An Efficient ID-Based Message Recoverable Privacy-Preserving Auditing Scheme ⋆

Mehmet Sabır Kiraz, İsa Sertkaya, Osmanbey Uzunkol

Mathematical and Computational Sciences
TÜBİTAK BİLGEM, Turkey
{mehmet.kiraz, isa.sertkaya, osmanbey.uzunkol}@tubitak.gov.tr

Abstract. One of the most important benefits of public cloud storage is outsourcing of management and maintenance with easy accessibility and retrievability over the internet. However, outsourcing data on the cloud brings new challenges such as integrity verification and privacy of data. More concretely, once the users outsource their data on the cloud they have no longer physical control over the data and this leads to the integrity protection issue. Hence, it is crucial to guarantee proof of data storage and integrity of the outsourced data. Several pairing-based auditing solutions have been proposed utilizing the Boneh-Lynn-Shacham (BLS) short signatures. They basically provide a desirable and efficient property of non-repudiation protocols. In this work, we propose the first ID-based privacy-preserving public auditing scheme with message recoverable signatures. Because of message recoverable auditing scheme, the message itself is implicitly included during the verification step that was not possible in previously proposed auditing schemes. Furthermore, we point out that the algorithm suites of existing schemes is either insecure or very inefficient due to the choice of the underlying bilinear map and its baseline parameter selections. We show that our scheme is more efficient than the recently proposed auditing schemes based on BLS like short signatures.

Keywords: Data storage, public auditability, privacy preserving, message recoverable signatures, bilinear maps

1 Introduction

Cloud service providers lead to rapidly increasing data storage in the cloud servers. They give opportunities to edit and share the data on the fly, while enabling the users to work with arbitrarily large amount of data without downloading into their local machines. Such an elasticity enables users also to perform expensive computation like big data analysis or search on the cloud. Even if the cloud service providers build powerful, reliable and maintainable infrastructures internal and external breach may still happen (e.g., [3,4,14,19,22]). In particular,

⋆ A preliminary version has appeared in the 13th Annual IEEE Conference on Privacy, Security and Trust (PST 2015).

cloud storage solutions demand new data security and privacy policies [23, 24]. For example, the cloud provider may behave unfaithful by means of modifying and deleting the data because the control over the remotely stored data is limited [5, 27, 32, 33, 36, 37, 39].

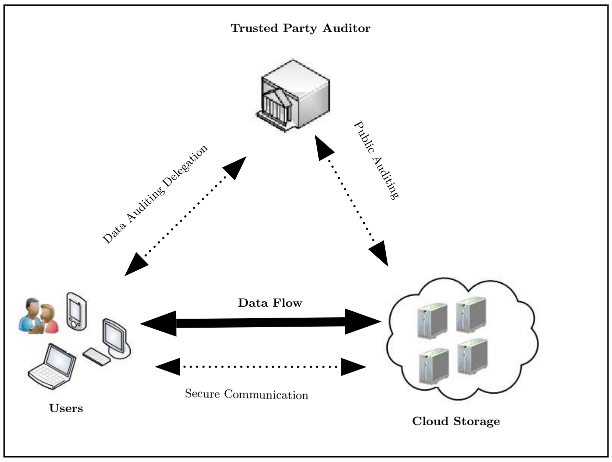

Fig. 1. Public Auditing Model

Public auditing is an assurance of the integrity for outsourced data. To overcome this challenge, the trivial solution is to download the whole outsourced data locally, and to check its integrity. However, such a solution is rapidly infeasible for big data. Trusted party auditor is introduced in order to eliminate the online involvement of users from auditing, to perform verification, and to minimize computational burdens (which can be important to scale the cloud computing) [2, 36]. Henceforth, all the existing auditing schemes in the cloud include three entities: 1) the data owner who outsources her data, 2) the cloud service provider with large amount of storage space and computation power and 3) a honest-but-curious third party auditor that is only responsible for auditing tasks on behalf of the users. Note that the auditor is assumed to be a stateless machine for usability concerns. An illustration of a typical public auditing scheme can be seen in Figure 1.

A typical privacy-preserving audit scheme has four main steps; namely setup, signature generation, challenge based proof generation and verification. Once the system parameters are generated, the user signs her data and sends it to the cloud storage. Cloud storage subsequently verifies the signatures and stores the data with the corresponding signatures. Later, the trusted party auditor challenges the cloud storage on behalf of users. Upon receiving the challenges the cloud storage prepares a proof using the challenge, the data and the corresponding signatures. Finally, the trusted party auditor verifies the proof. There are various attacks to be considered in this scenario. On the one hand, a malicious server may apply replace attacks (server may arbitrarily behave, and disobey the challenge and use another valid data block), replay attacks (server generates proofs without querying the actual data), or forgery attacks (server may forge the signatures). On the other, since the auditor is honest-but-curious it can internally try to gather extra information about the data. Therefore, the existence of honest-but-curious trusted party auditor for integrity checking of remotely stored data on the cloud requires additional privacy enhancing solutions. However, conventional cryptographic primitives alone (like symmetric encryption or hash functions) do not suffice to ensure data integrity and privacy on the cloud because these primitives lack certain level of malleability.

Related work.

For a comprehensive survey and taxonomy on remote data auditing we refer to [28, 39]. As stated in [28], data auditing approaches can be grouped into three different models; provable data possession-based (PDP) [5, 15], proof of retrievability-based (POR) [21, 26], and proof of ownership-based (POW) [25]. Studies on remotely stored data auditing problem dates back to Ateniese *et al.*'s paper in which RSA-based homomorphic tags in PDP model for static data storage scenario is first proposed [5]. Later, they proposed an enhanced PDP model for limited dynamical data storage scenario [6].

Based on [39], it is possible to group auditing methods regarding the utilized methods: Message Authentication Code based [26], RSA-based homomorphic [20,40], the Boneh-Lynn-Shacham (BLS) signatures based homomorphic [9] and algebraic signatures based [11, 41]. Moreover, approaches may also differ for the underlying scenarios; the stored data is assumed to be static [5], dynamic [6, 35], shared or version controlled [30].

