Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 614375, 8 pages
http://dx.doi.org/10.1155/2015/614375

Research Article

A Secure and Effective Anonymous Integrity Checking Protocol for Data Storage in Multicloud

Lingwei Song,[1] Dawei Zhao,[2] Xuebing Chen,[3] Chenlei Cao,[1] and Xinxin Niu[1]

[1]*Information Security Center, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*Shandong Provincial Key Laboratory of Computer Network, Shandong Computer Science Center (National Supercomputer Center in Jinan), Jinan 250014, China*
[3]*University of International Business and Economics, Beijing 100029, China*

Correspondence should be addressed to Lingwei Song; songlw@bupt.edu.cn

Received 17 September 2014; Accepted 24 December 2014

Academic Editor: Florin Pop

How to verify the integrity of outsourced data is an important problem in cloud storage. Most of previous work focuses on three aspects, which are providing data dynamics, public verifiability, and privacy against verifiers with the help of a third party auditor. In this paper, we propose an identity-based data storage and integrity verification protocol on untrusted cloud. And the proposed protocol can guarantee fair results without any third verifying auditor. The theoretical analysis and simulation results show that our protocols are secure and efficient.

1. Introduction

With the growing popularity of clouds, the tools and technologies for hybrid clouds have been emerging recently; cloud storage has become a hot research topic that aims to provide a comparably low cost, scalable, position-independent platform for data owners data [1]. However, this new paradigm of data hosting service also introduces new security challenges [2]. A list of security threats to cloud computing is presented in [3]. These issues range from the required trust in the cloud server for storage and attacks on cloud interfaces to misusing the cloud services for attacks in the complex systems. When considering using the complex cloud service, the data owner must be aware of the fact that all data given to the cloud server leave the owner control protection sphere [4]. Huge measurement data, huge environment monitoring data, hydrological data, marine biological data, and GIS information are provided by the complex multicloud. In this situation, it is a strong demand that the data owners can check the data integrity confidentially, dynamically, and publicly; besides, the anonymous is also demanded for smart phone users.

In the past few years, some work has been done on insuring remote data integrity checking, which allows data integrity to be checked without completely downloading the data. Prior studies were based on two-party storage checking protocols that the data owner can check the data integrity [4–12]. Deswarte et al. [5] and Filho and Barreto [9] introduced RSA-based methods for solving remote data integrity checking. After that Shah et al. [12] proposed a remote storage auditing method based on precomputed challenge-response pairs. In practical application, to guarantee fair results, neither the cloud service provider nor the data owner should be the auditor in a cloud storage system. In this case, the protocols [13–15] employed the third party audit (TPA) performing the verification. However, none of them provided privacy against third party verifiers under the condition of introducing TPA. Wang et al. [14, 16] recognized the need of privacy against third party verifiers and proposed a random masking technique to cope with this problem. Scheme [17–21] required an additional trusted organizer to send a commitment to the auditor to ensure data privacy during auditing. The auditing protocol may make a performance bottleneck for the auditor. On some cases, without requiring any trusted organizer during the batch auditing for multiple clouds the client may delegate the remote data integrity checking task to the third party. It results in the untrusted

third party auditing in cloud computing [22, 23]. Yang and Jia [22] introduced an index table (ITable) to record the abstract information of the data; they proposed that the cloud server could be dishonest and may launch attacks just like replay attack, forge attack, and replace attack but only used ITable with time stamps to solve the problems. Wang [23] introduced identity-based distributed provable data possession in multicloud storage to check the certificate when it checks the remote data integrity. Chen et al. [24] also propose a new secure outsourcing algorithm for (variable-exponent, variable-base) exponentiation modulo a prime in the two untrusted program models.

However, one of benefits of cloud storage is to enable universal data access with independent geographical locations. This implies that the end devices may be mobile and limited in computation and storage. Efficient integrity checking protocols are more suitable for cloud clients equipped with mobile end devices. Meanwhile, when a mobile user remotes into a foreign network, mutual authentication must first be solved to prevent illegal use from accessing services and to ensure that mobile users are connected to trusted networks [25]. Both Zhao and Liu used smart-card to resolve the authentication. To compensate for these shortcomings, our construction can be observed as an adaptation of the protocol of [20, 22, 23, 25, 26].

This paper aims to fill the gap on a secure and effective anonymous authentication protocol for remote verification protocol in multicloud storage based on complex system. To the best of our knowledge, our scheme is the first to provide the authentication and establishment of remote verification scheme when mobile user is located in his/her home network; therefore it is more practical and universal for complex multicloud storage system. The scheme does not use timestamp; thus it avoids the clock synchronization problem. Additionally, the performance and cost analysis also show that our scheme is more suitable for low-power and resource-limited mobile devices and thus availability for real implementation.

The rest of the paper is organized as follows. The layered security architecture and definitions are present in Section 2. In Section 3, a novel anonymous authentication protocol for remote verification user authentication scheme is proposed in multicloud storage. In Section 4, we analyze the security of our proposed scheme. Next, we analyze the functionality and performance of our proposed scheme and make comparisons with other related schemes in Section 5. Finally, Section 6 gives the concluding remark of the whole paper.

