u_0, is the master user who can revoke any other group users when necessary. We briefly review the following algorithms in [2] to support our analysis.

Key Generation. u_0 chooses K random numbers $\{\epsilon_k \in Z_q{}^*\}_{0 \leq k \leq K-1}$ and a random $\alpha \in Z_q{}^*$, computes $\nu = g^{\alpha\epsilon_0}$, $\kappa_0 = g^{\epsilon_0}$, $\{\kappa_k = g^{\epsilon_k}, g^{\frac{\epsilon_0}{\epsilon_k}}\}_{1 \leq k \leq K-1}$, $\{g^{\alpha^j}\}_{0 \leq j \leq s+1}$. The public key of u_0 is $PK = (g, u, q, \nu, \{g^{\alpha^j}\}_{0 \leq j \leq s+1}, \kappa_0, \{\kappa_k, g^{\frac{\epsilon_0}{\epsilon_k}}\}_{1 \leq k \leq K-1})$ and the master key of u_0 is $MK = (\epsilon_0, \alpha)$. The secret key of user u_k is ϵ_k, where $1 \leq k \leq K-1$.

File Processing. u_0 first splits the file F into n blocks, and each block into s sectors: $\{m_{ij}\}_{1 \leq i \leq n, 0 \leq j \leq s-1}$, and computes authentication tag $\{\sigma_i\}_{1 \leq i \leq n}$ for each block as:

$$\sigma_i = (u^{B_i} \cdot \textstyle\prod_{j=0}^{s-1} g^{m_{ij}\alpha^{j+2}})^{\epsilon_0} = (u^{B_i} \cdot g^{f_{\vec{\beta_i}}(\alpha)})^{\epsilon_0}.$$

where $\overrightarrow{\beta_i} = \{0, 0, \beta_{i,0}, \beta_{i,1}, \cdots, \beta_{i,s-1}\}$ and $\beta_{ij} = m_{ij}$, $B_i = H(fname||i||t_i||k)$, in which $fname$ is the filename, i is the index of data block m_i, t_i is the time stamp and k is the index of the user in the group. Finally, u_0 uploads data blocks and tags to the cloud.

Challenge. The TPA randomly chooses d data blocks as a set D. Suppose the chosen d blocks are modified by a set of users, denoted by C. The TPA generates two random numbers R and μ, and produces a set $X = \{(g^{\frac{\epsilon_0}{\epsilon_k}})^R\}_{k \in C}$. If D contains blocks lastly modified by any revoked user, the TPA adds $(g^{\frac{\epsilon_0}{\epsilon_0+\rho}})^R$ to the set X. Then, the TPA sends the challenge $CM = (D, X, g^R, \mu)$ to the cloud.

Prove. Upon receiving the challenge $CM = (D, X, g^R, \mu)$, the cloud server generates a proof $Prf = (\pi, \psi, y)$ and forwards it to the TPA for verification. We omit the details of generating a proof due to space limit.

Verify. Upon receiving the proof, the TPA verifies the integrity of F by first computing $\eta = u^{\omega}$, where $w = \sum_{i \in D} B_i p_i$. Then it checks if the following equation holds:

$$e(\eta, \kappa_0{}^R) \cdot e(\psi^R, \nu \cdot \kappa_0{}^{-\mu}) \stackrel{?}{=} \pi \cdot e(\kappa_0{}^{-y}, g^R).$$

If it holds, it outputs *Accept*; otherwise outputs *Reject*.

User Revocation. The following steps are executed to revoke a user, say u_k $(k \neq 0)$:

1) u_0 randomly chooses $\rho \in Z_q{}^*$, computes $\chi = \frac{\epsilon_0+\rho}{\epsilon_k} \mod q$ and sends it to the cloud. u_0 also computes $g^{\frac{\epsilon_0}{\epsilon_0+\rho}}$ and forwards it to the valid group users and the TPA as a part of the PK.
2) Upon receiving χ, the cloud updates the authentication tags of blocks that are lastly modified by u_k as: $\sigma'_i = \sigma_i{}^{\chi} = (u^{B_i} \cdot g^{f_{\vec{\beta_i}}(\alpha)})^{\epsilon_0+\rho}$.
3) The TPA and the valid group users discard the public information $g^{\frac{\epsilon_0}{\epsilon_k}}$.

