Hindawi
Security and Communication Networks
Volume 2018, Article ID 3205898, 9 pages
https://doi.org/10.1155/2018/3205898

Research Article

ID-Based Public Auditing Protocol for Cloud Data Integrity Checking with Privacy-Preserving and Effective Aggregation Verification

Baoyuan Kang,[1] **Lin Si**,[1] **Hong Jiang**,[2] **Chunqing Li**,[1] **and Mingming Xie**[1]

[1]*School of Computer Science and Software, Tianjin Polytechnic University, Tianjin 300387, China*
[2]*School of Management, Tianjin Polytechnic University, Tianjin 300387, China*

Correspondence should be addressed to Baoyuan Kang; baoyuankang@aliyun.com

Received 23 March 2018; Revised 7 May 2018; Accepted 2 October 2018; Published 1 November 2018

Academic Editor: Michael Vassilakopoulos

With the rapid development of cloud service, people with limited storage space can store their data files to the cloud and delete the file in their memory. However, the cloud service provider may change or partly delete user's file for his benefit. Therefore, it is necessary for the user to periodically check the data file integrity. Public auditing protocols are just designated for checking the data file integrity by an auditor on behalf of the user. Recently, based on ID-based cryptography many ID-based public auditing protocols for cloud data integrity checking are proposed. However, some existing protocols are subjected to forgery attack. Other existing protocols cannot preserve the privacy of the user, as the auditor can obtain user's file content through times of auditing the same file blocks. In this paper, we propose a new ID-based public auditing protocol for cloud data integrity checking with optimized structure, privacy-preserving, and effective aggregation verification. We also prove that the proposed protocol can resist forgery attack under the assumption that the Diffie-Hellman problem is hard. Furthermore, we compare our protocol with other ID-based auditing protocols.

1. Introduction

With the rapid development of cloud service, people with limited storage space like to store their large data file to the cloud, but cloud storage service also causes some security issues [1]. The cloud service provider may change or partly delete user's data file for his benefit. Therefore, it is necessary for the user to periodically check data file integrity. However, once the user transfers his file to the cloud, he will delete the file in his memory. Later, he cannot check the data integrity in conventional method. Public auditing protocols [2] are just designated for checking the data file integrity. In a public auditing protocol a data user firstly signs every block of his data file. Then the user sends his file and the signatures on file blocks to the cloud service provider and deletes the file locally. In the protocol there is an auditor who can periodically contact with cloud service provider to check the data file integrity on behalf of the user.

After first auditing protocols [2] many auditing protocols based on public key cryptographic system [3–33] were proposed. Recently, to eliminate public key management burden, a few public auditing protocols based on ID-based cryptographic system are proposed [16–22]. However, some existing ID-based public auditing protocols are subjected to malicious cloud server forgery attack [20]. Other existing ID-based public auditing protocols cannot preserve the privacy of the user as the auditor can obtain user's file content through times of auditing the same file blocks [19, 29]. A common ID-based public auditing protocol consists of six phases: setup, key extraction, tag generation, challenge, prove, and verify [17]. In challenge phase the auditor generates challenge information and sends it to the cloud servicer. When the cloud server returns the proof information of the date file integrity, the auditor verifies the proof information using the parameters from the cloud servicer. In our views, since the auditor has more computation and storage resources than the

data users, the auditor should store a few parameters and do more computations for the verification of the proof information. This may effectively resist the forgery attack from the cloud server. Another problem is that the existing ID-based public auditing protocols [17] lack necessary signature authentications on the messages between the data user, the cloud server, and the auditor. This problem leads to the lack of strictness in the protocols.

Based on the above understanding, we proposed a new ID-based public auditing protocol for data integrity checking. Our contributions are fivefold. Firstly we optimize the structure of ID-based public auditing protocol. We compress the six phases of common ID-based public auditing protocols into four phases. We also add necessary signature authentications. These measures make the proposed protocol more compact, clear, and rigorous. Secondly we use the method of aggregate signatures to make the proposed protocol more effective due to aggregation verification. Thirdly in the challenge and prove phase of the proposed protocol, to prove the proof information from the cloud server, the auditor must provide some parameters. This makes the protocol more secure than existing protocols in preventing forgery attack. The proposed protocol proves to be secure against forgery attack under the assumption that the Diffie-Hellman problem is hard. Fourthly the proposed protocol has privacy-preserving security features. The auditor cannot obtain any information of user's file content even through times of auditing the same file blocks.

The rest of the paper is organized as follows. In Section 2, we review bilinear pairing and computational Diffie-Hellman problem relevant to the security of the 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.

2. Preliminary

In this section, we briefly introduce the definitions of bilinear pairings and computational Diffie-Hellman (CDH) problem relevant to the security of the proposed protocol [17].

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 $e : G_1 \times G_1 \longrightarrow G_2$ be a pairing map which satisfies the following conditions.

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

$$e(P+Q,R) = e(P,R)\, e(Q,R) \tag{1}$$

and

$$e(P,Q+R) = e(P,Q)\, e(P,R). \tag{2}$$

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

(2) Nondegeneracy: 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. Given a generator P of an additive cyclic group G with order q and given (aP, bP) for unknown $a, b \in Z_q^*$, it is hard to compute abP.

3. The Proposed Protocol

As in [17], there are a data user, a cloud server, an auditor, and a private key generator (PKG) in an ID-based public auditing protocol. The cloud server is a semitrusted party. He might change or delete the data user's file for his benefit. Here we consider the cloud server as the only adversary to launch the forgery attack of the proof information for integrity checking. The new protocol consists of four algorithms: setup, key extraction, tag generation, challenge, and prove phase. The following is the detailed description of the proposed protocol. The two phases of setup and key extraction are the same as the general method of ID-based signatures [25].