2. Definitions and Preliminiaries

In this section, we present our system model and briefly introduce the elliptic curve cryptosystem and some related mathematical assumptions.

2.1. Definitions of System Model. A representative network architecture for a secure and effective anonymous dynamics integrity checking protocol for data storage in multicloud (SA-DVCP) in global mobility networks is illustrated in

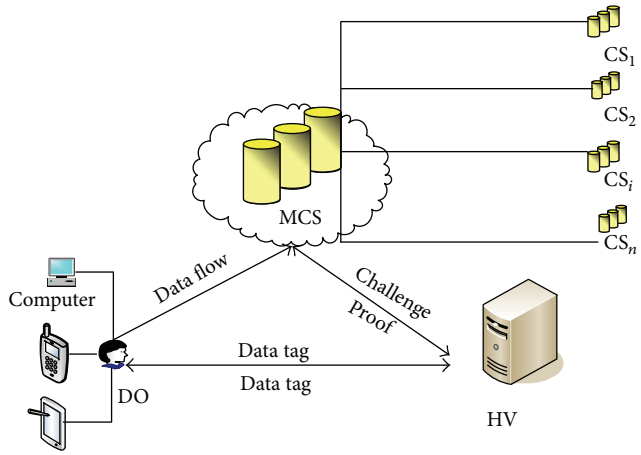

Figure 1: The system model of SA-DVCP.

Figure 1. Three different network entities can be identified as follows.

(1) The data owner, that has massive data to be stored on the multicloud for maintenance and computation, can be either individual consumer or corporation who has large amount of data files to be stored in the cloud. DO has the ability to check the storage integrity of their outsourced data, while hoping to keep their data private from any entity which is untrusted. The checking devices may be mobile and limited in computation and storage, which need a secure and effective anonymous integrity checking protocol.

(2) The data user/client/requested (DU), who accesses the CS or downloads the data from CS, has capabilities to check the integrity of data.

(3) Data stakeholder (DS): we define both DO and DU as data stakeholder.

(4) The multicloud server (MCS), which has significant storage space and computation resources to store the owners data and provides the data access to data users (data client/requesters), stores its whole data on the different cloud servers according to their importance and sensitivity.

(5) The HV (Home Verifier) is a home third party that has expertise and capabilities to provide data storage auditing service for both the DS and DU. The HV can provide unbiased result for both the DO and the CS.

2.2. Notation and Preliminaries. Let f be a pseudorandom function and let π be a pseudorandom permutation. They can be described in detail as follows:

$$f : Z_P^* \times \{1, 2, \ldots, n\} \rightarrow Z_P^*,$$

$$\pi : Z_P^* \times \{1, 2, \ldots, n\} \rightarrow \{1, 2, \ldots, n\},$$

in which k and d are two security parameters. Furthermore, denote the length of n in bits by $|P|$. We now introduce some necessary cryptographic background for our proposed scheme.

Bilinear Map. Let G_1 be a cyclic additive group generated by P and let G_2 be a cyclic multiplicative group generated by P with a bilinear map $\widehat{e} : G_1 \times G_2 \rightarrow G_T$.

(a) $\forall P \in G_1, \forall Q \in G_2, \widehat{e}(aP, bQ) = \widehat{e}(P, Q)^{ab}$.

(b) Nondegenerate: there exists $P, Q \in G_1$, such that $\widehat{e}(P, Q) \neq I_G$, where denotes the identity element of the group G_2.

(c) Computational discrete logarithm (CDL) problem: given $R = aP$, where $P, R \in G_1$. It is easy to calculate R given a and P, but it is hard to determine a given P and R.

(d) Computational co-Diffie-Hellman: given $P, aP \in G_1$, and $aQ \in G_1$, compute $aQ \in G_2$.

For providing high insecurity level of the proposed scheme, some important mathematical assumptions are introduced for bilinear pairings defined on elliptic curves.

(e) Define $A = aP$, $B = bP$, and $C = cP$; the computational bilinear Diffie-Hellman (CBDH) problem is computing the value $\mathrm{bdh}(A, B, C)$ given randomly. The CBDH assumption asserts that the CBDH problem is hard that is for all PPT algorithms A.

(f) Decision co-Diffie-Hellman: given $P, aP \in G_1$, and $Q, bQ \in G_2$, output is yes if $a = b$ and no otherwise. When the answer is yes we say that it is a co-Diffie-Hellman tuple.

3. The Proposed Schemes

In this section, we propose a novel anonymous dynamics integrity checking protocol for data storage in multicloud (SA-DVCP), using elliptic curve cryptosystem to not only protect the scheme from security breaches but also emphasize the efficient features. Before describing the auditing protocol definition, some notations are defined as in Notations and Descriptions section.