III. Remarks on the protocol

Firstly, we revisit the system model of public integrity auditing for dynamic data sharing with multi-user modification [2]. As illustrated in Fig. 1, the system is composed of three major entities, namely the cloud server, the TPA and group users. Some users are valid users who can access and modify

Yong Yu, Yannan Li and Jianbing Ni are with Big Data Research Center, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China. Email: yuyong@uestc.edu.cn

Yong Yu is with State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100093, China.

Yong Yu, Guomin Yang, Yi Mu and Willy Susilo are with the Center for Computer and Information Security Research, School of Computing and Information Technology, University of Wollongong, Wollongong, NSW 2522, Australia. Email: {gyang, ymu, wsusilo}@uow.edu.au.

Corresponding Author: Willy Susilo

the cloud data while some other users have been revoked due to a variety of reasons. For example, the secret key of the user has been exposed or the user becomes dishonest. Consequently, it is fair to assume that valid users in the group are honest but the revoked users have incentive to become an adversary of the system, i.e., they might collude with a malicious cloud server to deceive a verifier.

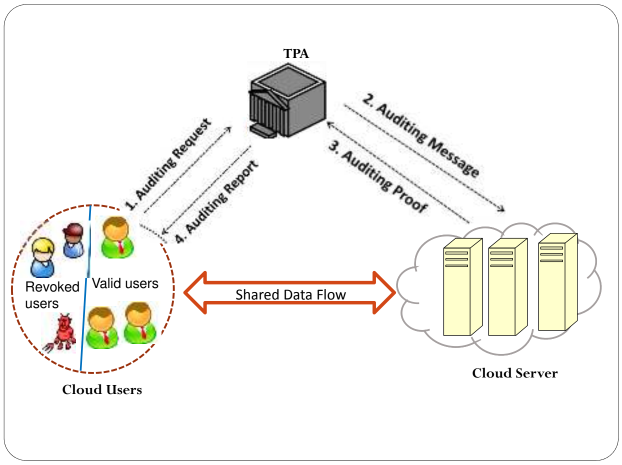

Fig. 1. System model of public auditing with multi-user modification

A. Soundness of the protocol with basic user revocation

In the following, we describe a feasable attack on the protocols in [2] where a malicious cloud server can collude with a revoked user to generate a valid response even if the entire file stored on cloud has been deleted.

During the execution of the protocol, the **Key Generation**, **File Processing** and **Update algorithm** are the same as those in [2]. Given a challenge $CM = \{D, X, E = g^R, \mu\}$, assume the set D contains blocks lastly modified by a revoked user, say $u_k(1 \le k \le K-1)$ with secret key ϵ_k. According to the protocol, $(g^{\frac{\epsilon_0}{\epsilon_0+\rho}})^R$ is added to set X and $\chi = \frac{\epsilon_0+\rho}{\epsilon_k} \pmod q$ is sent to the cloud server. With such information and the system public parameters, the cloud server and the revoked user can collude to generate a response for challenge CM without using the challenged data file as follows.

- Generate $\{p_i = \mu^i \pmod q)\}$, $i \in D$ and $B_i = H(\{fname \parallel i \parallel t_i \parallel k\})$.
- Compute $w = \sum_{i \in D} B_i p_i$ and $\eta = u^\omega$.
- Retrieving $(g^{\frac{\epsilon_0}{\epsilon_0+\rho}})^R$ from X, with the value ϵ_k from the revoked user and χ from the cloud server, they compute
$$A = ((g^{\frac{\epsilon_0}{\epsilon_0+\rho}})^R)^{\chi\epsilon_k} = g^{\epsilon_0 R} = \kappa_0^R.$$
- Pick a random $t^* \in Z_q$, compute $\psi^* = g^{t^*}$ and $B = E^{t^*}$.
- Pick a random $y^* \in Z_q$.
- Compute
$$\pi^* = \frac{e(\eta, A) \cdot e(B, \nu \cdot \kappa_0^{-\mu})}{e(\kappa_0^{-y^*}, E)}.$$

The forgery $Prf^* = \{\pi^*, \psi^*, y^*\}$ is a valid proof for the challenge CM because it can pass the **Verify** algorithm.

$$\begin{aligned} e(\eta, \kappa_0^R) \cdot e((\psi^*)^R, \nu \cdot \kappa_0^{-\mu}) &= e(\eta, A) \cdot e((g^{t^*})^R, \nu \cdot \kappa_0^{-\mu}) \\ &= e(\eta, A) \cdot e(E^{t^*}, \nu \cdot \kappa_0^{-\mu}) \\ &= e(\eta, A) \cdot e(B, \nu \cdot \kappa_0^{-\mu}) \\ &= \pi^* \cdot e(\kappa_0^{-y^*}, E) \\ &= \pi^* \cdot e(\kappa_0^{-y^*}, g^R) \end{aligned}$$

Therefore, the forged proof Prf^* is valid. The cloud server can deceive TPA that the file keeps intact even the entire file has been polluted or deleted. Note that the same attack can be applied to the protocol for efficient multi-file auditing, where the TPA can handle integrity auditing of multiple files simultaneously at the cost comparable to the single file scenario, and we don't repeat the attack here.