Setup. Given a security parameter $k \in Z$, the algorithm 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 $e : G_1 \times G_1 \longrightarrow G_2$.

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

(3) Choose two cryptographic hash functions

$$\begin{aligned} H &: \{0,1\}^* \longrightarrow G_1, \\ h &: \{0,1\}^* \longrightarrow Z_q. \end{aligned} \tag{3}$$

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

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

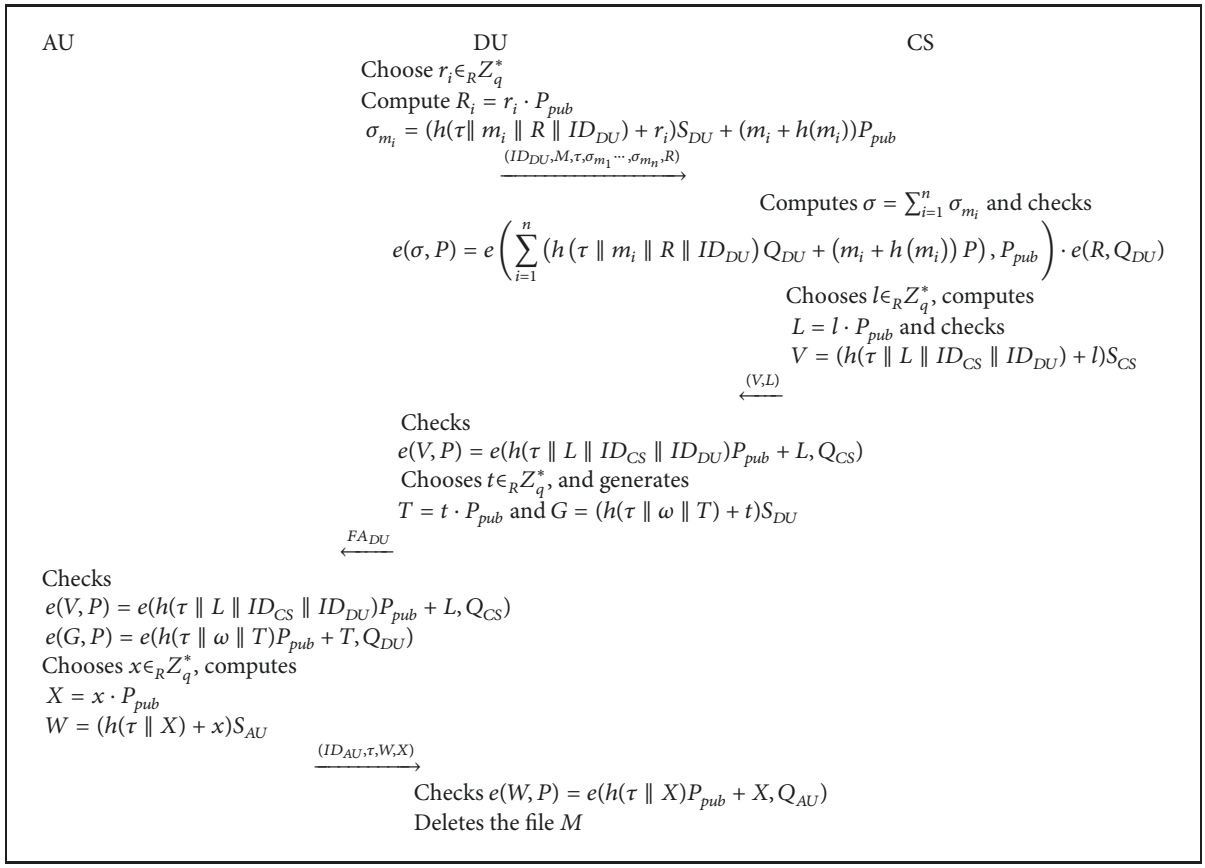
(1) Compute $Q_{ID} = H(ID) \in G_1$.

(2) Set the private key $S_{ID} = s \cdot Q_{ID}$, 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. This phase consists of five steps showing the messages transfer between the data user (DU) and both the cloud server (SC) and the auditor (AU). For a data file $M = m_1 \parallel \cdots \parallel m_n$, the data user (DU) selects a random file name, *name*, and lets $\tau = name \parallel n$ be the file tag. The tag generation phase is shown in Algorithm 1.

(1) $DU \longrightarrow CS : (ID_{DU}, M, \tau, \sigma_{m_1} \cdots, \sigma_{m_n}, R)$



Algorithm 1: The tag generation phase.

For τ and each file block m_i, DU chooses $r_i \in_R Z_q^*$, lets $R_i = r_i \cdot P_{pub}$, and computes

$$R = \sum_{i=1}^{n} R_i, \quad 1 \le i \le n,$$
$$\sigma_{m_i} = \left(h\left(\tau \| m_i \| R \| ID_{DU}\right) + r_i\right) S_{DU} + \left(m_i + h\left(m_i\right)\right) P_{pub}. \tag{4}$$

Then, DU sends $(ID_{DU}, M, \tau, \sigma_{m_1} \cdots, \sigma_{m_n}, R)$ to CS.