Suppose a file F has m data components as $M = (m_1, \ldots, m_n)$. Each data component has its physical meanings and can be updated dynamically by the data owners. For public data components, the data owner does not need to encrypt it, but for private data component, the DO needs to encrypt it with its corresponding key.

For simplicity, we only consider one data component in our construction and constant number of sectors for each data block. Suppose there is a data component M, which is divided into n data blocks, and each data block is further split into s sectors. For data blocks with different sector number. Then for $1 \leq i \leq n$, each block M_i is split into s sectors; that is, $M_i = \{M_{i1}, \ldots, M_{is}\}$. Our storage auditing protocol consists of the following algorithms.

Setup $(1^k) \rightarrow (pk, sk)$. Input the security parameter k and the bilinear map $e : G_P \times G_Q \rightarrow G_T$. Let G_T be multiplicative cyclic groups of prime order p, $G_P = \langle P \rangle$, $G_Q = \langle Q \rangle$, and $g = e(P, Q)$. Let $e(P, Q) \neq I_{G_T}$, $\mathrm{pk} = (P, Q, g)$, and $\mathrm{sk} = (\mathrm{PW}_{\mathrm{DS}}, \mathrm{ID}_{\mathrm{DS}})$; pk is the public key and sk is the private key. Let $h : \{0, 1\}^* \rightarrow G_P$ be a keyed secure hash function that maps the M_{ij} to a point in G_P.

TagGen $(pk, sk, MCS, M) \rightarrow D_i$. The tag generation algorithm takes as inputs each data component M and a set of $\mathrm{CSP} = \{\mathrm{CS}_j\}$, the private key sk. For each data block F_i, it computes a data tag D_i as $D_i = h(w_i, \mathrm{CS}_i) \cdot \prod_{j=1}^{s} M_{ij} \cdot P$.

Where $w_i = \mathrm{name} \parallel i$ and name is chosen by the DO uniformly at random from Z_p as the identifier of file M and i represents the block number of m_{ij}. It outputs the set of data tags $D_m = \{D_1, D_2, D_3, \ldots, D_n\}$. Without loss of generality, we suppose that every block has its uniqueness. After finishing computing all the block tags, the DO sends the file M to MCS and releases D_m to be publicly known to everyone.

Proof (P, C (MCS), V (Home Verifier)). SA-DVCP is a protocol among P, C, and V. At the end of the interactive protocol, HV outputs the auditing result as 0 or 1. If DS delegates the verification task to HV, it needs to register himself/herself to his/her HV.

(1) Registration. The details of DS registration phase are shown in Figure 2.

The interaction protocol can be given in detail as follows.

Step R1. DS freely chooses his/her identity $\mathrm{ID}_{\mathrm{DS}}$ and password $\mathrm{PW}_{\mathrm{DS}}$ and generates a random number N_{DS}. Then DS submits $\{\mathrm{ID}_{\mathrm{DS}}, h(\mathrm{PW}_{\mathrm{DS}} \parallel N_{\mathrm{DS}})\}$ to HV for registration via a secure channel.

Step R2. When receiving the message $\{\mathrm{ID}_{\mathrm{DS}}, h(\mathrm{PW}_{\mathrm{DS}} \parallel N_{\mathrm{DS}})\}$ HV computes $Q = h(\mathrm{ID}_{\mathrm{DS}} \parallel y) \oplus h(\mathrm{PW}_{\mathrm{DS}} \parallel N_{\mathrm{DS}})$ and $H = h(\mathrm{ID}_{\mathrm{DS}} \parallel h(\mathrm{PW}_{\mathrm{DS}} \parallel N_{\mathrm{DS}}))$, $C = cP$, where y is a secret number of HV, and picks the challenge $\mathrm{chal} = (c, k_1, k_2)$, $1 \leq c \leq n, k_1, k_2 \in Z_P^*$. Then HV submits $\{Q, H, \mathrm{ID}_{\mathrm{HV}}, C, \mathrm{chal}\}$ to DS through a secure channel.

(2) The Authentication and Proof. The details of the authentication and proof DS registration phase are shown in Figure 3. When roaming into a foreign network MCS, DS needs to verify the validity of MCS and proves to DS that he is a legitimate user. The authentication and proof phase used to solve the above issue in our proposed scheme is described as follows.

Step P1. DS generates a random number a and computes $A = aP$, $R_{AC} = acP$, $N = Q \oplus h(\mathrm{PW}_{\mathrm{DS}} \parallel N_{\mathrm{DS}})$, $\mathrm{DID}_{\mathrm{DS}} = \mathrm{ID} \oplus h(R_{AC})$, and $V_1 = h(N \parallel \mathrm{ID}_{\mathrm{DS}} \parallel A \parallel C \parallel AC)$ and