Based on Wang *et al.*'s proposals [32–34], in [37] Worku *et al.* proposed a more efficient auditing protocol based on a variant of the Boneh-Lynn-Shacham (BLS) signature [9]. But later, in [12, 38], the authors give certain linear attacks to the verification phase of Worku *et al.*'s auditing protocol. We would like to highlight that these attack scenarios are arguable since the proof of data possession by the cloud includes indirectly the valid data and corresponding signatures. Still, these attacks point out security flaws with respect to replace and replay attacks. In particular, an adversary interacts both with the cloud server and with the auditor, and can subsequently manipulate proof generation and verification steps.

An ID-based certificateless scheme is proposed in [31]. However, their parameter selection in the setup phase uses ordinary elliptic curves in Type 1 bilinear map setting [13]. In the next section, we explain that usage of ordinary elliptic curves in Type 1 setting is very inefficient.

Our contributions.

To the best of our knowledge, all the existing auditing schemes try to verify the data integrity with corresponding signatures in case of a dispute. However, if a proof verification step fails, these schemes are not capable of message recovery. This is the starting point of this work. In order to overcome this problem, we propose an efficient privacy-preserving public auditing scheme based on message recoverable signatures. Hence, whenever the proof verification fails, the valid signature itself will be sufficient to recover the original message. The proposed auditing protocol will be utilizing a modified version of Tso *et al.*'s ID-based message recoverable signature scheme [29]. Tso *et al.*'s scheme has a deficiency due to recent quasi-polynomial discrete logarithm attacks [1, 7, 18]. In order to make Tso *et al.*'s scheme realizable and resistant, we modify their scheme by utilizing a Type 3 bilinear map and without changing its security margins. This approach has the efficiency advantage (due to smaller group sizes) when compared to the existing auditing schemes utilizing a variant of the Boneh-Lynn-Shacham (BLS) short signatures. We prove the security of our protocol by considering each cases, i.e. malicious users, malicious cloud servers and honest-but-curious auditor. We finally compare the complexity of our protocol with the existing auditing schemes.

Roadmap. The rest of the paper is organized as follows: In Section 2, we give the necessary preliminaries about bilinear maps and our notation. In Section 3, we define the system and the security model. Next, in Section 4, we present our proposed scheme and provide security analysis in Section 5, respectively. We further show the practicality of our schemes in Section 6. Finally, Section 7 concludes the paper.

2 Preliminaries

2.1 Bilinear Maps and DLP Security

Auditing schemes are mostly realized using certain malleability property of the underlying signature primitives. Bilinear maps are one of the good candidate for enabling this property. Efficient construction of bilinear maps uses Weil, Tate or optimal pairings of abelian varieties (e.g. elliptic curves) having reasonably small embedding degrees [13]. Abelian varieties of dimension ≤ 2 (elliptic curves or jacobians of hyperelliptic curves of genus 2 [13]), are the main mathematical objects. Although bilinear maps are used as a black box in our scheme, we revisit preliminaries of the pairing types and pairing-friendly elliptic curves. These choices effect not only the security but also the complexity of the proposed scheme. Unless otherwise stated, we follow the lines of [8, Chapter IX] for the properties of pairings.

Let $(\mathbf{G}_1, +)$ and $(\mathbf{G}_2, +)$ be two additive cyclic groups of order q with $\mathbf{G}_1 =< Q >$ and $\mathbf{G}_2 =< P >$, $(\mathbf{G}_3, \cdot)$ be a multiplicative cyclic group of order q, where q is a prime number and $0_{\mathbf{G}_1}, 0_{\mathbf{G}_2}$ and $1_{\mathbf{G}_3}$ are the identity elements of the groups $\mathbf{G}_1$, $\mathbf{G}_2$ and $\mathbf{G}_3$, respectively. Assume that Discrete Logarithm Problem (DLP) is hard in both $\mathbf{G}_1$ and $\mathbf{G}_2$ (i.e., given a random $y \in \mathbf{G}_1$ (or $\in \mathbf{G}_2$), it computationally infeasible to find an integer $x \in \mathbb{Z}$ such that $y = g^x$). If it is clear from the context we write 0 for the identity elements of $\mathbf{G}_1$, $\mathbf{G}_2$ and 1 for $\mathbf{G}_3$. A *bilinear map* is a map $e: \mathbf{G}_1 \times \mathbf{G}_2 \to \mathbf{G}_3$ satisfying the following properties:

- **Bilinearity:** For all $P_1, Q_1 \in \mathbf{G}_1, P_1', Q_1' \in \mathbf{G}_2$, e is a group homomorphism in each component, i.e.
  1. $e(P_1 + Q_1, P_1') = e(P_1, P_1') \cdot e(Q_1, P_1')$,
  2. $e(P_1, P_1' + Q_1') = e(P_1, P_1') \cdot e(P_1, Q_1')$.
- **Non-degeneracy:** e is non-degenerate in each component, i.e.
  1. For all $P \in \mathbf{G}_1$, $P \neq 0$, there is an element $Q \in \mathbf{G}_2$ such that $e(P, Q) \neq 1$,
  2. For all $Q \in \mathbf{G}_2$, $Q \neq 0$, there is an element $P \in \mathbf{G}_1$ such that $e(P, Q) \neq 1$.
- **Computability:** There exists an algorithm which computes the bilinear map e efficiently.