(2) $CS \longrightarrow DU : (V, L)$

CS computes

$$\sigma = \sum_{i=1}^{n} \sigma_{m_i} \tag{5}$$

and checks the following equation:

$$e\left(\sigma, P\right) = e\left(\sum_{i=1}^{n} \left(h\left(\tau \| m_i \| R \| ID_{DU}\right) Q_{DU} + \left(m_i + h\left(m_i\right)\right) P\right), P_{pub}\right) \cdot e\left(R, Q_{DU}\right). \tag{6}$$

If the equation holds, CS chooses $l \in_R Z_q^*$, computes

$$L = l \cdot P_{pub} \tag{7}$$

$$V = \left(h\left(\tau \| L \| ID_{CS} \| ID_{DU}\right) + l\right) S_{CS}, \tag{8}$$

sends (V, L) to DU, and then stores $(ID_{DU}, M, \tau, \sigma_{m_1} \cdots, \sigma_{m_n}, R)$.

(3) $DU \longrightarrow AU : FA_{DU} = (ID_{DU}, ID_{CS}, \tau, V, L, T, \omega, G, z_1, \cdots, z_n, R_1, \cdots, R_n)$

DU checks the following equation.

$$e\left(V, P\right) = e\left(h\left(\tau \| L \| ID_{CS} \| ID_{DU}\right) P_{pub} + L, Q_{CS}\right) \tag{9}$$

If it holds, DU chooses $t \in_R Z_q^*$ and generates the signature on information ω expressing the a request for auditing agency.

$$T = t \cdot P_{pub}$$
$$G = \left(h\left(\tau \| \omega \| T\right) + t\right) S_{DU} \tag{10}$$

sends

$$FA_{DU} = (ID_{DU}, ID_{CS}, \tau, V, L, T, \omega, G, z_1, \cdots, z_n, R_1, \cdots, R_n) \tag{11}$$

to AU. Here $z_i = h(\tau \| m_i \| R \| ID_{DU})$.

(4) $AU \longrightarrow DU : (ID_{AU}, \tau, W, X)$

AU checks following equations

$$e\left(V, P\right) = e\left(h\left(\tau \| L \| ID_{CS} \| ID_{DU}\right) P_{pub} + L, Q_{CS}\right)$$
$$e\left(G, P\right) = e\left(h\left(\tau \| \omega \| T\right) P_{pub} + T, Q_{DU}\right) \tag{12}$$

AU		CS
Choose a set $I \subseteq [1,n]$ and a number $a \in Z_q$		
Generate $Chall = [ID_{DU}, \tau, a, I]$		
	$\xrightarrow{Chall,(ID_{AU},\omega,G,T)}$	
		Checks $e(G,P) = e(h(\tau \parallel \omega \parallel T)P_{pub} + T, Q_{DU})$
		Produces set $\omega = \{(i, b_i)\}, i \in I$
		Computes $\sigma' = \sum_{i\in I} b_i \sigma_{m_i}$, $\mu = \sum_{i\in I} b_i (m_i + h(m_i))$
	$\xleftarrow{(\sigma',\mu)}$	
Compute $h = \sum_{i\in I} b_i h_i$, $R' = \sum_{i\in I} b_i R_i$.		
Check $e(\sigma', P) = e(zQ_{DU} + \mu P, P_{pub}) \cdot e(R', Q_{DU})$		

Algorithm 2: The challenge and prove phase.

If the equations hold, AU chooses $x \in_R Z_q^*$, computes

$$\begin{aligned} X &= x \cdot P_{pub} \\ W &= (h(\tau \parallel X) + x) S_{AU}, \end{aligned} \tag{13}$$

sends (ID_{AU}, τ, W, X) to DU for expressing that he accepts the auditing agency, and stores FA_{DU}.

(5) When DU receives the (W, X) from AU, DU checks the following equation:

$$e(W,P) = e\left(h(\tau \parallel X) P_{pub} + X, Q_{AU}\right). \tag{14}$$

If the equation holds, DU deletes the file M.

Challenge and Prove Phase. This phase consists of three steps showing the messages transfer between the auditor AU and the cloud server CS.

(1) $AU \longrightarrow CS : (Chall, ID_{AU}, \omega, G, T)$

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 $Chall = [ID_{DU}, \tau, a, I]$ and sends $Chall$ and (ID_{AU}, ω, G, T) to CS.

(2) $CS \longrightarrow AU : (\sigma', \mu)$

Upon receiving $Chall = [ID_{DU}, \tau, a, I]$ and (ID_{AU}, ω, G, T), CS checks the equation

$$e(G,P) = e\left(h(\tau \parallel \omega \parallel T) P_{pub} + T, Q_{DU}\right). \tag{15}$$

If the equation holds, CS finds $(ID_{DU}, M, \tau, \sigma_{m_1} \cdots, \sigma_{m_n}, R)$ and produces set $\omega = \{(i, b_i)\}, i \in I$.

Here, $b_i = a^i \bmod q$. Then using $M = m_1 \parallel \cdots \parallel m_n$ and $(\sigma_{m_1} \cdots, \sigma_{m_n})$, CS computes

$$\begin{aligned} \sigma' &= \sum_{i\in I} b_i \sigma_{m_i}, \\ \mu &= \sum_{i\in I} b_i (m_i + h(m_i)) \end{aligned} \tag{16}$$

and sends (σ', μ) to AU.

(3) Upon receiving the proof information (σ', μ), based on stored information FA_{DU}, AU computes

$$\begin{aligned} z &= \sum_{i\in I} b_i z_i, \\ R' &= \sum_{i\in I} b_i R_i. \end{aligned} \tag{17}$$

Then AU checks the following equation:

$$e\left(\sigma', P\right) = e\left(zQ_{DU} + \mu P, P_{pub}\right) \cdot e\left(R', Q_{DU}\right) \tag{18}$$

If the equation holds, AU accepts the proof.

The challenge and prove phases are shown in Algorithm 2.

4. Security of the Proposed Protocol

Theorem 1. *The proposed protocol is correct.*

Proof of Theorem 1. In order to save space, to prove the correctness of proof of the proposed protocol, we only prove the correctness of three representative equations.

Firstly, we show that the aggregate signature σ can be verified by equation

$$\begin{aligned} e(\sigma, P) = e\Bigg(&\sum_{i=1}^{n} \left(h(\tau \parallel m_i \parallel R \parallel ID_{DU}) Q_{DU} \right. \\ &\left. + (m_i + h(m_i)) P\right), P_{pub} \Bigg) \cdot e(R, Q_{DU}). \end{aligned} \tag{19}$$

In fact,

$$\begin{aligned} e(\sigma, P) &= e\left(\sum_{i=1}^{n} \left(\left(h(\tau \parallel m_i \parallel R \parallel ID_{DU}) + r_i\right) S_{DU} \right.\right. \\ &\quad \left.\left. + (m_i + h(m_i)) P_{pub} \right), P \right) \\ &= e\left(\sum_{i=1}^{n} h(\tau \parallel m_i \parallel R \parallel ID_{DU}) S_{DU}, P \right) e\left(\sum_{i=1}^{n} r_i S_{DU}, \right. \\ &\quad \left. P \right) \cdot e\left(\sum_{i=1}^{n} (m_i + h(m_i)) P_{pub}, P \right) \end{aligned}$$

$$
\begin{aligned}
&= e\left(\sum_{i=1}^{n} h\left(\tau \parallel m_i \parallel R \parallel ID_{DU}\right) Q_{DU}, P_{pub}\right) \\
&\cdot e\left(\sum_{i=1}^{n} r_i P_{pub}, Q_{DU}\right) \cdot e\left(\sum_{i=1}^{n} \left(m_i + h\left(m_i\right)\right) P, \right. \\
&\left. P_{pub}\right) \\
&= e\left(\sum_{i=1}^{n} \left(h\left(\tau \parallel m_i \parallel R \parallel ID_{DU}\right) Q_{DU}\right.\right. \\
&\left.\left. + \left(m_i + h\left(m_i\right)\right) P\right), P_{pub}\right) \cdot e\left(R, Q_{DU}\right).
\end{aligned} \tag{20}
$$

Secondly, the signature V can be verified by equation

$$
e\left(V, P\right) = e\left(h\left(\tau \parallel L \parallel ID_{CS} \parallel ID_{DU}\right) P_{pub} + L, Q_{CS}\right). \tag{21}
$$

In fact,

$$
\begin{aligned}
e\left(V, P\right) &= e\left(\left(h\left(\tau \parallel L \parallel ID_{CS} \parallel ID_{DU}\right) + l\right) S_{CS}, P\right) \\
&= e(\left(h\left(\tau \parallel L \parallel ID_{CS} \parallel ID_{DU}\right) S_{CS}, P\right) \\
&\quad \cdot e\left(lS_{CS}, P\right) \\
&= e(\left(h\left(\tau \parallel L \parallel ID_{CS} \parallel ID_{DU}\right) P_{pub}, Q_{CS}\right) \\
&\quad \cdot e\left(l \cdot P_{pub}, Q_{CS}\right) \\
&= e\left(h\left(\tau \parallel L \parallel ID_{CS} \parallel ID_{DU}\right) P_{pub} + L, Q_{CS}\right).
\end{aligned} \tag{22}
$$

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

$$
e\left(\sigma', P\right) = e\left(zQ_{DU} + \mu P, P_{pub}\right) \cdot e\left(R', Q_{DU}\right) \tag{23}
$$

In fact,

$$
\begin{aligned}
e\left(\sigma', P\right) &= e\left(\sum_{i \in I} b_i \sigma_{m_i}, P\right) = \prod_{i \in I} e\left(b_i \sigma_{m_i}, P\right) \\
&= \sum_{i \in I} e\left(b_i \left(h\left(\tau \parallel m_i \parallel R \parallel ID_{DU}\right) + r_i\right) S_{DU}\right. \\
&\left. + \left(m_i + h\left(m_i\right)\right) P_{pub}\right), P) = \sum_{i \in I} \left(e\left(b_i z_i S_{DU}, P\right)\right. \\
&\left. \cdot e\left(b_i r_i S_{DU}, P\right) \cdot e\left(b_i \left(m_i + h\left(m_i\right)\right) P_{pub}, P\right)\right) \\
&= e\left(\sum_{i \in I} b_i z_i S_{DU}, P\right) \cdot e\left(\sum_{i \in I} b_i r_i S_{DU}, P\right) \\
&\cdot e\left(\sum_{i \in I} b_i \left(m_i + h\left(m_i\right)\right) P_{pub}, P\right) = e\left(\sum_{i \in I} b_i z_i Q_{DU}, \right. \\
&\left. P_{pub}\right) \cdot e\left(\sum_{i \in I} b_i r_i P_{pub}, Q_{DU}\right) \\
&\cdot e\left(\sum_{i \in I} b_i \left(m_i + h\left(m_i\right)\right) P, P_{pub}\right) = e\left(zQ_{DU}, P_{pub}\right) \\
&\cdot e\left(R', Q_{DU}\right) \cdot e\left(\mu P, P_{pub}\right) = e\left(zQ_{DU} + \mu P, P_{pub}\right) \\
&\cdot e\left(R', Q_{DU}\right).
\end{aligned} \tag{24}
$$

□

Theorem 2. *If the CDH assumption is hard, then the proposed protocol is secure against existential forgery attack.*

Proof of Theorem 2. Similar to general proof thought, it will be shown that the challenger can solve the CDH problem when CS can provide forged valid proof information for the data integrity checking.