B. Soundness of the protocol with advanced user revocation

An advanced user revocation protocol was also proposed in [2] to prevent the revoked users from generating valid authentication tags by compromising a single cloud server node. In this enhanced scheme, the master user u_0 runs a (U, N)-Shamir secret sharing protocol to generate N points $(i, f(i))$ of a $U-1$ degree polynomial $f(x) = \chi + a_1 x + a_2 x^2 + \cdots + a_{U-1} x^{U-1}$. Those N points will be sent to N nodes of a cloud server. Unfortunately, this technique cannot make the protocol sound because it is the cloud server who acts as the adversary when considering soundness and any U out of N cloud nodes are able to recover χ using Lagrange interpolation over a finite field. Specifically, given any U pairs $(i, f(i))$, χ can be recovered as

$$\chi = \sum_{1 \le i \le U} f(i) \prod_{1 \le i \le U, j \ne i} \frac{-j}{i-j}.$$

With the recovered χ, the malicious cloud server can collude with a revoked user to break the soundness of the improved protocol in [2].

Real-world dangers. If the protocols in [2] are adopted in reality, after corrupting a revoked user, the malicious cloud service providers can deceive the data owners by making use of the aforementioned attacks. Consequently, the cloud servers can charge the data owners for storage without hesitation but do not store their data. As a result, the date owners cannot retrieve the outsourced data when needed, which might incur a significant loss to data owners since the lost data might be important and unique.

ACKNOWLEDGEMENTS. This work is supported by the NSFC of China under Grant Number 61300213, 61272436.

REFERENCES

[1] G. Ateniese, R. C. Burns, R. Curtmola, J. Herring, L. Kissner, Z. N. J. Peterson, and D. X. Song, "Provable data possession at untrusted stores," in *Proc. of ACM Conference on Computer and Communications Security* 2007: 598-609, 2007.

[2] J. Yuan, and S. Yu, "Public integrity auditing for dynamic data sharing with multi-user modification," IEEE Trans. on Information Forensics and Security, DOI: 10.1109/TIFS.2015.2423264.

Yong Yu received his Ph.D. degree in cryptography from Xidian University in 2008. He is currently an associate professor of University of Electronic Science and Technology of China. His research focuses on cryptography and its applications, especially public encryption, digital signature and secure cloud storage. He has published more than 40 research papers in reputable conferences and journals.

Yannan Li received her bachelor degree from University of Electronic Science and Technology of China in 2014. She is currently a master student in School of Computer Science and Engineering at University of Electronic Science and Technology of China. Her research interest is secure cloud data storage.

Jianbing Ni received his master degree from University of Electronic Science and Technology of China in 2014. His research interests are applied cryptography and information security, with current focus on digital signature and secure data storage in cloud computing.

Guomin Yang received his Ph.D. degree from the Computer Science Department at City University of Hong Kong in 2009. He is currently a senior lecturer in University of Wollongong, Australia. His research interests are cryptography and network security.

Yi Mu received his Ph.D. from the Australian National University in 1994. He is currently a professor and the co-director of Centre for Computer and Information Security Research at the University of Wollongong, Australia. His current research interests include network security, computer security, and cryptography. He is the editor-in-chief of International Journal of Applied Cryptography and serves as associate editor for nine other international journals. He is a senior member of the IEEE and a member of the IACR.

Willy Susilo received the Ph.D. degree in computer science from the University of Wollongong, Australia. He is a Professor and the Head of School of Computing and Information Technology at the University of Wollongong in Australia. He is also the Director of Centre for Computer and Information Security Research, University of Wollongong. He has been awarded the prestigious ARC Future Fellow by the Australian Research Council. His main research interests include cloud security, cryptography and information security. He has served as a program committee member in major international conferences. He is a senior member of IEEE and a member of the IACR.