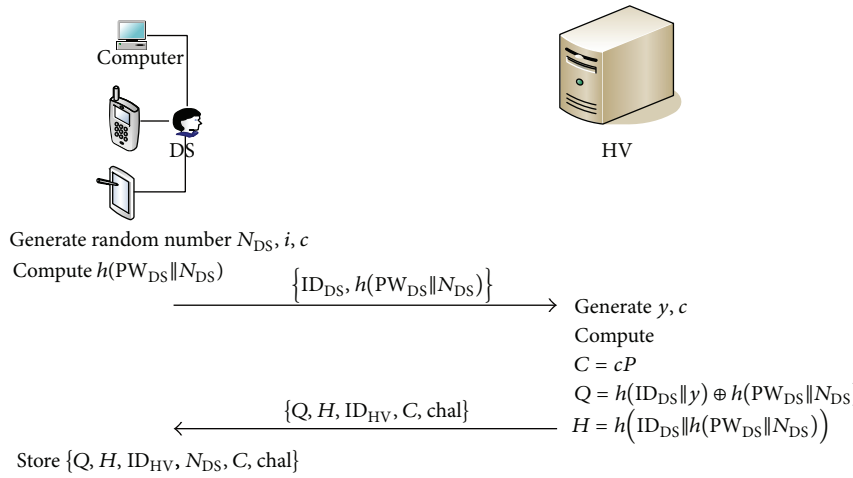

Figure 2: Registration phase of SA-DVCP.

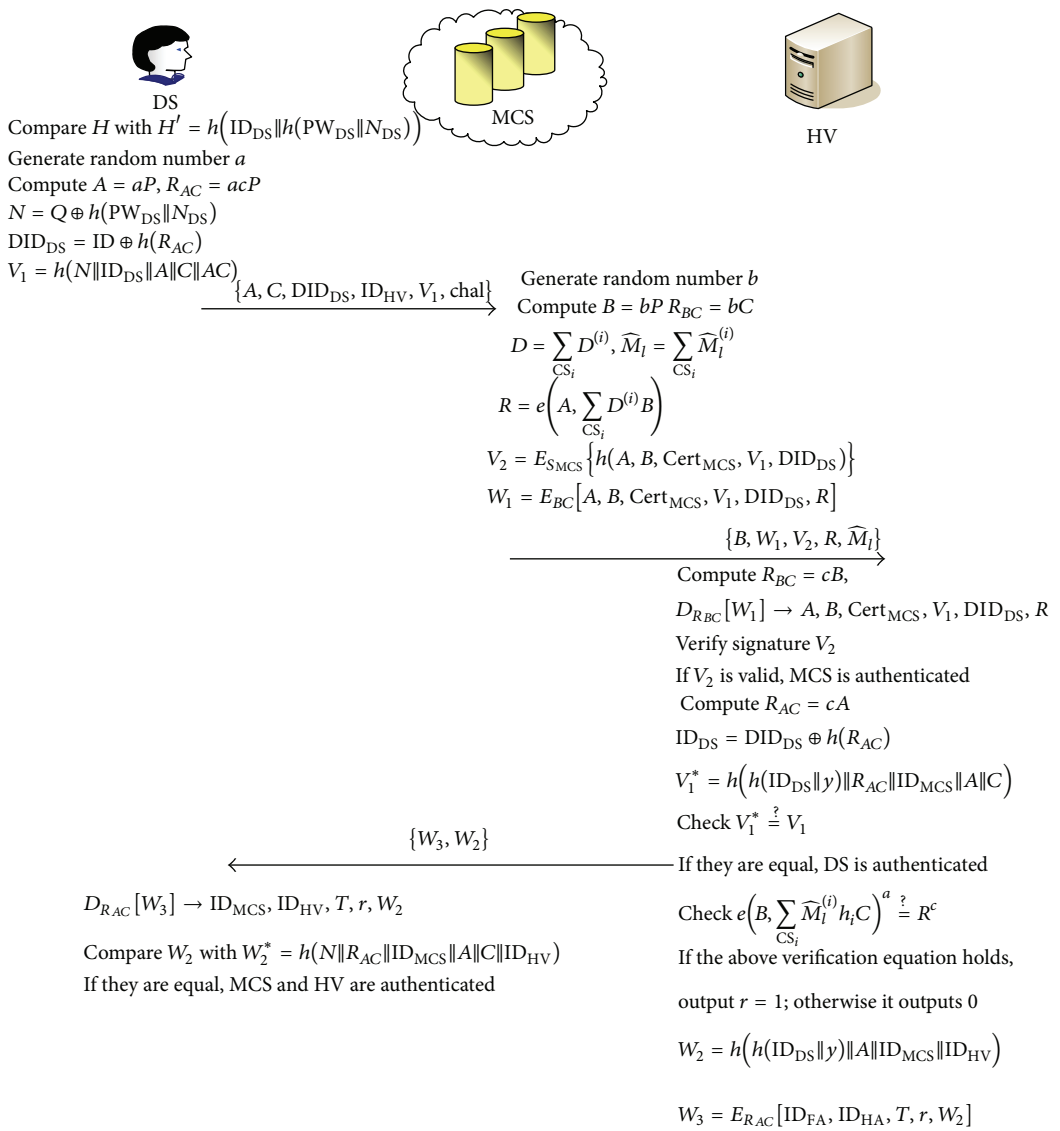

Figure 3: Proof phase of SA-DVCP.

DS sends the request message $\{A, C, \mathrm{DID}_{\mathrm{DS}}, \mathrm{ID}_{\mathrm{HA}}, V_1, \mathrm{chal}\}$ to MCS over a public channel.

Step P2. After receiving the message $\{A, C, \mathrm{DID}_{\mathrm{DS}}, \mathrm{ID}_{\mathrm{HA}}, V_1, \mathrm{chal}\}$, MCS generates a random number b and computes $B = bP$ and $R_{BC} = bC$ and $V_2 = E_{S_{\mathrm{MCS}}}\{h(A, B, \mathrm{Cert}_{\mathrm{MCS}}, V_1, \mathrm{DID}_{\mathrm{DS}})\}$ and $W_1 = E_{BC}[A, B, \mathrm{Cert}_{\mathrm{MCS}}, V_1, \mathrm{DID}_{\mathrm{DS}}, R]$. Here, S_{MCS} is the private key of MCS, and $\mathrm{Cert}_{\mathrm{MCS}}$ is MCSs certificate. Next, the MCS calculates $v_i = \pi_{k_1}(i)$, $1 \le i \le c$ and looks up the table T_o to get the records that correspond to $\{v_1, v_2, \ldots, v_c\} = F_1 \cup F_2 \cup \cdots \cup F_{\widehat{n}}$ denoting the index set where the corresponding block-tag pair is stored in CS_i. Then, C sends (F_i, k_2) to $\mathrm{CS}_i \in P$.

Response 1 ($P \leftarrow C$). For $\mathrm{CS}_i \in P$, it performs the following procedures.

(a) For $v_l \in M_i$, CS_i splits M_{vl} into s sectors $M_{vl} = \{\widetilde{M}_{vl1}, \widetilde{M}_{vl2}, \ldots, \widetilde{M}_{vls}\}$ and $1 \le j \le s$.

(b) CS_i calculates $a_l = f_{k_2}(l)$, $v_l \in M_i$, and $D^{(i)} = \sum_{v_l \in M_i} D_{vl} a_l$.

(c) For $1 \le j \le s$, CS_i calculates $\widehat{M}_j^{(i)} = \sum_{v_l \in F_i} a_l M_{v_l j}$ denoting $M_i = (\widehat{M}_1^{(i)}, \ldots, \widehat{M}_s^{(i)})$.

(d) CS_i sends $\rho_i = (\widehat{M}^{(i)}, D^{(i)})$ to C.

Response 2 ($C \rightarrow V$). After receiving all the responses from $\mathrm{CS}_i \in P$, the combiner aggregates $\{\rho_i\}_{\mathrm{CS}_i \in P}$ into the final response as $D = \sum_{\mathrm{CS}_i} D^{(i)}$, $\widehat{M}_l = \sum_{\mathrm{CS}_i} \widehat{M}_l^{(i)}$ denote $\widehat{M} = (\widehat{M}_1, \widehat{M}_2, \ldots, \widehat{M}_S)$. The MCS generates the data proof R as $R = e(A, Db)$. Then MCS sends $\{B, W_2, V_2, R, \widehat{M}_l\}$ to HV.

Step P3. When receiving $\{B, W_2, V_2, R\}$, HV first computes $R_{BC} = cB$ and decrypts $D_{R_{BC}}[W_1]$ to reveal $A, B, \mathrm{Cert}_{\mathrm{MCS}}, V_1$ and $\mathrm{DID}_{\mathrm{DS}}$. Then, HV verifies the MCSs signature V_2 by using the MCSs certificate $\mathrm{Cert}_{\mathrm{MCS}}$. If they are valid, MCS is authenticated. After that, HV computes the following:

$$R_{AC} = cA, \quad \mathrm{ID}_{\mathrm{DS}} = \mathrm{DID}_{\mathrm{DS}} \oplus h(R_{AC}),$$
$$V_1^* = h(h(\mathrm{ID}_{\mathrm{DS}} \parallel y) \parallel R_{AC} \parallel \mathrm{ID}_{\mathrm{MCS}} \parallel A \parallel C).$$

Then HV checks whether $V_1^* \stackrel{?}{=} V_1$. If they are equal, DS is authenticated by HV. Next, HV calculates

$$v_i = \pi_{k_1}(i),$$
$$h_i = h(N_{v_i}, \mathrm{CS}_{l_{v_i}}, v_i),$$
$$a_i = f_{k_2}(i).$$

Then, it verifies whether the following formula holds: $e(B, \sum_{\mathrm{CS}_i} \widehat{M}_l^{(i)} h_i C)^a = R^c$.

If the formula holds, then the verifier outputs $r = 1$. Otherwise, the verifier outputs $r = 0$. Next compute

$$W_2 = h(h(\mathrm{ID}_{\mathrm{DS}} \parallel y) \parallel A \parallel \mathrm{ID}_{\mathrm{MCS}} \parallel \mathrm{ID}_{\mathrm{HV}})$$

$W_3 = E_{R_{AC}}[\mathrm{ID}_{\mathrm{MCS}}, \mathrm{ID}_{\mathrm{HV}}, T, r, W_2]$; at last, HV sends $\{W_3, W_2\}$ to DS.

Step P4. DS decrypts $D_{R_{AC}}[W_3]$ to reveal $D_{\mathrm{MCS}}, \mathrm{ID}_{\mathrm{HA}}, T, r, W_2$. Then, the MU compare W_2 with $W_2^* = h(N \parallel R_{AC} \parallel \mathrm{ID}_{\mathrm{MCS}} \parallel A \parallel C \parallel \mathrm{ID}_{\mathrm{HV}})$. If it is valid, HV and MCS are all authenticated by DS.

4. Security Analysis of the Proposed Scheme

In this section, we show that the proposed scheme can withstand all possible security attacks.

4.1. Storage Correctness Guarantee

Theorem 1. *A SA-DVCP protocol must be workable and correct. That is, if the DS, MCS, and HV are honest and follow the specified procedures, the response $\{R, \widehat{M}_l\}$ can pass HV's checking. The correctness follows from*

$$\begin{aligned} e\left(B, \sum_{CS_i} \widehat{M}_l^{(i)} h_i C\right)^a &= e\left(bP, \sum_{CS_i} \widehat{M}_l^{(i)} h_i cP\right)^a \\ &= e\left(P, \sum_{CS_i} \sum_{vl \in M_i} h_i \sum_{j=1}^{s} a_l M_{vlj} cP\right)^{ab} \\ &= e\left(aP, \sum_{CS_i} D^{(i)} P\right)^{bc} \\ &= e\,(A, Db)^c = R^c. \end{aligned} \tag{1}$$

This completes the proof.

4.2. Privacy-Preserving Guarantee

Theorem 2. *The proposed protocol can provide users privacy-preserving.*

Proof. In our proposed scheme, the DS sends the login request message $\{A, C, \mathrm{DID}_{\mathrm{DS}}, \mathrm{ID}_{\mathrm{HA}}, V_1, \mathrm{chal}\}$ to MCS, where $\mathrm{DID}_{\mathrm{DS}} = \mathrm{ID} \oplus h(R_{AC})$ is used to protect the real identity $\mathrm{DID}_{\mathrm{DS}}$ of DS. Based on the CDL problem, any attacker cannot obtain the random number a from A and thus cannot retrieve $\mathrm{ID}_{\mathrm{DS}}$ from $\mathrm{DID}_{\mathrm{DS}}$. At the same time, the attacker cannot trace the moving history and current location of DS according to the login request message since A, $\mathrm{DID}_{\mathrm{DS}}$, and V_1 are dynamically changed in different login request messages of DS. Therefore, the proposed scheme can provide privacy-preserving of DS. □

4.3. Resist Impersonation Attack. Our proposed protocol can efficiently prevent impersonation attacks by considering the following scenarios.

Proof. Our proposed scheme can efficiently prevent impersonation attacks by considering the following scenarios.

(1) Any attacker cannot impersonate DS to cheat MCS and HV. In the proposed scheme, whether DS is located in a foreign network or in his/her home

Table 1: Comparison of cost.

	TagGen	Verify	$P + C$
Zhu et al. [17]	$(s + n(s + 2))C_{\text{exp}} + nsC_{\text{mul}}$	$3C_e + (c + s)C_{\text{exp}} + (c + s - 2)C_{\text{mul}}$	$nsC_e + (3n + c + 2)C_{\text{exp}} + (2\widehat{n} + c - 1)C_{\text{mul}}$
Zhu et al. [20]	$(s + 2n)C_{\text{exp}} + nC_{\text{mul}}$	$3C_e + (c + s)C_{\text{exp}} + (c + s - 2)C_{\text{mul}}$	$cC_{\text{exp}} + (c - 1)C_{\text{mul}}$
Wang [23]	$n(s + 1)C_{\text{exp}} + nsC_{\text{mul}} + nsC_{h_1}$	$2C_e + (c + s + 1)C_{\text{exp}} + (c + s)C_{\text{mul}}$	$cC_{\text{exp}} + (c - 1)C_{\text{mul}} + csC_{h_1}$
Our protocol	$nsC_{\text{mul}} + nsC_{h_1}$	$C_e + (c + s)C_{\text{mul}}$	$C_e + (c + 1)C_{\text{mul}} + 2C_{\text{exp}} + csC_{h_1}$

network, the HV authenticates DS by verifying the computed $V_1^* = h(h(\text{ID}_{\text{DS}} \parallel y) \parallel R_{AC} \parallel \text{ID}_{\text{MCS}} \parallel A \parallel C)$ with the received $V_1 = h(N \parallel \text{ID}_{\text{DS}} \parallel A \parallel C \parallel AC)$. Since the attacker does not possess DSs password PW_{DS}, he/she cannot compute the correct $N = Q \oplus h(\text{PW}_{\text{DS}} \parallel N_{\text{DS}})$ and thus cannot cheat HV by forging a login request message. At the same time, since a is a one-time random number and only possessed by DS, V_1 is dynamically changed in each login request message. Therefore, the attacker cannot cheat the HV by replaying a previous login request message. Besides, when DS is located in a foreign network, the authentication of MCS to DS is completely dependent on the authentication of HV to DS. If an attacker cannot successfully cheat HV by masquerading as DS, he/she cannot cheat MCS successfully.

(2) Any attacker cannot impersonate MCS to cheat HV and DS. In the proposed scheme, the HV authenticates MCS by checking whether $D_{P_{\text{MCS}}}\{V_2\}$ equals $h(A, B, \text{Cert}_{\text{MCS}}, V_1, \text{DID}_{\text{DS}})$, where V_2 is MCSs digital signature. Obviously, the attacker cannot compute the correct MCSs digital signature without knowing MCSs private key $\S_{\text{MCS}}$. Therefore, the attacker cannot cheat HV successfully by masquerading as MCS. At the same time, the authentication of DS to MCS is completely dependent on the authentication of HV to MCS. If an attacker cannot successfully cheat HV by masquerading as MCS, he/she cannot cheat DS successfully.

(3) Any attacker cannot impersonate HV to cheat DS. In the proposed scheme, the DS authenticates HV by verifying $W_2^* = h(N \parallel R_{AC} \parallel \text{ID}_{\text{MCS}} \parallel A \parallel C \parallel \text{ID}_{\text{HV}})$ with the received $\text{W}_2 = h(h(\text{ID}_{\text{DS}} \parallel y)R \parallel A \parallel \text{ID}_{\text{MCS}} \parallel \text{ID}_{\text{HV}})$. Obviously, any attacker cannot compute the correct W_2 without knowing ID_{DS} and y, and the attacker cannot cheat DS successfully.

□

4.4. Forward Secerecy

Theorem 3. *The proposed protocol meets the security requirement for perfect forward secrecy.*

Proof. Perfect forward secrecy means that even if an attacker compromises all the passwords of the entities of the system, he/she still cannot compromise the session key. In the proposed scheme, these three one-time random numbers a, b, and c are only held by the DS, MCS, and HV, respectively, and cannot be retrieved from $A = aP$, $B = bP$, $R_{AC} = aC = cA$, and $R_{BC} = bC = cB$ based on the security of CDL and CDH problem. Thus, even if an adversary obtains all the passwords of the entities, previous session keys, and all the transmitted messages, he/she still cannot compromise other session keys. Hence, the proposed scheme achieves perfect forward secrecy. □

5. Performance Comparison and Functionality Analysis

It is well known that most of the mobile devices have limited energy resources and computing capability. Hence, one of the most important issues in wireless networks is power consumption caused by communication and computation. In fact, the communication cost in the GLOMONET is higher than computation cost in terms of power consumption. In Table 1, we list the numbers of the TagGen, Verify and the $P + C$ phases of our scheme and some related previous schemes.

Computation. Suppose there are n message blocks which will be stored in n cloud servers. The blocks sector number is s and the challenged block number is c. We will consider the computation overhead in the different phases. On group G_P, bilinear pairings, exponentiation, multiplication, and the hash function h_1 contribute most computation cost. Compared with them, the hash function h and the operations on Z_P and G_Q are faster; the hash function H can be done once for all. On the DS, the computation cost mainly comes from the procedures of TagGen and verification (i.e., phase 5 in the protocol proof (P, C, V)). In the phase TagGen, the client performs ns multiplication on G_P, n, and hash function h_1. At the same time, for every file, the corresponding record ρ_i is stored by DS and CS. This stored metadata is small. In the phase proof, in order to respond the challenge $\text{chal} = (c, k_1, k_2)$ and generate the response ρ and the MCS perform $c + 1$ multiplication on the group G_P, cs hash function h_1. In the verification of the response, HV performs 2 exponentiations, 2 pairings, and $c + s$ multiplication on the group G_P and c hash function h. On the other hand, in 2012, Zhu et al. proposed the cooperative provable data possession for integrity in multicloud storage [17]. Almost at the same time, Zhu et al. proposed the dynamic audit services for outsourced storage in clouds [20]. Compared with them, our proposed scheme is more efficient in the computation cost. The computation comparison can be summarized in Table 1.

In Table 1, C_{exp} denotes the time cost of exponentiation on the group G_P; C_{mul} denotes the time cost of multiplication

Table 2: Comparison of communication cost.

Protocols	Chal	Response	Communication (rounds)
Zhu et al. [17]	$o(kc(\log_2 n + \log_2 q)$	$1G_P + 1G_Q + s\log_2 q$	5
Zhu et al. [20]	$c(\log_2 n + \log_2 q)$	$1G_P + s\log_2 q$	5
Wang [23]	$\log_2 n + 2\log_2 q$	$G_P + s\log_2 q$	5
Our protocol	$\log_2 n + 2\log_2 q$	$s\log_2 q$	5

on the group G_P; Ce denotes the time cost of bilinear pairing; c_{h_1} denotes the time cost of the hash function h_1. In other schemes, the sector must be in Z_P. Our scheme only requires the hash function h1s value which lies in Z_P. Thus, the hash function h_1 can be used to generate less block-tag pairs for the same file. Less block-tag pairs only incur less computation cost. This shows that our protocol can be implemented in mobile devices which have limited computation power.

Communication. In the phase proof, the communication overhead mainly comes from the challenge chal and response. The block-tag pairs are uploaded once and for all. After that, the phase proof will be performed periodically. Thus, the communication overheads mainly come from the Chal and responses. Suppose there are n message blocks stored in the CS. G_P and G_Q have the same order q. In chal, the verifier sends the challenge chal $= (c, k_1, k_2)$ to MCS. That is, the communication overhead is $\log_2 n + 2\log_2 q$. On the other hand, Zhu et al. [17], Zhu et al. [20], and Wang [23] proposed three different provable data possession schemes. We do the comparison under the same probability of detection. Our scheme and Wang's ID-PDP have the same total communication cost during the challenge phase. During the proof phase, the communication cost of the proof incurs less communication cost than Wang's ID-PDP. Compared with these three schemes, our scheme is more efficient in the communication cost. The communication comparison can be summarized in Table 2. In Table 2, $1G_1$ denotes one element of G_P and $1G_1$ denotes one element of G_Q.

6. Conclusion

In this paper, we propose a novel anonymous authentication scheme for roaming service in global mobility networks. Security and performance analysis show that the proposed scheme is more suitable for the low-power and resource-limited mobile devices and is secure against various attacks and has many excellent features.

Notations and Descriptions

G_p: Cyclic multiplicative group with generator q
G_Q: Cyclic multiplicative group with generator Q
Z_P^*: $\{1, 2, \ldots, P-1\}$
h, h_1: Three cryptographic hash functions
f: Pseudorandom function
π: Pseudorandom permutation
n: The block number
s: The sector number
$M = (M_1, \ldots, M_n)$: The stored file M is split into n blocks
$M_i = (\widetilde{M}_{i1}, \ldots, \widetilde{M}_{in})$: The block M_i is split into s blocks
$\widehat{n}$: The cloud server number
l_i: The index of the CS which stores the ith block-tag pair
CS_{l_i}: The CS which stores the ith block
$D_i = h(w_i, \mathrm{CS}_i) \cdot \prod_{j=1}^{s} M_{ij} \cdot P$: The record where i denotes the ith block
C_{exp}: The computation of exponentiation
C_{h_1}: The computation of hash function
C_{mul}: The computation of multiplications in group G
C_e: The computation of bilinear pairings
DO: Data owner
DU: The data user/client/requested
DS: Data stakeholder used to define both DO and DU
MCS: The multicloud server
HV: Home Verifier
v_i: The permutated index of $v_i = \pi_{k_1}(i)$.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported in part by Digital Right Management Technology Research and Development Project (1681300000119), Beijing Higher Education Young Elite Teacher Project (YETP0448), Specialized Research Fund for the Doctoral Program of Higher Education (2013114), Fundamental Research Funds for the Central Universities (2013RC0310), National Key Technology Research and Development Program (2012BAH08B02), National Natural Science Foundation of China (U1433105), and National 863 Program (2012AA012606).

References

[1] Y. Zhu, H. Hu, G. J. Ahn et al., "Collaborative integrity verification in hybrid clouds," in *Proceedings of the 7th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom '11)*, pp. 191–200, IEEE, 2011.

[2] T. Velte, A. Velte, and R. Elsenpeter, *Cloud Computing: A Practical Approach*, chapter 7, McGraw-Hill, New York, NY, USA, 1st edition, 2010.

[3] D. Hubbard and M. Sutton, "Top Threats to Cloud Computing V1.0, Cloud Security Alliance," 2010, https://cloudsecurityalliance.org/research/top-threats/.

[4] M. Lillibridge, S. Elnikety, A. Birrell, M. Burrows, and M. Isard, "A cooperative internet backup scheme," in *Proceedings of the Annual Conference on USENIX Annual Technical Conference*, pp. 29–41, 2003.

[5] Y. Deswarte, J. Quisquater, and A. Saidane, "Remote integrity checking," in *Integrity and Internal Control in Information Systems VI: IFIP TC11/WG11.5 Sixth Working Conference on Integrity and Internal Control in Information Systems (IICIS) 13-14 November 2003, Lausanne, Switzerland*, vol. 140 of *IFIP International Federation for Information Processing*, pp. 1–11, 2004.

[6] M. Naor and G. N. Rothblum, "The complexity of online memory checking," *Journal of the ACM*, vol. 56, no. 1, article 2, 2009.

[7] A. Juels and B. S. Kaliski Jr., "Pors: proofs of retrie vability for large files," in *Proceedings of the ACM Conference on Computer and Communications Security*, P. