Wuhan University
Journal of Natural Sciences

2018, Vol.23 No.4, 362-368

Article ID 1007-1202(2018)04-0362-07
DOI https://doi.org/10.1007/s11859-018-1335-9

ID-Based Public Auditing Protocol for Cloud Storage Data Integrity Checking with Strengthened Authentication and Security

☐ JIANG Hong[1], XIE Mingming[2], KANG Baoyuan[2] , LI Chunqing[2], SI Lin[2]

1. School of Management, Tianjin Polytechnic University, Tianjin 300387, China;

2. School of Computer Science and Software, Tianjin Polytechnic University, Tianjin 300387, China

Abstract: Cloud storage service reduces the burden of data users by storing users' data files in the cloud. But, the files might be modified in the cloud. So, data users hope to check data files integrity periodically. In a public auditing protocol, there is a trusted auditor who has certain ability to help users to check the integrity of data files. With the advantages of no public key management and verification, researchers focus on public auditing protocol in ID-based cryptography recently. However, some existing protocols are vulnerable to forgery attack. In this paper, based on ID-based signature technology, by strengthening information authentication and the computing power of the auditor, we propose an ID-based public auditing protocol for cloud data integrity checking. We also prove that the proposed protocol is secure in the random oracle model under the assumption that the Diffie-Hellman problem is hard. Furthermore, we compare the proposed protocol with other two ID-based auditing protocols in security features, communication efficiency and computation cost. The comparisons show that the proposed protocol satisfies more security features with lower computation cost.

Key words: ID-based auditing; data integrity checking; digital signature; security; bilinear map

CLC number: TP 309

Received date: 2017-12-23

Foundation item: Supported by the Applied Basic and Advanced Technology Research Programs of Tianjin (15JCYBJC15900) and the National Natural Science Foundation of China (51378350)

Biography: JIANG Hong, female, Master, research direction: electronic commerce protocol. E-mail: jianghong.comcn@aliyun.com

0 Introduction

With the development of computer and network technology, cloud computing emerges as a service on-demand via the Internet. Presently, many cloud services are available based on data and applications outsourcing. More and more people like to store their data in the cloud to reduce their burden and costs.

Although cloud storage service reduces the burden of data users, cloud storage service may cause new security issues [1]. Once data users transfer their data files to the cloud, they no longer possess their data files locally. But, the users might worry about the correct and safe storage of their data and expect to check their data integrity periodically. So, it is becoming the primary issue that periodical data integrity check is needed in the cloud storage service. But, without the local copy of data file, data users cannot check the data integrity in conventional method. This case breeds auditing agency service in which there is an auditor who has capabilities to help data users check data integrity. Based on the third-party auditor, there are many auditing protocols [2-15]. But these protocols were constructed on public key cryptographic system, thus more storage space and higher computation cost are needed in these protocols.

ID-based cryptography [16] provides entities with public and private key pairs without the need for certificates and CA deployment. Each entity uses one of its identifiers as its public key and its private key is generated by the public key generator using its public key. To reduce the computation and management cost, a few

ID-based public auditing protocols have been proposed recently[17-20], based on ID-based cryptography. Yu *et al*[19] proposed an ID-based auditing mechanism from RSA for cloud data integrity checking. Wei *et al*[20] proposed an ID-based protocol for security storage and computation in cloud computing. Wang *et al* proposed an ID-based data integrity auditing protocol [17]. But Zhang *et al* [18] pointed that Wang *et al*'s protocol was not ID-based auditing since data tag generation algorithm was not ID-based signature. Zhang *et al* also proposed an efficient ID-based public auditing protocol for cloud data integrity checking [18] and claimed that the proposed protocol was provably secure in the random oracle model. However, in Ref.[21], He *et al* proposed two attacks to show that Zhang *et al*'s protocol was not secure against the malicious cloud server. In fact, in Ref.[18], Zhang *et al* only proved that the file block tag could not be forged, which could not ensure that the proof information cannot be forged. In Refs. [22-30], the public auditing protocols for shared data and privacy-preserving were proposed.