In the proof process, hash H and h are random oracles. For given CDH problem instance $(\alpha P, \beta P)$, the challenger sets system public key $P_{pub} = \alpha P$, user DU's private as $t_i(\beta P)$, $t_i \in_R Z_q$, for timely oracles.

Assuming that for the same challenge information $Chall = [ID_{DU}, \tau, a, I]$, CS produces two valid forged proof pieces of information $(\sigma_1'^*, \mu^*)$ and $(\sigma_2'^*, \mu^*)$ in two forgeries, then the following two equations hold.

$$
\begin{aligned}
e\left(\sigma_1'^*, P\right) &= e\left(z_1^* Q_{DU} + \mu^* P, P_{pub}\right) \cdot e\left(R'^*, Q_{DU}\right) \\
&= e\left(z_1^* \left(t_1 \alpha\left(\beta P\right)\right) + \mu P_{pub}, P\right) \\
&\quad \cdot e\left(R'^*, Q_{DU}\right) \\
e\left(\sigma_2'^*, P\right) &= e\left(z_2^* Q_{DU} + \mu^* P, P_{pub}\right) \cdot e\left(R'^*, Q_{DU}\right) \\
&= e\left(z_2^* \left(t_2 \alpha\left(\beta P\right)\right) + \mu P_{pub}, P\right) \\
&\quad \cdot e\left(R'^*, Q_{DU}\right)
\end{aligned} \tag{25}
$$

Then,

$$
\begin{aligned}
\sigma_2'^* - \sigma_1'^* &= \left(z_2^* t_2 - z_1^* t_1\right) \left(\alpha\beta P\right) \\
\left(ab\right) P &= \left(z_2^* t_2 - z_1^* t_1\right)^{-1} \left(\sigma_2'^* - \sigma_1'^*\right).
\end{aligned} \tag{26}
$$

□

Theorem 3. *In the proposed protocol, the author cannot derive any information of data file content.*

Proof of Theorem 3. In the whole auditing procedure, the author AU only obtains messages

$$
\begin{aligned}
&FA_{DU} \\
&= \left(ID_{DU}, ID_{CS}, \tau, V, L, T, \omega, G, z_1, \cdots, z_n, R_1, \cdots, R_n\right)
\end{aligned} \tag{27}
$$

from DU and

$$
\begin{aligned}
\sigma' &= \sum_{i \in I} b_i \sigma_{m_i}, \\
\mu &= \sum_{i \in I} b_i \left(m_i + h\left(m_i\right)\right)
\end{aligned} \tag{28}
$$

Table 1: Comparison of features.

	F1	F2	F3	F4	F5	F6	F7
Wang et al. [16]	No	No	No	No	No	Yes	No
Zhang et al. [17]	No -	Yes	No	No	No	No	Yes
Ours	Yes	Yes	Yes	Yes	Yes	Yes	Yes

F1: file tag verification, F2: block tag verification, F3: accept storage verification, F4: application agent verification, F5: accept auditing verification, F6: unforgeability, F7: privacy-preserving.

Table 2: Required communication number.

	P1	P2	P3
Wang et al. [16]	1	1	2
Zhang et al. [17]	1	1	2
Ours	1	4	2

P1: key extraction phase, P2: tag generation phase, P3: challenge and prove phase.

from CS. However, (τ, V, L) and (T, ω, G) are signatures irrelevant to the file content. $(R_1, \cdots, R_n)$ is also irrelevant to the file content. $(z_1, \cdots, z_n)$ and σ' are relevant to the file content, but the file content is protected by hash function.

It is impossible for AU to obtain block m_i from equation $\mu = \sum_{i \in I} b_i(m_i + h(m_i))$. Even through times of auditing the same file blocks, AU also does not obtain any block of the file. Because $\mu = \sum_{i \in I} b_i(m_i + h(m_i))$ is not linear equation of m_i. Therefore, AU cannot derive any information about DU's data file content during the whole auditing procedure. □

5. Comparisons

In this section, the comparisons of the proposed protocol with other two ID-based auditing protocols [16, 17] are shown. The comparison results of the security features, communication number, and computation costs are shown, respectively, in Tables 1, 2, and 3.

From Table 2, in tag generation phase, the communication number in the proposed protocol is obviously higher than the other two protocols. This is caused by the following two facts. One is that in our protocol when the cloud server accepts the data file from cloud user, the cloud server must return an 'accept service' authentication information to the user. The other one is that the date user must send information to the auditor for begging auditing agency, and once accepting the auditing agency, the auditor also sends a response to the user.

Since there is no detail file tag signature algorithm description in [16], in Table 3, we only compare the computation costs of the parts common to the proposed protocol with [17] in key extraction phase, block tag generation phase, challenge phase, and prove phase. In addition, we mainly count the exponential operation, scalar multiplication, hash computation, and bilinear pairings operation. Also we assume in challenge and prove phase that the challenging blocks number is |I|. The computation cost of Zhang et al.'s protocol [17] is (4n+2|I|+2)H+(6n+4|I|+2)S+(3n+3)B+(n+|I|)E. However, the computation cost of our protocol is (4n+|I|+8)H+(5n+2|I|+13)S+14B+|I|E. According to [33], $H \approx 23t$, $S \approx 29t$, $E \approx 21t$, $B \approx 1440t$. Here, t represents the time cost of a modular multiplication in z_q. Then, the computation cost of Zhang et al.'s protocol is about $(4607n + 183|I| + 4424)t$. However, the computation cost of our protocol is about $(237n+102|I|+20721)t$. The computation cost of Zhang et al.'s protocol in tag generation phase is about $4607nt$. However, the computation cost of our protocol in tag generation phase is about $(237n + 13049)t$.

We simulate the computational cost of our protocol and Zhang et al.'s protocol [17] on a Mac OS High Sierra system with an Intel Core i7 at 2.9 GHz and 16-GB RAM. The algorithms are implemented using the pairing-based cryptography (PBC) library version 0.5.14. When the file is 1024 Bytes, the comparison of computation cost in tag generation phase between our protocol and Zhang et al.'s protocol is shown in Figure 1. The whole computation costs of our protocol and Zhang et al.'s protocol are shown in Figures 2 and 3, respectively. When the number n of the blocks of the file is 48, the comparison of computation cost between our protocol and Zhang et al.'s protocol is shown in Figure 4. When the file is large and the number of its blocks is correspondingly large, our protocol needs significant low computation cost.

Another need for comparison and explanation is the relationship between our protocol and the one in [25]. In [25] a certificateless public auditing protocol with privacy-preserving for cloud-assisted wireless body area networks was proposed. Since the same issue of public auditing is researched in the two protocols, there are some unavoidable similarities in structure and concern. However, the protocol in [25] is based on certificateless public cryptography, while the protocol in this paper is based on ID-based public cryptography. There is a great difference in the concrete structure of the two protocols. In the tag generation phase of the protocol in this paper aggregation verification technology is used to greatly reduce the amount of computation. Therefore, on the whole, the efficiency and design concept of the protocol in this paper are higher than the one in [25].

6. Conclusion

In this paper, we propose a new ID-based public auditing protocol for cloud data integrity checking. The proposed protocol has not only optimized structure but also effective aggregation verification to reduce the computation cost. Furthermore, the proposed protocol has privacy-preserving feature as the auditor cannot obtain any information of user's file content even through times of auditing the same file blocks. We prove that the proposed protocol can resist forgery attack under the assumption that the Diffie-Hellman problem

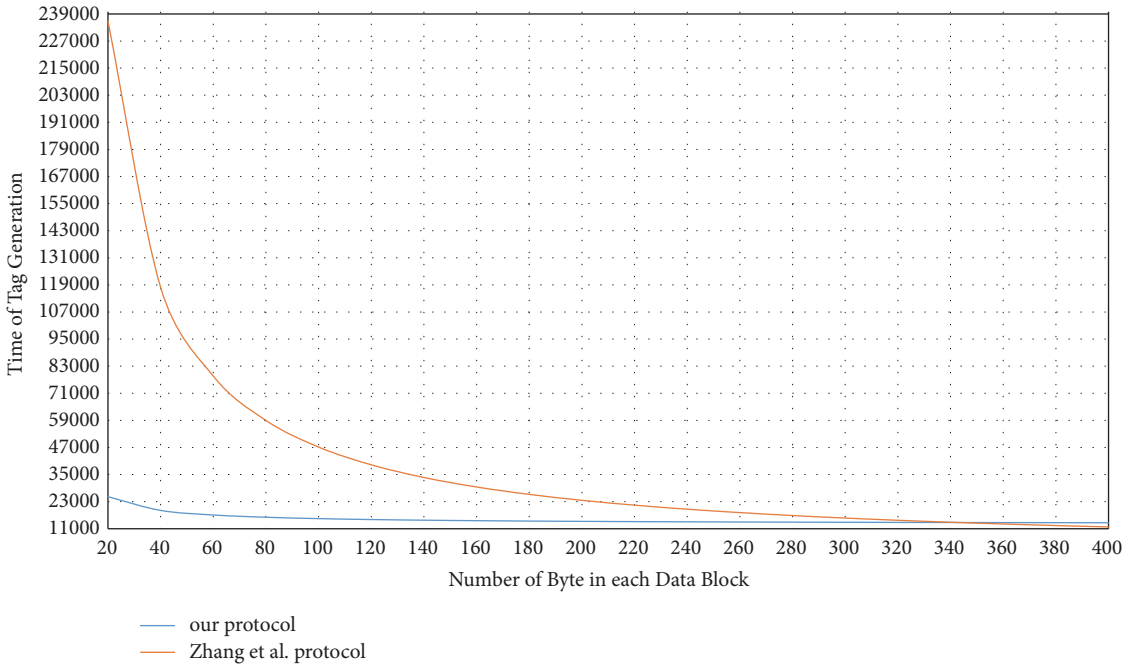


Figure 1: The computation time of tag generation for 1024-Byte data.

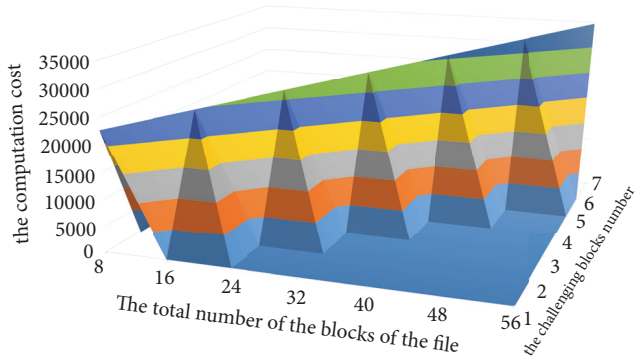


Figure 2: The computation cost of our protocol.

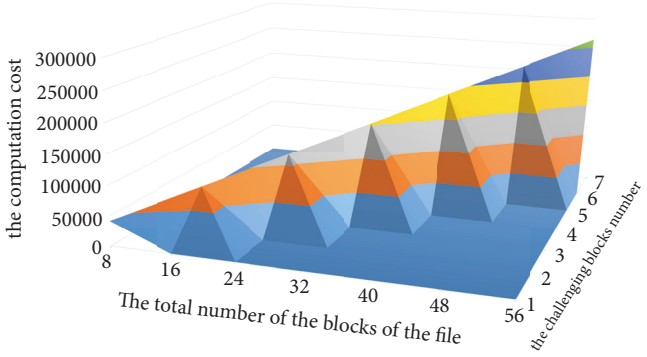


Figure 3: The computation cost of Zhang et al.'s protocol.

Table 3: Comparison of computation costs.

	P1	P2	P3	P4
Zhang et al. [17]	2H+2S	4nH+6nS+3nB+ nE	—I—H+2—I—S+—I—E	—I—H+2—I—S+3B
Ours	H+S	(4n+7)H+(5n+12)S+11B	—I—H +—I—S+—I—E	—I—S+3B

P1: key extraction phase, P2: block tag generation phase, P3: prove phase, P4: verify phase; E: exponential operation and its time cost, S: scalar multiplication and its time cost, H: hash computation and its time cost, B: bilinear pairing and its time cost.

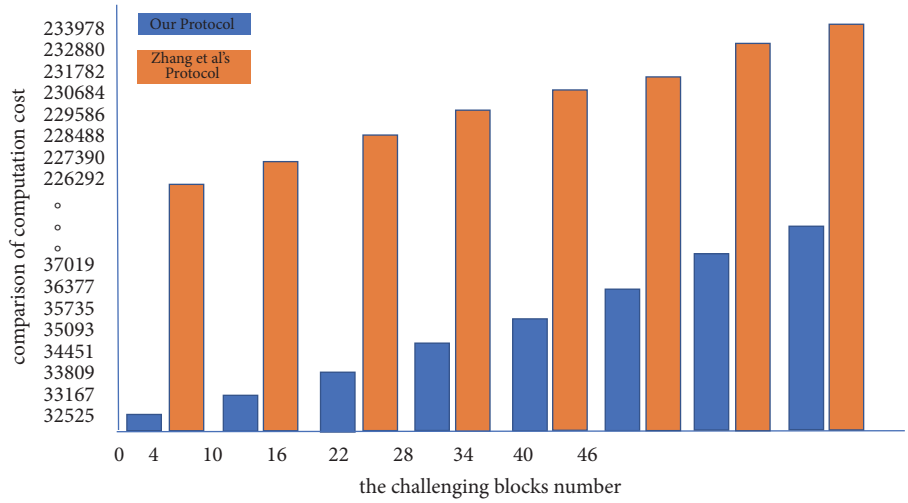


Figure 4: The comparison of computation cost between our protocol and Zhang et al.'s protocol.

is hard. We also compare the proposed protocol with other ID-based auditing protocols. The proposed protocol is shown to be more secure and efficient in computation cost.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the Applied Basic and Advanced Technology Research Programs of Tianjin (No. 15JCYBJC15900) and the National Natural Science Foundation of China (No. 51378350).

References

[1] G. Ateniese, R. Burns, R. Curtmola et al., "Provable data possession at untrusted stores," in *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07)*, pp. 598–609, Virginia, Va, USA, November 2007.

[2] G. Ateniese, S. Kamara, and J. Katz, "Proofs of storage from homomorphic identification protocols," in *Proceedings of International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*, pp. 319–333, Springer-Verlag, London, UK, 2009.

[3] G. Yang, J. Yu, W. Shen, Q. Su, Z. Fu, and R. Hao, "Enabling public auditing for shared data in cloud storage supporting identity privacy and traceability," *The Journal of Systems and Software*, vol. 113, pp. 130–139, 2016.

[4] R. Swathi and T. Subha, "Enhancing data storage security in Cloud using Certificateless public auditing," in *Proceedings of the 2nd International Conference on Computing and Communications Technologies, ICCCT 2017*, pp. 348–352, India, February 2017.

[5] L. Wu, J. Wang, N. Kumar, and D. He, "Secure public data auditing scheme for cloud storage in smart city," *Personal and Ubiquitous Computing*, vol. 21, no. 5, pp. 949–962, 2017.

[6] M. Swapnali and C. Sangita, "Third Party Public Auditing Scheme for Cloud Storage," in *Proceedings of International Conference on Communication , Computing and Virtualization, ICCCV*, vol. 79, pp. 69–76, 2016.

[7] R. S. Anjali and A. Ravikumar, "Preserving privacy in public auditing for shared cloud data," in *Proceedings of the 2016 International Conference on Inventive Computation Technologies, ICICT 2016*, India, August 2016.

[8] S. Singh and S. Thokchom, "Public integrity auditing for shared dynamic cloud data," in *Proceedings of the 6th International Conference on Smart Computing and Communications, ICSCC 2017*, pp. 698–708, India, December 2017.

[9] T. Subha and S. Jayashri, "Public auditing scheme for data storage security in cloud computing," *Journal of Information Science and Engineering*, vol. 33, no. 3, pp. 773–787, 2017.

[10] T. Jiang, X. Chen, and J. Ma, "Public integrity auditing for shared dynamic cloud data with group user revocation," *IEEE Transactions on Computers*, vol. 65, no. 8, pp. 2363–2373, 2016.

[11] W. Shen, J. Yu, R. Hao, and X. Wang, "A public cloud storage auditing scheme for resource-constrained clients," *International Journal of High Performance Systems Architecture*, vol. 6, no. 3, pp. 121–130, 2016.

[12] V. Saranya, R. S. Kumar, and T. Nalini, "A Study on the Public Auditing Mechanisms for Privacy Preserving and Maintaining Data Integrity in Cloud Computing," *Journal of Database Theory and Application*, vol. 9, no. 6, pp. 103–108, 2016.

[13] J. Zhang, X. Zhao, and W. Zhen, "S2PAD: Secure self-certified public auditing for data integrity in cloud storage and its extension," *International Journal of Information and Communication Technology*, vol. 12, no. 1-2, pp. 113–130, 2018.

[14] D. He, S. Zeadally, and L. Wu, "Certificateless public auditing scheme for cloud-assisted wireless body area networks," *IEEE Systems Journal*, vol. 12, no. 5, pp. 64–73, 2018.

[15] S. Anbuchelian, C. M. Sowmya, and C. Ramesh, "Efficient and secure auditing scheme for privacy preserving data storage in cloud," *Cluster Computing*, pp. 1–9, 2017.

[16] H. Wang, J. Domingo-Ferrer, Q. Wu, and B. Qin, "Identity-based remote data possession checking in public clouds," *IET Information Security*, vol. 8, no. 2, pp. 114–121, 2014.

[17] J. Zhang and Q. Dong, "Efficient ID-based public auditing for the outsourced data in cloud storage," *Information Sciences*, vol. 343-344, pp. 1–14, 2016.

[18] Y. Yu, L. Xue, and M. H. Au, "Cloud data integrity checking with an identity-based auditing mechanism from RSA," *Future Generation Computer Systems*, vol. 62, pp. 85–91, 2016.

[19] L. Wei, H. Zhu, Z. Cao et al., "Security and privacy for storage and computation in cloud computing," *Information Sciences*, vol. 258, pp. 371–386, 2014.

[20] D. He, H. Wang, J. Zhang, and L. Wang, "Insecurity of an identity-based public auditing protocol for the outsourced data in cloud storage," *Information Sciences*, vol. 375, pp. 48–53, 2017.

[21] L.-B. Wu, J. Wang, D.-B. He, and M.-K. Khan, "Cryptanalysis of an identity-based public auditing protocol for cloud storage," *Frontiers of Information Technology and Electronic Engineering*, vol. 18, no. 12, pp. 1972–1977, 2017.

[22] X. Zhang, C. Xu, and C. Jin, "Enabling identity-based cloud storage public auditing with quantum computers resistance," *International Journal of Electronic Security and Digital Forensics*, vol. 8, no. 1, pp. 82–98, 2016.

[23] D. Kim, H. Kwon, C. Hahn, and J. Hur, "Privacy-preserving public auditing for educational multimedia data in cloud computing," *Multimedia Tools and Applications*, vol. 75, no. 21, pp. 13077–13091, 2016.

[24] T. Yang, B. Yu, H. Wang, J. Li, and Z. Lv, "Cryptanalysis and improvement of Panda—public auditing for shared data in cloud and internet of things," *Multimedia Tools and Applications*, vol. 76, pp. 19411–19428, 2015.

[25] B. Kang, J. Wang, and D. Shao, "Certificateless Public Auditing with Privacy Preserving for Cloud-Assisted Wireless Body Area Networks," *Mobile Information Systems*, vol. 2017, Article ID 2925465, 5 pages, 2017.

[26] H. Yu, Y. Cai, S. Kong, Z. Ning, Fei. Xue, and H. Zhong, "Efficient and Secure Identity-Based Public Auditing for Dynamic Outsourced Data with Proxy," *KSII Transactions on Internet and Information systems*, vol. 11, pp. 5039–5061, 2017.

[27] D. Kim and I. R. Jeong, "Provably-secure public auditing with deduplication," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 4, pp. 2219–2236, 2017.

[28] W. Shen, J. Yu, G. Yang, Y. Zhang, Z. Fu, and R. Hao, "Access-authorizing and privacy-preserving auditing with group dynamic for shared cloud data," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 7, pp. 3319–3338, 2016.

[29] B. Kang, J. Wang, and D. Shao, "Attack on Privacy-Preserving Public Auditing Schemes for Cloud Storage," *Mathematical Problems in Engineering*, vol. 2017, Article ID 8062182, 6 pages, 2017.

[30] Y. Li, Y. Yu, B. Yang, G. Min, and H. Wu, "Privacy preserving cloud data auditing with efficient key update," *Future Generation Computer Systems*, vol. 78, pp. 789–798, 2018.

[31] L. Xue, J. Ni, Y. Li, and J. Shen, "Provable data transfer from provable data possession and deletion in cloud storage," *Computer Standards & Interfaces*, vol. 54, pp. 46–54, 2017.

[32] H. Jin, K. Zhou, H. Jiang, D. Lei, R. Wei, and C. Li, "Full integrity and freshness for cloud data," *Future Generation Computer Systems*, vol. 80, pp. 640–652, 2016.

[33] C.-I. Fan, W.-Z. Sun, and V. S. Huang, "Provably secure randomized blind signature scheme based on bilinear pairing," *Computers & Mathematics with Applications. An International Journal*, vol. 60, no. 2, pp. 285–293, 2010.