Bilinear maps can be realized by finding a suitable pairing-friendly elliptic curve E (or more generally an abelian variety) over a finite field $\mathbb{F}_l$. Then, appropriate subgroups $\mathbf{G}_1$ and $\mathbf{G}_2$ are constructed. The group $\mathbf{G}_3$ is a subgroup of $\mathbb{F}_{l^k}$, where k is the embedding degree of E [16]. We revisit the types of realizations of bilinear maps due to its security and efficiency for our auditing scheme. There are essentially 3 types of bilinear maps [17, pp. 3115]:

- **Type 1:** ($\mathbf{G}_1 = \mathbf{G}_2$) $\mathbf{G}_1$ is generally determined by a supersingular elliptic curve which is typically defined over a finite field of characteristic 2 and 3.
- **Type 2:** ($\mathbf{G}_1 \neq \mathbf{G}_2$ and there is an efficiently computable homomorphism $\phi : \mathbf{G}_2 \to \mathbf{G}_1$) In this case, $\mathbf{G}_1$ and $\mathbf{G}_2$ can be realized by using any elliptic curve with small embedding degree. The disadvantage of Type 2 pairings is that there exists no random sampling algorithm from $\mathbf{G}_2$ yielding to a secure hash function which maps arbitrary elements to $\mathbf{G}_2$, [17, pp. 3119].
- **Type 3:** ($\mathbf{G}_1 \neq \mathbf{G}_2$ and there exists no efficiently computable homomorphism $\phi : \mathbf{G}_2 \to \mathbf{G}_1$) Like in Type 2 pairings, $\mathbf{G}_1$ and $\mathbf{G}_2$ are determined by constructing an elliptic curve with small embedding degree. Note that a general method transforming protocols from Type 2 to Type 3 is given in [10, Section 5].

Type 3 pairings are the most efficient realization of bilinear maps due to their efficiency (less group operations, more efficient membership testing and bandwidth) [10, pp. 1313]. Furthermore, the protocols based on Type 1 pairings are mostly insecure due to recent quasi-polynomial algorithms on solving discrete logarithms in finite fields and their implications to the weakness of discrete logarithms of supersingular elliptic curves [1, 7, 18]. Additionally, if one uses supersingular elliptic curves over large prime fields, the protocol will be very inefficient since we have the embedding degree $k = 2$. Type 3 bilinear maps realize more efficient protocols since it is possible to have embedding degree larger than 2 (e.g. typically for Barreto-Naehrig (BN) curves with $k = 12$ are chosen for optimal efficiency [16]). In this case, the same level of security can be assured with much smaller key sizes.

Because of security and efficiency reasons as described above, we deliberately use a Type 3 version of message recoverable signature scheme of Tso *et al.* in our proposed scheme [29]. We note that the security assumptions and the proof of security remain unchanged (because no specific property of Type 1 bilinear maps is used in the security proof). The only difference will be the replacement of the precomputed value $\mu := e(P, P)$ with $\mu := e(Q, P)$ to adapt the scheme into a Type 3 setting.

2.2 Notations

We fix a Type 3 pairing $e: \mathbf{G}_1 \times \mathbf{G}_2 \rightarrow \mathbf{G}_3$ for the rest of the paper. We mainly follow Tso *et al.*'s notation as follow [29]:

- $\mathbf{G}_1 =< Q >$, $\mathbf{G}_2 =< P >$ of prime order q. Let $|q| = \ell_1 + \ell_2$ be the bit length of q.
- μ : the value of $e(Q, P)$.
- $a \parallel b$: a concatenation of two bit strings a and b.
- $\oplus$: XOR computation in the binary system.
- $[x]_{10}$: the decimal notation of $x \in \{0,1\}^*$.
- $[x]_2$: the binary notation of $x \in \mathbb{N}$.
- $_{\ell_2}|\beta|$: the first ℓ_2 bits of β from the left hand side.
- $|\beta|_{\ell_1}$: the first ℓ_1 bits of β from the right hand side.
- $H : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$: a cryptographic one-way hash function.
- $H_1 : \{0,1\}^* \rightarrow \{0,1\}^{\ell_1+\ell_2}$: a cryptographic one-way hash function.
- $F_1 : \{0,1\}^{\ell_1} \rightarrow \{0,1\}^{\ell_2}$: a cryptographic one-way hash function.
- $F_2 : \{0,1\}^{\ell_2} \rightarrow \{0,1\}^{\ell_1}$: a cryptographic one-way hash function.

3 Security Model

In the public auditing scheme there are three different entities as follows:

- A cloud server (S) is a data storage owner to provide data storage services for its users to create, store, update and request for retrievability. S is assumed to have a large storage space and large computation resources.
- A user (C) is a client who has large amount of data to be stored in the cloud. Furthermore, C is assumed to delegate the checkability property to a third party whether her data is indeed stored in the cloud correctly.
- A trusted party auditor (TPA) is assumed to be stateless (memoryless) which has expertise and capabilities to check the cloud storage reliability and validity. TPA has always an interaction with S to check the integrity and validity of users' data.

In the proposed security model, C and S are assumed to be malicious which may arbitrarily deviate from the protocol whereas TPA is assumed to be honest-but-curious (semi-honest) to asses the reliability of S on behalf of the users whenever needed. Hence, S and TPA are deployed by different organizations and are assumed not to collude each other.

The proposed privacy-preserving public auditing model satisfies the following security properties.

Setup:

On input a security parameter $\kappa \in \mathbb{N}$, the algorithm
- outputs a random number $s \in \mathbb{Z}_q^*$ as Key Generator Center (KGC)'s private key
- sets $P_{pub} = sP$ as KGC's public key.

System parameters: params $= \{\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3, e, q, Q, P, P_{pub}, \mu, H, H_1, F_1, F_2, \ell_1, \ell_2\}$

KeyGen(1^k):

KGC computes C's private key $S_{ID_\mathsf{C}} = (H(ID_\mathsf{C}) + s)^{-1}Q$, where $ID_\mathsf{C} \in \{0,1\}^*$ and $P_{ID_\mathsf{C}} = (H(ID_\mathsf{C}) + s)P$
C now generates a random signing key pair (ssk, spk)
C forms her secret key $sk := (S_{ID_\mathsf{C}}, ssk)$ and $pk := (P_{ID_\mathsf{C}}, spk)$ as his public parameters.

SigGen(sk, F):

For file naming, C
- chooses a random element $name \in_R \mathbb{Z}_q$ for the file $F = (m_1, \cdots, m_n)$
- computes the file tag as $t = name||Sig_{ssk}(name)$ with a signature on the element $name$.