In an auditing protocol the auditor should not only have computation ability and integrity checking expertise, but also have certain storage space compared with the data user. But in many existing protocols, the auditor only produces challenge information and does a few calculations almost without fundamental relation with the proof information from the cloud server. Also in existing protocols except for Ref. [20], the cloud server does not return signature authentication information to the data user when he receives the user's data. This is irrational for security. Based on these views, we propose in this paper an ID-based public auditing protocol for data integrity checking with strengthened information authentication and security. In the new protocol, the auditor will compute some substantial parameters used in verification phase. So, in verification phase, part parameters come from the cloud server, while other parameters needed come from the auditor. This solution might prevent forgery from the cloud server, so as to enhance the security of the protocol.

Our contributions are three-folds:

1) Based on ID-based signatures, we propose a new ID-based public auditing protocol for data integrity checking. The new protocol is proved secure in the random oracle model under the assumption that the Diffie-Hellman problem is hard.

2) In the verification phase of the new protocol, since some data relevant to file blocks come from auditor's computation, it is more difficult for the malicious cloud server to forge data integrity proving information.

3) In the new protocol, the cloud server returns the acceptation information when he receives valid date from the data user. After the verification of the file tag signature generated by data user and the verification of the file acceptation information from the cloud server, the auditor returns an acceptation auditing agency information to the data user. These tactics make the protocol more rigorous and practical.

The rest of the paper is organized as follows. In Section 1, we propose the system and security model. In Section 2, we review bilinear pairing and computational Diffie-Hellman problem relevant to the security of proposed protocol. An ID-based public auditing protocol is proposed in Section 3. In Section 4, we provide security proofs of the proposed protocol. In Section 5, we compare the proposed protocol with other two protocols in security, communication efficiency and computation cost. Conclusion is given in Section 6.

1 System and Security Model

As Ref. [24], there are a data user (DU), a cloud server (CS), a third-party auditor (AU) and a private key generator (PKG) in an ID-based public auditing protocol. The data user DU uploads his data file to the cloud server CS for storage. Then DU entrusts the trusted-third party AU who has expertise and computation capabilities to check the data file integrity periodically. PKG is a trusted authority. DU is honest, and AU is honest but-curious. DU and AU perform all the procedure steps honestly. But CS is a semi-trusted party. He might change or delete the data user's file for his benefit [24].

Here we mainly consider the forge attack launched by CS: CS tries to forge the proof information for passing data file integrity checking. Based on Refs. [18, 19], the security game consists of the following phases.

Setup: The challenger who acts as a user generates system parameters, system public key, and forwards the system parameters and system public key to the adversary.

Queries: The adversary might make lots of following queries.

1) Hash queries: When the adversary queries the hash values, the challenger must answer him.

2) Key extraction queries: When the adversary queries the private key of someone, the challenger must run key extraction algorithm, and forward it to the adversary.

Output: Finally, the adversary outputs a valid proof information on chosen data file blocks. But, the adver-

sary never makes key extraction queries on the user.

2 Preliminary

2.1 The Bilinear Pairing

Let G_1 be a cyclic additive group generated by p, whose order is a prime q, and G_2 be a cyclic multiplicative group of the same order. Let $e: G_1 \times G_1 \to G_2$ be a pairing map which satisfies the following conditions [24]:

1) Bilinearity: for any $P,Q,R \in G_1$, then

$$e(P+Q,R)=e(P,R)e(Q,R)$$

and

$$e(P,Q+R)=e(P,Q)e(P,R)$$

In particular, for any $a,b \in Z_q$, $e(aP,bP)=e(P,abP)=e(abP,P)=e(P,P)^{ab}$.

2) Non-degeneracy: There exists $P,Q \in G_1$, such that $e(P,Q) \neq 1$.

3) Computability: There is an efficient algorithm to compute $e(P,Q)$ for all $P,Q \in G_1$.

The typical way of obtaining such pairings is by deriving them from the Weil-pairing or the Tate-pairing on an elliptic curve over a finite field.

2.2 Computational Diffie-Hellman (CDH) Problem

A challenger chooses an additive cyclic group G with order q, P is a generator of G. Given (aP, bP) for unknown $a,b \in Z_q^*$, the adversary must compute abP.