By picking random element $r_1^i \in_R \mathbb{Z}_q^*$, for each block $m_i \in \{0,1\}^{\ell_1}$, C generates a signature σ_i as follows:
- for simplicity denote $w_i := H(i||name)$,
- computes $\mu^{r_1^i}$ and $R_i := r_1^i Q$
- computes $\alpha_i = H_1(ID_\mathsf{C}, \mu^{r_1^i}) \in \{0,1\}^{\ell_1+\ell_2}$,
- computes $\beta_i = F_1(m_i) \parallel (F_2(F_1(m_i)) \oplus m_i)$ and $r_2^i = [\alpha_i \oplus \beta_i]_{10}$,
- computes $U_i = (r_1^i + w_i + r_2^i)S_{ID_\mathsf{C}}$.

The signature σ_i on m_i is (r_2^i, U_i). C then sends $\{\phi = \{\sigma_i\}_{1\le i\le n}, \psi = \{R_i\}_{1\le i\le n}, t\}$ to S.

Signature Verification and Message Recovery:

Upon receiving the signatures from C, S:
- computes $\tilde{\alpha}_i = H_1(ID_\mathsf{C}, e(U_i, P_{ID_\mathsf{C}})\mu^{-(r_2^i+w_i)})$ and $\tilde{\beta}_i = [r_2^i]_2 \oplus \tilde{\alpha}_i$,
- recovers the message $\tilde{m}_i = |\tilde{\beta}_i|_{\ell_1} \oplus F_2({}_{\ell_2}|\tilde{\beta}_i|)$,
- outputs 1 and accepts σ_i as a valid signature of the message $\tilde{m}_i (= m_i)$ if and only if ${}_{\ell_2}|\tilde{\beta}_i| = F_1(\tilde{m}_i)$.
- extracts each m_i for $1 \le i \le n$ and stores $(F = \{m_i\}_{1\le i\le n}, \phi = \{\sigma_i\}_{1\le i\le n}, \psi = \{R_i\}_{1\le i\le n}, t)$.

Proof of the Signature and Message Recovery:

$$\begin{aligned} e(U_i, P_{ID_\mathsf{C}}) \cdot \mu^{-(r_2^i+w_i)} &= e((r_1^i + w_i + r_2^i) \cdot S_{ID_\mathsf{C}}, P_{ID_\mathsf{C}}) \cdot e(Q,P)^{-(r_2^i+w_i)} \\ &= e(S_{ID_\mathsf{C}}, P_{ID_\mathsf{C}})^{(r_1^i+w_i+r_2^i)} \cdot e(Q,P)^{-(r_2^i+w_i)} \\ &= e((H(ID_i)+s)^{-1}Q, (H(ID_i)+s)P)^{(r_1^i+w_i+r_2^i)} \cdot e(Q,P)^{-(r_2^i+w_i)} \\ &= e(Q,P)^{(r_1^i+w_i+r_2^i)} \cdot e(Q,P)^{-(r_2^i+w_i)} \\ &= e(Q,P)^{r_1^i} \\ &= \mu^{r_1^i}. \end{aligned}$$

Message Recovery:

If σ_i is valid, then $H_1(ID_\mathsf{C}, \mu^{r_i}) = \alpha_i$ and $F_1(m_i) \parallel (F_2(F_1(m_i)) \oplus m_i) = \beta_i = [r_2^i]_2 \oplus \alpha_i$. Hence,

$$|\beta_i|_{\ell_1} \oplus F_2({}_{\ell_2}|\beta_i|) \ = (F_2(F_1(m_i)) \oplus m_i) \oplus F_2(F_1(m_i)) = m_i.$$

Fig. 2. Key Generation and Message Recoverable Signature for Cloud

- **Public verifiability:** It allows TPA to verify the correctness of cloud data without retrieving the entire data or without having online connections with the cloud users.
- **Storage correctness:** It ensures that a server can pass TPA's verification only if it indeed keeps user's data.
- **Privacy-preserving:** It assures that no information about data is leaked to TPA during the auditing process.

4 Our Proposed Public Auditing Scheme

A privacy-preserving public verifiable auditing scheme consists of basically four algorithms *KeyGen*, *SigGen*, *GenProof* and *VerifyProof*. *KeyGen* and *SigGen* are performed by the Client C to generate public/private keys, signatures and related information.

We assume that C partitions the file $\mathcal{F}$ into n blocks $m_1, m_2, \ldots, m_n$, where each block $m_i \in \{0,1\}^{\ell_1}$ for processing the data. The proposed scheme is illustrated in Figure 2.

Whenever TPA starts the auditing protocol, the tag t for the file F is retrieved and validated by using *spk*, and the process is ended if the test fails. Next, TPA randomly chooses $x_1, x_2 \in_R \mathbb{Z}_q^*$, constructs a challenge $chal$=$\{\{s_j, v_{s_j}\}_{1\leq j\leq c}, P_1 = x_1 \cdot P, H(x_1||x_2)\}$, where $\{s_j\}_{1\leq j\leq c}$ is a random subset with $S_c := \{s_1, \cdots, s_c\} \subseteq \{1, \cdots, n\}$ and $\{v_{s_j}\}_{1\leq j\leq c}$ are random mask values for $s_j \in S_c$. TPA subsequently sends *chal* to S.

GenProof($\boldsymbol{F, \phi, \psi, chal}$) After receiving the challenge $chal = \{\{s_j, v_{s_j}\}_{1\leq j\leq c}, P_1 = x_1 \cdot P, H(x_1||x_2)\}$, S picks firstly a random mask $\lambda \in_R \mathbb{Z}_q$, then computes

$$\Phi := \lambda \cdot \sum_{j=1}^{c} v_{s_j} \cdot U_{s_j} \text{ and } \Psi := \lambda \cdot \sum_{j=1}^{c} v_{s_j} \cdot (R_{s_j} + r_2^{s_j} \cdot Q),$$

and finally sends $(\Phi, e(\Psi, P_1), e(Q, \lambda \cdot P))$ to TPA.

VerifyProof($\boldsymbol{chal, \Phi, e(\Psi, P_1), e(Q, \lambda \cdot P)}$) TPA checks

$$e(\Phi, x_1 x_2 P_{ID_{\mathsf{C}}}) \stackrel{?}{=} e(\Psi, P_1)^{x_2} \cdot e(Q, \lambda P)^{x_1 x_2 \sum_{j=1}^{c} v_{s_j} w_{s_j}} \quad .$$

5 Security Analysis

5.1 Termination and Correctness

Theorem 1. *The algorithm of the above described public verifiable auditing scheme is correct and it terminates.*

Proof. First of all, note that

$$\begin{aligned} e(S_{ID_\mathsf{C}}, P_{ID_\mathsf{C}}) &= e((H(ID_\mathsf{C}) + s)^{-1} \cdot Q, (H(ID_\mathsf{C}) + s) \cdot P) \\ &= e(Q, P)^{(H(ID_\mathsf{C})+s)^{-1} \cdot (H(ID_\mathsf{C})+s)} \\ &= \mu \ . \end{aligned}$$

Then, the result follows by using bilinear property of e:

$$\begin{aligned} e(\Phi, x_1 x_2 P_{ID_\mathsf{C}}) &= e(\Phi, P_{ID_\mathsf{C}})^{x_1 x_2} \\ &= e\left(\lambda \sum_{j=1}^{c} v_{s_j} (r_1^{s_j} + w_{s_j} + r_2^{s_j}) \cdot S_{ID_\mathsf{C}}, P_{ID_\mathsf{C}}\right)^{x_1 x_2} \\ &= e(S_{ID_\mathsf{C}}, P_{ID_\mathsf{C}})^{\lambda x_1 x_2 \sum_{j=1}^{c} v_{s_j} \left(r_1^{s_j} + w_{s_j} + r_2^{s_j}\right)} \\ &= e(Q, P)^{\lambda x_1 x_2 \sum_{j=1}^{c} v_{s_j} \left(r_1^{s_j} + w_{s_j} + r_2^{s_j}\right)} \\ &= e(Q, P)^{\lambda x_1 x_2 \sum_{j=1}^{c} v_{s_j} \left(r_1^{s_j} + r_2^{s_j}\right) + \lambda x_1 x_2 \sum_{j=1}^{c} v_{s_j} w_{s_j}} \\ &= e(Q, P)^{\lambda x_1 x_2 \sum_{j=1}^{c} v_{s_j} \left(r_1^{s_j} + r_2^{s_j}\right)} \cdot \mu^{\lambda x_1 x_2 \sum_{j=1}^{c} v_{s_j} w_{s_j}} \\ &= e\left(\lambda \sum_{j=1}^{c} v_{s_j} \left(r_1^{s_j} \cdot Q + r_2^{s_j} \cdot Q\right), x_1 \cdot P\right)^{x_2} \cdot \\ &\qquad e\left(Q, \lambda \cdot P\right)^{x_1 x_2 \sum_{j=1}^{c} v_{s_j} w_{s_j}} \\ &= e(\Psi, P_1)^{x_2} \cdot e(Q, \lambda \cdot P)^{x_1 x_2 \sum_{j=1}^{c} v_{s_j} w_{s_j}} \ . \end{aligned}$$

□

5.2 Security Against Cloud Provider

Theorem 2. TPA *passes the verification of the auditing successfully only if* S *possesses truly the specified data.*

Proof. In this case, the cloud server is treated as an adversary and the TPA is treated as a challenger controlling the random oracle. If there is a non-negligible probability in the adversary's success, we can construct a simulator that can solve the computational Diffie-Hellman problem.

Let $(\Phi, e(\Psi, P_1), e(Q, \lambda \cdot P))$ be the output of an honest S. Then, it satisfies

$$e(\Phi, x_1 x_2 \cdot P_{ID_\mathsf{C}}) = e(\Psi, P_1)^{x_2} \cdot e(Q, \lambda \cdot P)^{x_1 x_2 \sum_{j=1}^{c} v_{s_j} w_{s_j}} .$$

Given for the same x_1, x_2 and λ, let $(\Phi', e(\Psi', P_1), e(Q, \lambda \cdot P))$ be the adversary's response satisfying

$$e(\Phi', x_1 x_2 \cdot P_{ID_\mathsf{C}}) = e(\Psi', P_1)^{x_2} e(Q, \lambda \cdot P)^{x_1 x_2 \sum_{j=1}^{c} v_{s_j} w_{s_j}} .$$

Dividing both equations, we obtain the equality

$$e(\Phi - \Phi', x_1 x_2 \cdot P_{ID_{\mathsf{C}}}) = e(\Psi - \Psi', P_1)^{x_2}$$

More concretely, we have

$$\begin{aligned} & e\left(\lambda \sum_{j=1}^{c} v_{s_j} \left(r_1^{s_j} + r_2^{s_j} - {r'}_1^{s_j} - {r'}_2^{s_j}\right) \cdot Q, x_1 \cdot P\right)^{x_2} \\ & = e\left(\lambda \sum_{j=1}^{c} v_{s_j} \left(r_2^{s_j} - {r'}_2^{s_j}\right) \cdot Q, x_1 \cdot P\right)^{x_2} \end{aligned}$$

By letting $\Delta_{r_1^{s_j}} := r_1^{s_j} - {r'}_1^{s_j}$ and $\Delta_{r_2^{s_j}} := r_2^{s_j} - {r'}_2^{s_j}$, we get

$$\begin{aligned} & e\left(\lambda \sum_{j=1}^{c} v_{s_j} \left(\Delta_{r_1^{s_j}} + \Delta_{r_2^{s_j}}\right) \cdot Q, x_1 \cdot P\right)^{x_2} \\ & = e\left(\lambda \sum_{j=1}^{c} v_{s_j} \left(\Delta_{r_2^{s_j}}\right) \cdot Q, x_1 \cdot P\right)^{x_2} \quad , \end{aligned}$$

and dividing right hand side to the left hand side, we obtain the following:

$$e\left(\lambda \sum_{j=1}^{c} v_{s_j} \left(\Delta_{r_1^{s_j}}\right) \cdot Q, x_1 x_2 \cdot P\right) = 1 \quad .$$

In order to obtain this equality we must have $\lambda \sum_{j=1}^{c} v_{s_j} \Delta_{r_1^{s_j}} \equiv 0 \bmod q$. This only holds if

$$\Delta_{r_1^{s_j}} \equiv 0 \bmod q \quad .$$

The probability of this event is $1/q$ which is negligible, therefore $r_1^{s_j} = {r'}_1^{s_j}$ for all s_j. If the adversaries success probability in this case is non-negligible, we can construct a simulator that can solve the discrete logarithm problem as follows:

$$\begin{aligned} U_{s_j} - U'_{s_j} = & (r_1^{s_j} + w_{s_j} + r_2^{s_j}) \cdot S_{ID_{\mathsf{C}}} \\ & - ({r'}_1^{s_j} + w_{s_j} + {r'}_2^{s_j}) \cdot S_{ID_{\mathsf{C}}} \\ = & (r_2^{s_j} - {r'}_2^{s_j}) \cdot S_{ID_{\mathsf{C}}} \quad . \end{aligned}$$

Hence, the simulator can compute

$$S_{ID_{\mathsf{C}}} = (r_2^{s_j} - {r'}_2^{s_j})^{-1} \cdot (U_{s_j} - U'_{s_j}).$$

$\square$

Table 1. Complexity of the Proposed Protocol

	FMult	FExp	ECSMult	BComp	Bandwidth between TPA and S
TPA	$c+3$	2	2	1	$\log n + 3(\ell_1 + \ell_2)$
S	c	1	$3c+2$	1	$3(\ell_1 + \ell_2)$

Table 2. Comparison with Previous Results (Considering only TPA)

	FMult	FExp	ECSMult	BComp	Message Recoverable
Wang *et al.* [32]	1	0	$c+3$	2	✗
Worku *et al.* [37]	0	0	$c+1$	2	✗
Ours	$c+3$	2	2	1	✓

Theorem 3. *A malicious* S *cannot perform replay and replace attacks. In particular,* S *cannot generate proofs without querying or computing the actual data or cannot modify the data and its signatures.*

Proof. The only reason for integrating $P_1 = x_1 \cdot P$ and $H(x_1||x_2)$ into the equation by TPA is to prevent replay attack of S. For instance, when $x_2 = 1$, S can manipulate the exponents simply by using the bilinear properties of the pairing function e. More concretely, if x_1 and x_2 were excluded in our scheme, S could easily manipulate the Φ, Ψ values accordingly and could pass successfully since S knows what TPA will compute at the verification phase. That is, one could modify the data by adding any random element to the exponents on both sides of the equality which passes the validation step of **TPA** since it would not affect the equality. Hence, randomizing the exponent with the value x_2 enables to prevent replay attacks. □

5.3 Security Against TPA

Theorem 4. *An honest-but-curious* TPA *cannot obtain any information about the message blocks* $F = \{m_1, \ldots, m_t\}$.

Proof. TPA sends a challenge set and obtains a valid response of the proof $(\Phi, e(\Psi, P_1), e(P, \lambda \cdot P))$. The challenges are completely random and are independent of the message blocks. Moreover, each signature block is multiplied with a random element λ which randomizes Φ, Ψ and P. □

6 Complexity Analysis

Overall complexity of a typical auditing scheme is typically analyzed by means of computation, communication and round complexity. All existing protocols have constant rounds therefore will be omitted. Note that the complexity between the client C and the server S is pretty standard, i.e., authentication and generation/verification of signatures are used. Both BLS like short signatures and message recoverable signatures are used to minimize the communication cost between C and S. Although our modified message recoverable signature scheme for auditing purposes seems to add extra complexity overhead to S, our Type 3 version of Tso *et al.*'s [29] protocol tolerates the additional complexity. In this way, message recoverable signatures with Type 3 protocols considerably hinder possible disadvantages of communication overhead due to more efficient choice of underlying group structures. Since auditing is the main concern of this work and due to space constraints, we omit the further details about complexity between C and S.

In Table 1, we demonstrate both computation and communication overhead of our auditing protocol (for both S and TPA) by counting basic group operations including field multiplication (FMult), field exponentiation (FExp), elliptic curve scalar multiplication ($\mathsf{ECSMult}$), and bilinear computation (BComp).

In Table 2, we compare our auditing protocol with the recently proposed auditing schemes using BLS like structures of Wang *et al.* and Worku *et al.* [32, 37]. The number of operations has been calculated for only TPA because in real-life scenarios TPA is assumed to be a stateless machine and has rather low computational power with respect to S. Therefore, it is essential to reduce the computational overhead for TPA. Table 2 shows that the computational complexity of TPA in our scheme is significantly better. More concretely, we only need 2 ECMults and only 1 BComps whereas others need elliptic curve scalar multiplications increasing linearly in c and 2 bilinear pairings.

6.1 Further Discussion: Reducing Number of Group Elements

For the communication overhead between C and S one can observe the following. In our scheme, the user sends the group elements

$$\{\phi = \{\sigma_i\}_{1\leq i\leq n}, \psi = \{R_i\}_{1\leq i\leq n}, t\}$$

to S. Since we have groups of order q, $3n(\ell_1 + \ell_2)$ bits are required to be transmitted for a single run of the message recoverable signature scheme. This can be reduced to transmission of $2n(\ell_1 + \ell_2)$ bits of information, hence gaining a linear factor on the block size of a message. The reason comes from the following simple observation:

Instead of working with a group $\mathbf{G}_1$ of prime order q, at the beginning one can simply choose a group $\mathbf{G}_1$ having order $N = p_1p_2$, where p_1 and p_2 are different prime numbers with bit lengths ℓ_1^*, ℓ_1^{**} respectively such that $\ell_1 = \ell_1^* + \ell_1^{**}$. Furthermore, by Chinese Remainder Theorem we have the property that

$$\mathbf{G}_1 \cong \mathbf{H}_1 \times \mathbf{H}_2,$$

where $\mathbf{H}_1$ and $\mathbf{H}_2$ are groups of order p_1 and p_2, respectively. We can easily identify the isomorphic subgroups of $\mathbf{G}_1$ also with $\mathbf{H}_i$, $i = 1, 2$. Instead of sending the message blocks m_i, the user sends the blocks $\tilde{m_i} = m_i||R_i$ by restricting the elements $m_i \in \mathbf{H}_1$ and $R_i \in \mathbf{H}_2$, respectively. We note that we need to have the property that the DLP has to be intractable in both $\mathbf{H}_1$ and $\mathbf{H}_2$, since otherwise Pohling-Hellman reduction technique solves DLP also in $\mathbf{G}_1$ [13].

In order to construct a group $\mathbf{G}_1$ with composite order, we need to generate pairing friendly abelian varieties using complex multiplication techniques [16]. Due to our efficiency concerns, we need to construct elliptic curves having composite orders and reasonably small embedding degree k such as $k = 1$ or $k = 2$ [16]. Hence, this idea would be impractical due to underlying key sizes. We note that it would be interesting to find an efficient way of reducing the communication complexity to $2n(\ell_1 + \ell_2)$ by using Type 3 bilinear maps with prime or nearly prime order.

7 Conclusion

In this study, we proposed the first privacy-preserving public auditing scheme using ID-based message recoverable signatures. In all existing schemes, the server has to protect the messages together with their corresponding signatures. Our scheme is robust, in the sense that the messages will be still recoverable unless the signatures are damaged. We prove the security of our scheme against forgery, replay and replace attacks in the random oracle model. We give the efficiency and the complexity comparisons of our scheme with the previously proposed auditing schemes and show that our scheme is significantly more efficient than the most efficient auditing schemes based on BLS like short signatures. In particular, the complexity of the stateless third party auditor has been considerably improved. Unlike previous schemes, we chose a variant of Type 3 version of message recoverable signature scheme to achieve a desired security level with small key sizes and to optimize the efficiency.

Acknowledgments. Kiraz's research is supported by a grant from Ministry of Development of Turkey provided to the Cloud Computing and Big Data Research Lab Project. Uzunkol's research is supported by the project (114C027) funded by EU FP7-The Marie Curie Action and TÜBİTAK (2236-CO-FUNDED Brain Circulation Scheme). Uzunkol's work is also partly supported by a joint research project funded by Bundesministerium für Bildung und Forschung (BMBF), Germany (01DL12038) and TÜBİTAK, Turkey (TBAG-112T011).

References

1. Adj, G., Menezes, A., Oliveira, T., Rodríguez-Henríquez, F.: Weakness of $\mathbb{F}_{3^{6\cdot 509}}$ for discrete logarithm cryptography. In: Pairing-Based Cryptography - Pairing 2013 - 6th International Conference, Beijing, China, November 22-24, 2013, Revised Selected Papers. pp. 20–44 (2013), http://dx.doi.org/10.1007/978-3-319-04873-4_2

2. Alliance, C.S.: Security guidance for critical areas of focus in cloud computing (2009), http://www.cloudsecurityalliance.org
3. Amazon.com: Amazon s3 availability event: July 20, 2008 (2008), http://status.aws.amazon.com/s3-20080720.html
4. Arrington, M.: Gmail disaster: Reports of mass email deletions (2006), http://www.techcrunch.com/2006/12/28/gmail-disasterreports-of-mass-email-deletionsTech Crunch Web Article
5. Ateniese, G., Burns, R., Curtmola, R., Herring, J., Kissner, L., Peterson, Z., Song, D.: Provable data possession at untrusted stores. Cryptology ePrint Archive, Report 2007/202 (2007), http://eprint.iacr.org/2007/202
6. Ateniese, G., Pietro, R.D., Mancini, L.V., Tsudik, G.: Scalable and efficient provable data possession. Cryptology ePrint Archive, Report 2008/114 (2008), http://eprint.iacr.org/2008/114
7. Barbulescu, R., Gaudry, P., Joux, A., Thomé, E.: A quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. CoRR abs/1306.4244 (2013)
8. Blake, I., Seroussi, G., Smart, N., Cassels, J.W.S.: Advances in Elliptic Curve Cryptography (London Mathematical Society Lecture Note Series). Cambridge University Press, New York, NY, USA (2005)
9. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the weil pairing. In: Boyd, C. (ed.) Advances in Cryptology, ASIACRYPT 2001, Lecture Notes in Computer Science, vol. 2248, pp. 514–532. Springer Berlin Heidelberg (2001), http://dx.doi.org/10.1007/3-540-45682-1_30
10. Chatterjee, S., Menezes, A.: On cryptographic protocols employing asymmetric pairings - the role of ψ revisited. Discrete Applied Mathematics 159(13), 1311–1322 (2011), http://dx.doi.org/10.1016/j.dam.2011.04.021
11. Chen, L.: Using algebraic signatures to check data possession in cloud storage. Future Generation Computer Systems 29(7), 1709 – 1715 (2013)
12. Chen, Y., Chou, J.S.: Crypto-analyses on "secure and efficient privacy-preserving public auditing scheme for cloud storage". Cryptology ePrint Archive, Report 2014/723 (2014), http://eprint.iacr.org/2014/723
13. Cohen, H., Frey, G., Avanzi, R., Doche, C., Lange, T., Nguyen, K., Vercauteren, F.: Handbook of elliptic and hyperelliptic curve cryptography. Chapman & Hall, Boca Raton, FL, 1st edn. (2006)
14. Ferdowsi, A.: Dropbox blog. yesterday's authentication bug (2011 (Accessed 15 April 2015)), http://blog.dropbox.com/index.php/yesterdays-authentication-bug
15. Filho, D.L.G., Barreto, P.S.L.M.: Demonstrating data possession and uncheatable data transfer. Cryptology ePrint Archive, Report 2006/150 (2006), http://eprint.iacr.org/2006/150
16. Freeman, D., Scott, M., Teske, E.: A taxonomy of pairing-friendly elliptic curves. Journal of Cryptology 23(2), 224–280 (2010)
17. Galbraith, S.D., Paterson, K.G., Smart, N.P.: Pairings for cryptographers. Discrete Appl. Math. 156(16), 3113–3121 (Sep 2008), http://dx.doi.org/10.1016/j.dam.2007.12.010
18. Granger, R., Kleinjung, T., Zumbrägel, J.: Breaking '128-bit secure' supersingular binary curves - (or how to solve discrete logarithms in $\mathbb{F}_{2^{4 \cdot 1223}}$ and $\mathbb{F}_{2^{12 \cdot 367}}$). In: Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part II. pp. 126–145 (2014), http://dx.doi.org/10.1007/978-3-662-44381-1_8

19. Group, T.T.W.: The notorious nine: Cloud computing top threats in 2013 (2013 (Accessed 15 April 2015)), https://downloads.cloudsecurityalliance.org
20. Hao, Z., Zhong, S., Yu, N.: A privacy-preserving remote data integrity checking protocol with data dynamics and public verifiability. Knowledge and Data Engineering, IEEE Transactions on 23(9), 1432–1437 (Sept 2011)
21. Juels, A., Kaliski, Jr., B.S.: Pors: Proofs of retrievability for large files. In: Proceedings of the 14th ACM Conference on Computer and Communications Security. pp. 584–597. CCS '07, ACM, New York, NY, USA (2007), http://doi.acm.org/10.1145/1315245.1315317
22. Kincaid, J.: Mediamax/the linkup closes its doors (2008), http://www.techcrunch.com/2008/07/10/mediamaxthelinkup-closes-its-doors Tech Crunch Web Article
23. Liu, B., Chen, Y.: Auditing for data integrity and reliability in cloud storage. In: Khan, S.U., Zomaya, A.Y. (eds.) Handbook on Data Centers, pp. 535–559. Springer New York (2015), http://dx.doi.org/10.1007/978-1-4939-2092-1_17
24. Liu, C., Ranjan, R., Zhang, X., Yang, C., Chen, J.: A big picture of integrity verification of big data in cloud computing. In: Khan, S.U., Zomaya, A.Y. (eds.) Handbook on Data Centers, pp. 631–645. Springer New York (2015), http://dx.doi.org/10.1007/978-1-4939-2092-1_21
25. Mandagere, N., Zhou, P., Smith, M.A., Uttamchandani, S.: Demystifying data deduplication. In: Proceedings of the ACM/IFIP/USENIX Middleware '08 Conference Companion. pp. 12–17. Companion '08, ACM, New York, NY, USA (2008), http://doi.acm.org/10.1145/1462735.1462739
26. Shacham, H., Waters, B.: Compact proofs of retrievability. In: Pieprzyk, J. (ed.) Advances in Cryptology - ASIACRYPT 2008, Lecture Notes in Computer Science, vol. 5350, pp. 90–107. Springer Berlin Heidelberg (2008), http://dx.doi.org/10.1007/978-3-540-89255-7_7
27. Shah, M.A., Swaminathan, R., Baker, M.: Privacy-preserving audit and extraction of digital contents. Cryptology ePrint Archive, Report 2008/186 (2008), http://eprint.iacr.org/
28. Sookhak, M., Talebian, H., Ahmed, E., Gani, A., Khan, M.K.: A review on remote data auditing in single cloud server: Taxonomy and open issues. Journal of Network and Computer Applications 43(0), 121 – 141 (2014)
29. Tso, R., Gu, C., Okamoto, T., Okamoto, E.: Efficient id-based digital signatures with message recovery. In: Cryptology and Network Security, Lecture Notes in Computer Science, vol. 4856, pp. 47–59. Springer Berlin Heidelberg (2007)
30. Wang, B., Li, B., Li, H.: Oruta: privacy-preserving public auditing for shared data in the cloud. Cloud Computing, IEEE Transactions on 2(1), 43–56 (Jan 2014)
31. Wang, B., Li, B., Li, H., Li, F.: Certificateless public auditing for data integrity in the cloud. In: Communications and Network Security (CNS), 2013 IEEE Conference on. pp. 136–144 (Oct 2013)
32. Wang, C., Chow, S., Wang, Q., Ren, K., Lou, W.: Privacy-preserving public auditing for secure cloud storage. Computers, IEEE Transactions on 62(2), 362–375 (Feb 2013)
33. Wang, C., Ren, K., Lou, W., Li, J.: Toward publicly auditable secure cloud data storage services. Network, IEEE 24(4), 19–24 (July 2010)
34. Wang, C., Wang, Q., Ren, K., Lou, W.: Privacy-preserving public auditing for data storage security in cloud computing. In: INFOCOM, 2010 Proceedings IEEE. pp. 1–9 (March 2010)

35. Wang, Q., Wang, C., Li, J., Ren, K., Lou, W.: Enabling public verifiability and data dynamics for storage security. Cryptology ePrint Archive, Report 2009/281 (2009), http://eprint.iacr.org/2009/281
36. Wang, Q., Wang, C., Ren, K., Lou, W., Li, J.: Enabling public auditability and data dynamics for storage security in cloud computing. Parallel and Distributed Systems, IEEE Transactions on 22(5), 847–859 (May 2011)
37. Worku, S.G., Xu, C., Zhao, J., He, X.: Secure and efficient privacy-preserving public auditing scheme for cloud storage. Computers and Electrical Engineering 40(5), 1703 – 1713 (2014)
38. Xu, C., Zhang, Y., Yu, Y., Zhang, X., Wen, J.: An efficient provable secure public auditing scheme for cloud storage. KSII Transactions on Internet and Information Systems 8(11), 4226 – 4241 (2014)
39. Yang, K., Jia, X.: Data storage auditing service in cloud computing: challenges, methods and opportunities. World Wide Web 15(4), 409–428 (2012), http://dx.doi.org/10.1007/s11280-011-0138-0
40. Yu, Y., Au, M., Mu, Y., Tang, S., Ren, J., Susilo, W., Dong, L.: Enhanced privacy of a remote data integrity-checking protocol for secure cloud storage. International Journal of Information Security pp. 1–12 (2014), http://dx.doi.org/10.1007/s10207-014-0263-8
41. Yu, Y., Zhang, Y., Ni, J., Au, M.H., Chen, L., Liu, H.: Remote data possession checking with enhanced security for cloud storage. Future Generation Computer Systems (2014)