3 The Proposed Protocol

3.1 Basic Idea and Protocol Outline

Although many auditing protocols are proposed, even with extended function [13, 14, 20], in our view, many existing protocols are thoughtless in the security of the whole system, and the auditor's capabilities are not fully utilized. For the security of the whole system and the practical application, firstly, all communication between any two entities should be encrypted. Secondly, once the cloud server accepts the data file from the data user, the cloud server should return acceptation information to the data user. Later, when data user asks the auditor to check the date file integrity for him, the auditor not only need verify the file tag generated by the data user, but also need verify the acceptation information from the cloud server. In proof information verification phase, the auditor should provide certain crucial parameters relevant to the file blocks to decrease the probability of the cloud server's cheat.

Based on above idea, we propose a new ID-based auditing protocol. The new protocol consists of six algorithms: setup, key extraction, tag generation, challenge phase, proof phase, and verify phase. In setup phase, PKG generates system parameters including system master key and system public key. In key extraction phase, using entities identities information, PKG generates private key for each entity. In tag generation phase, the data user generates signatures for data file's name and each file block, the cloud server accepts the storage service request from the data user. In challenge phase, the auditor produces the challenge information for data file integrity checking. In proof phase, using file blocks information and signatures from the data user, the cloud server generates proof information for file integrity checking. In verify phase, the auditor verifies whether the proof information is valid or not. Table 1 lists all notations that will be used in the proposed protocol.

Table 1 Notations in the proposed protocol

Notation	Description
PKG	Private key generator that is responsible for generating system parameters including system master key and system public key
DU	The data user
SC	The cloud server
AU	The auditor
G_1	An additive cyclic group
G_2	A multiplicative cyclic group
P	A generator of G_1
e	A bilinear map of $G_1 \times G_1 \to G_2$
s	System master key
P_{pub}	System public key
H	Hash function of $\{0,1\}^* \to G_1$
h	Hash function of $\{0,1\}^* \to Z_q$
M	A data file
m_i	The i-th block of the file M
N	The total number of the blocks of the file M
τ	The file tag
S_{DU}	The private key of the data user (DU)
S_{CS}	The private key of the cloud server (SC)
S_{AU}	The private key of the auditor (AU)

Note that for security all communication between any two entities should be encrypted. For entities ID_1 and ID_2, they can encrypt their communication using their shared key $e(S_{\mathrm{ID}_1}, Q_{\mathrm{ID}_2}) = e(S_{\mathrm{ID}_2}, Q_{\mathrm{ID}_1})$ (see key extraction algorithm). But, we omit encryption in construction description for brevity.

3.2 Construction Description

Setup: Given a security parameter $k \in Z$, the algorithm run by the PKG works as follows:

1) Run the parameter generator on input *k* to generate a prime *q*, an additive cyclic group G_1 and a multiplicative cyclic group G_2 of the same order *q*, a generator *P* of G_1 and a bilinear map $e: G_1 \times G_1 \to G_2$.

2) Pick a random $s \in Z_q^*$ as master key of PKG and set system public key $P_{\mathrm{pub}} = s \cdot P$.

3) Choose two cryptographic hash functions

$$H: \{0,1\}^* \to G_1, h: \{0,1\}^* \to Z_q$$

The system parameters are $\langle q, G_1, G_2, e, P, P_{\mathrm{pub}}, H, h \rangle$.

Key Extraction: When any one of the data user (DU), the cloud server (SC) and the auditor (AU) wants to register his identity ID to PKG, the algorithm run by the PKG works as follows:

1) Compute $Q_{\mathrm{ID}} = H(\mathrm{ID}) \in G_1$.

2) Set the private key $S_{\mathrm{ID}} = s \cdot Q_{\mathrm{ID}}$, where *s* is the master key of PKG.

By the two steps, the data user (DU), the cloud server (SC) and the auditor (AU) obtain their private key S_{DU}, S_{CS} and S_{AU}, respectively.

Tag Generation: For a data file $M = m_1 \| \cdots \| m_n$, the data user DU selects a random file name , and labels $\tau = \mathrm{name} \| n$ as the file tag, then

1) Choose $r_\tau \in_R Z_q^*$ and compute $R_\tau = r_\tau \cdot P_{\mathrm{pub}}$.

2) Compute $\sigma_\tau = (h(\tau \| R_\tau \| ID_{\mathrm{DU}}) + r_\tau) S_{\mathrm{DU}}$.

3) For each file block m_i, choose $r_i \in_R Z_q^*$ and compute $R_i = r_i \cdot P$,

$\sigma_{m_i} = h(m_i \| R_i \| \mathrm{ID}_{\mathrm{DU}}) S_{\mathrm{DU}} + (m_i + r_i) P_{\mathrm{pub}}$.

Let $\varphi = ((\sigma_{m_1}, R_1), \cdots, (\sigma_{m_n}, R_n))$.

4) DU sends $(\mathrm{ID}_{\mathrm{DU}}, M, (\tau, \sigma_\tau, R_\tau), \varphi)$ to CS.

5) CS checks the following equations

$$e(\sigma_\tau, P) = e(h(\tau \| R_\tau \| \mathrm{ID}_{\mathrm{DU}}) P_{\mathrm{pub}} + R_\tau, Q_{\mathrm{DU}})$$

$e(\sigma_{m_i}, P) = e(h(m_i \| R_i \| \mathrm{ID}_{\mathrm{DU}}) Q_{\mathrm{DU}} + R_i + m_i P, P_{\mathrm{pub}})$, $1 \leqslant i \leqslant n$.

6) If above two equations hold, CS chooses $r_v \in_R Z_q^*$ and computes $R_v = r_v \cdot P_{\mathrm{pub}}$,

$$V = (h(\tau \| R_v \| \mathrm{ID}_{\mathrm{CS}}) + r_v) S_{\mathrm{CS}}$$

and sends (V, R_v) to DU, then stores $(\mathrm{ID}_{\mathrm{DU}}, M, (\tau, \sigma_\tau, R_\tau), \varphi)$.

7) DU checks equation

$$e(V, P) = e(h(\tau \| R_v \| \mathrm{ID}_{\mathrm{CS}}) P_{\mathrm{pub}} + R_v, Q_{\mathrm{CS}})$$

If it holds, DU sends

$$\mathrm{RE}_{\mathrm{DU}} = (\mathrm{ID}_{\mathrm{DU}}, \mathrm{ID}_{\mathrm{CS}}, \tau, (\sigma_\tau, R_\tau), (V, R_v), \omega = ((h_1, R_1), \cdots, (h_n, R_n)))$$

to AU. Here $h_i = h(m_i \| R_i \| \mathrm{ID}_{\mathrm{DU}})$, *i*=1, ⋯, *n*.

8) AU checks the following equations

$$e(\sigma_\tau, P) = e(h(\tau \| R_\tau \| \mathrm{ID}_{\mathrm{DU}}) P_{\mathrm{pub}} + R_\tau, Q_{\mathrm{DU}})$$

$$e(V, P) = e(h(\tau \| R_v \| \mathrm{ID}_{\mathrm{CS}}) P_{\mathrm{pub}} + R_v, Q_{\mathrm{CS}})$$

If the two equations hold, AU sends a response to DU for expressing his accept of the auditing agency, and stores $\mathrm{RE}_{\mathrm{DU}}$.

9) When DU receives the response from AU, DU deletes the file *M*.

Challenge phase: To check the integrity of the outsourced data file *M*, AU randomly chooses a set $I \subseteq [1, n]$ and a number $a \in Z_q$ to generate the challenging information

$$\mathrm{Chall} = [\mathrm{ID}_{\mathrm{DU}}, \tau, a, I]$$

and sends it to CS.

Prove phase: Upon receiving $\mathrm{Chall}=[\mathrm{ID}_{\mathrm{DU}}, \tau, a, I]$, CS finds $(\mathrm{ID}_{\mathrm{DU}}, M, (\tau, \sigma_\tau, R_\tau), \varphi)$ and produces set $\omega = \{(i, b_i)\}, i \in I$. Here, $b_i = a^i \bmod q$.

Then using $M = m_1 \| \cdots \| m_n$ and φ, CS computes

$$\sigma = \sum_{i \in I} b_i \sigma_{m_i}, \quad \mu = \sum_{i \in I} b_i m_i$$

and sends (σ, μ) to AU.

Verify phase: Upon receiving the proof information (σ, μ), based on stored information $\mathrm{RE}_{\mathrm{DU}}$, AU computes $h = \sum_{i \in I} b_i h_i$, $R = \sum_{i \in I} b_i R_i$.

Then AU checks the equation

$$e(\sigma, P) = e(h Q_{\mathrm{DU}} + R + \mu P, P_{\mathrm{pub}})$$

If the equation holds, AU accepts the proof.

4 Security

In this section, we discuss the security of the proposed protocol .

4.1 Correctness

Firstly, we show that the signature σ_τ on the file tag can be verified by equation

$$e(\sigma_\tau, P) = e(h(\tau \| R_\tau \| \mathrm{ID}_{\mathrm{DU}})P_{\mathrm{pub}} + R_\tau, Q_{\mathrm{CU}})$$

In fact,

$$\begin{aligned} e(\sigma_\tau, P) &= e((h(\tau \| R_\tau \| \mathrm{ID}_{\mathrm{DU}}) + r_\tau)S_{\mathrm{DU}}, P) \\ &= e((h(\tau \| R_\tau \| \mathrm{ID}_{\mathrm{DU}}) + r_\tau)P_{\mathrm{pub}}, Q_{\mathrm{DU}}) \\ &= e(h(\tau \| R_\tau \| \mathrm{ID}_{\mathrm{DU}})P_{\mathrm{pub}} + R_\tau, Q_{\mathrm{DU}}) \end{aligned}$$

Secondly, the signature σ_{m_i} on the file block m_i can be verified by equation

$$e(\sigma_{m_i}, P) = e(h(m_i \| R_i \| \mathrm{ID}_{\mathrm{DU}})Q_{\mathrm{DU}} + R_i + m_i P, P_{\mathrm{pub}})$$

In fact,

$$\begin{aligned} e(\sigma_{m_i}, P) &= e(h(m_i \| R_i \| \mathrm{ID}_{\mathrm{DC}})S_{\mathrm{DC}} + (m_i + r_i)P_{\mathrm{pub}}, P) \\ &= e(h(m_i \| R_i \| \mathrm{ID}_{\mathrm{DC}})Q_{\mathrm{DC}} + (m_i + r_I)P, P_{\mathrm{pub}}) \\ &= e(h(m_i \| R_i \| \mathrm{ID}_{\mathrm{DC}})Q_{\mathrm{DC}} + R_i + m_i P, P_{\mathrm{pub}}) \end{aligned}$$

Thirdly, as σ_τ's verification, the signature V can be verified by equation

$$e(V, P) = e(h(\tau \| R_v \| \mathrm{ID}_{\mathrm{CS}})P_{\mathrm{pub}} + R_v, Q_{\mathrm{CS}})$$

Finally, the proof information (σ, μ) can be verified by equation

$$e(\sigma, P) = e(hQ_{\mathrm{DU}} + R + \mu P, P_{\mathrm{pub}})$$

In fact,

$$\begin{aligned} e(\sigma, P) &= e(\sum_{i\in I} b_i\sigma_{m_i}, P) = \prod_{i\in I} e(b_i\sigma_{m_i}, P) \\ &= \prod_{i\in I} e(b_i(h_i S_{\mathrm{DU}} + (m_i + r_i)P_{\mathrm{pub}}), P) \\ &= \prod_{i\in I} e(b_i h_i Q_{\mathrm{DU}} + b_i R_i + b_i m_i P, P_{\mathrm{pub}}) \\ &= e(\sum_{i\in I} b_i h_i Q_{\mathrm{DU}} + \sum_{i\in I} b_i R_i + \sum_{i\in I} b_i m_i P, P_{\mathrm{pub}}) \\ &= e(hQ_{\mathrm{DU}} + R + \mu P, P_{\mathrm{pub}}) \end{aligned}$$

4.2 Unforgeability

Theorem 1 If the CDH assumption is hard, then the proposed protocol is secure against proof information existential forgery attack.

Proof The thinking of proof is that if CS can forge valid proof information, the challenger will use the forged information to solve the CDH problem.

In the proof process, we look hash H and h as random oracles. Given CDH problem instance (aP, bP), the challenger sets system public key $P_{\mathrm{pub}} = aP$, and sets the target user DU's private as $t_i(bP)$, $t_i \in_R Z_q$.

When CS produces two valid forged proof information (σ_1^*, μ^*) and (σ_2^*, μ^*) for same challenge information (because the challenge information is the same, μ^* and R^* are the same, and the value of random oracles H and h will change in two forgeries), the following two equations hold:

$$\begin{aligned} e(\sigma_1^*, P) &= e(h_1^* Q_{\mathrm{CU}} + R^* + \mu^* P, P_{\mathrm{pub}}) \\ &= e(h_1^* t_1(bP) + R^* + \mu^* P, P_{\mathrm{pub}}) \\ e(\sigma_2^*, P) &= e(h_2^* Q_{\mathrm{CU}} + R^* + \mu^* P, P_{\mathrm{pub}}) \\ &= e(h_2^* t_2(bP) + R^* + \mu^* P, P_{\mathrm{pub}}) \end{aligned}$$

Then,

$$\sigma_2^* - \sigma_1^* = (h_2^* t_2 - h_1^* t_1)(abP)$$

$$abP = (h_2^* t_2 - h_1^* t_1)^{-1}(\sigma_2^* - \sigma_1^*)$$

4.3 Privacy-Preserving

Theorem 2 In the proposed protocol, AU cannot derive any information about DU's data file content during the whole auditing procedure.

Proof In the whole auditing procedure, AU can get information

$$\mathrm{RE}_{\mathrm{DU}} = (\mathrm{ID}_{\mathrm{DU}}, \mathrm{ID}_{\mathrm{CS}}, \tau, (\sigma_\tau, R_\tau), (V, R_v),\ \omega = ((h_1, R_1), \cdots, (h_n, R_n)))$$

$$\mathrm{Chall} = [\mathrm{ID}_{\mathrm{DU}}, \tau, a, I]$$

$$\sigma = \sum_{i\in I} b_i\sigma_{m_i},\ \ \mu = \sum_{i\in I} b_i m_i$$

However, $\tau, (\sigma_\tau, R_\tau), (V, R_v)$ and $\mathrm{Chall} = [\mathrm{ID}_{\mathrm{DU}}, \tau, a, I]$ is irrelevant to the file content.

Since equation $\mu = \sum_{i\in I} b_i m_i$ has $q^{|I|-1}$ solutions about the unknown blocks m_i, the probability for AU to successfully guess all m_i from this equation is $1/q^{|I|-1}$.

Finally, due to unknown σ_{m_i}, information σ and $\omega = ((h_1, R_1), \cdots, (h_n, R_n))$ are useless in inferring file blocks content.

5 Comparisons

In this section, we compare the proposed protocol with other two ID-based auditing protocols [17, 18] in security features, communication efficiency and computation cost. The features include file tag generation, file tag verification, block tag verification, and accepting auditing verification, unforgeability, and privacy-preserving. The proposed protocol satisfies all above features. But, except for unforgeability proof, Wang *et al*'s protocol[17] does not satisfy other features. Zhang *et al*'s protocol[18]

does not satisfy file tag verification, accepting auditing verification, and unforgeability. We show the comparison result of the features in Table 2.

In Table 3, the communication numbers of the proposed protocol and Refs. [17, 18] are shown. There are same communication numbers in key extraction phase, challenge phase and prove phase of each protocol. But in tag generation phase, the communication number in our protocol is obviously high. This is caused by the strengthened information authentication among the data user, the cloud server and the auditor.

In Table 4, we compare the main computation cost of the proposed protocol with Zhang *et al* [18]. Under the assumption that the challenging blocks number is $|I|$, the computation cost of Zhang *et al*'s protocol is $(6|I|+2)$H+$(10|I|+2)$S+$(3|I|+3)$B+$|I|$E. But, The computation cost of our protocol is $(2|I|+1)$H+$(7|I|+1)$S+$(2|I|+2)$B+$|I|$E. Obviously, our protocol has lower computation cost.

Table 2 Comparison of features

Protocol	F1	F2	F3	F4	F5	F6
Wang *et al* [17]	No	No	No	No	Yes	No
Zhang *et al* [18]	Yes	No	Yes	No	No	Yes
Ours	Yes	Yes	Yes	Yes	Yes	Yes

F1: File Tag Generation; F2: File Tag Verification; F3: Block Tag Verification; F4: Accepting Auditing Verification; F5: Unforgeability; F6: Privacy-Preserving

Table 3 Required communication number

Protocol	P1	P2	P3	P4
Wang *et al* [17]	1	1	1	1
Zhang *et al* [18]	1	1	1	1
Ours	1	4	1	1

P1: Key Extraction Phase; P2: Tag Generation Phase; P3: Challenge Phase; P4: Prove Phase

Table 4 Comparison of computation costs

Protocol	P1	P2	P3	P4
Zhang *et al* [18]	2H+2S	$4n$H+$6n$S+$3n$B	$\|I\|$H+$2\|I\|$S+$\|I\|$E	$\|I\|$H+$2\|I\|$S+3B
Ours	H+S	$2n$H+$5n$S+$2n$B	$\|I\|$S+$\|I\|$E	$\|I\|$S+2B

P1: Key Extraction Phase; P2: Block Tag Generation Phase; P3: Prove Phase; P4: Verify Phase
E: Exponential operation; S: Scalar multiplication; H: Hash computation; B: Bilinear pairings

6 Conclusion

In an auditing protocol the auditor should not only have computation ability and integrity checking expertise, but also have certain storage space. Also the cloud server should return signature authentication information to the data user when he receives the user's data. Based on these views, we propose a novel ID-based public auditing protocol for cloud storage data integrity checking, and prove that the proposed protocol is secure in the random oracle model under the assumption that the Diffie-Hellman problem is hard. Furthermore, we compare the proposed protocol with other two ID-based auditing protocols in security features, communication efficiency and computation cost. The proposed protocol is proved to be more secure and have lower computation cost. It must be recognized that the proposed protocol requires certain storage space. Compared with existing protocols, the proposed one can enhance the security without increasing too much burden of the auditor.

References

[1] Ateniese G, Burns R, Curtmola R, *et al*. Provable data possession at untrusted stores [C] // *Proceedings of the* 14*th ACM Conference on Computer and Communications Security* (*CCS*07). New York: ACM, 2007: 598-609.

[2] Ateniese G, Kamara S, Katz J. Proofs of storage from homomorphic identification protocols [C]// *Proceedings of the* 15*th International Conference on Theory and Application of Cryptology and Information Security*: *Advances in Cryptology*. Berlin, Heidelberg: Springer-Verlag, 2009: 319-333.

[3] Lu R, Lin X, Luan T, *et al*. Pseudonym changing at social spots: An effective strategy for location privacy in VANETs [J]. *IEEE Transaction on Vehicular Technology*, 2012, **61**(1) : 86-96.

[4] Kaaniche N, Boudguiga A, Laurent M. ID-based cryptography for secure cloud data storage [C]// *Proceedings of the IEEE Sixth International Conference on Cloud Computing*. Washington D C: IEEE Computer Society, 2013: 375-382.

[5] Wang Q, Wang C, Ren K, *et al*. Enabling public auditability and data dynamics for storage security in cloud computing [J]. *IEEE Transactions on Parallel and Distributed Systems*,

2011, **22**(5): 847-859.

[6] Wang C, Wang Q, Ren K, *et al*. Privacy-preserving public auditing for data storage security in cloud computing [C]// *Proceedings of the IEEE INFO-COM*. Washington D C: IEEE Computer Society, 2010: 525-533.

[7] Yuan J, Yu S. Public integrity auditing for dynamic data sharing with multiuser modification [J]. *IEEE Transactions on Information Forensics and Security*, 2015, **10**(8): 1717-1726.

[8] Zhang J, Zhao X. Privacy-preserving public auditing scheme for shared data with supporting multi-function [J]. *Journal of Communications*, 2015, **10**(7) : 535-542.

[9] Zeng K. Publicly verifiable remote data integrity [C]// *Proceedings of the* 10*th International Conference on Information and Communications Security*. New York: ACM, 2008: 419-434.

[10] Zhu Y, Hu H, Ahn G, *et al*. Cooperative provable data possession for integrity verification in multi-cloud storage [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2012, **23**(12): 2231-2244.

[11] Zhu Y, Wang H, Hu Z, *et al*. Dynamic audit services for integrity verification of outsourced storages in clouds [C]// *Proceedings of the ACM Symposium on Applied Computing*. New York: ACM, 2011: 1550-1557.

[12] Worku S, Xu C, Zhao J, *et al*. Secure and efficient privacy-preserving public auditing scheme [J]. *Computer and Electrical Engineering*, 2014, **40**(5): 1703-1713.

[13] Li Y, Yu Y, Yang B, *et al*. Privacy preserving cloud auditing with efficient key update [J]. *Future Generation Computer Systems,* 2018, **78**(2): 789-798.

[14] Xue L, Ni J, Li Y, *et al*. Provable data transfer from provable data possession and deletion in cloud storage [J]. *Computer Standard & Interfaces*, 2017, **54**(1): 46-54.

[15] Jin H, Zhou K, Jiang H, *et al*. Full integrity and freshness for cloud data [J]. *Future Generation Computer Systems*, 2018, **80**(3): 640-652.

[16] Kang B, Xu D. Secure electronic cash scheme with anonymity revocation [J]. *Mobile Information Systems*. 2016, Article ID 2620141, DOI: *http://dx.doi.org*/10. 1155/ 2016/ 2620141.

[17] Wang H, Wu Q, Qin B, *et al*. Identity-based remote data possession checking in public clouds [J]. *IET Information Security*, 2014, **8**(2) : 114-121.

[18] Zhang J, Dong Q. Efficient ID-based public auditing for the outsourced data in cloud storage [J]. *Information Sciences*, 2016, **343** (C):1-14.

[19] Yu Y, Xue L, Aub M, *et al*. Cloud data integrity checking with an identity-based auditing mechanism from RSA [J]. *Future Generation Computer Systems*, 2016, **62** (9): 85-91.

[20] Wei L, Zhu H, Cao Z, *et al*. Security and privacy for storage and computation in cloud computing [J]. *Information Sciences*, 2014, **258** (2): 371-386.

[21] He D, Wang H, Zhang J, *et al*. Insecurity of an identity-based public auditing protocol for the outsourced data in cloud storage [J]. *Information Sciences*, 2017, **375** (1) 48-53.

[22] Hou H, Yu J, Hao R. Research on an integrity auditing scheme based on algebraic signature in cloud storage [J]. *Netinfo Security*, 2017, **17** (10): 69-74.

[23] Yang T, Yu B, Wang H, *et al*. Cryptanalysis and improvement of Panda-public auditing for shared data in cloud and internet of things [J]. *Multimedia Tools and Applications*, 2017, **76**(19): 19411-19428.

[24] Kang B, Wang J, Shao D. Certificateless public auditing with privacy preserving for cloud-assisted wireless body area networks [J]. *Mobile Information Systems*, 2017, Article ID 2925465, DOI: *https://doi.org*/10.1155/2017/ 2925465.

[25] Bian G, Shao B, Cai W, *et al*. Research on multiple-replica integrity auditing method on supporting data dynamic updating in cloud environment [J]. *Netinfo Security*, 2017, **17** (10): 22-28.

[26] Kim D, Jeong I. Provably-secure public auditing with deduplication [J]. *KSII Transactions on Internet and Information systems*, 2017, **11**(4): 2219-2236.

[27] Shen W, Yu J, Yang G, *et al*. Access-authorizing and privacy-preserving auditing with group dynamic for shared cloud data [J]. *KSII Transactions on Internet and Information Systems*, 2017, **10**(7): 3319-3338.

[28] Zhang J, Li P. An efficient data integrity verification scheme for cloud storage [J]. *Netinfo Security*, 2017, **17** (3): 1-5.

[29] Yu H, Cai Y, Kong S, *et al*. Efficient and secure identity-based public auditing for dynamic outsourced data with proxy [J]. *KSII Transactions on Internet and Information Systems*, 2017, **11**(10): 5039-5061.

[30] Kim D, Kwon H, Hahn C, *et al*. Privacy-preserving public auditing for educational multimedia data in cloud computing [J]. *Multimedia Tools and Applications*, 2016, **75**(21): 13077-13091.

□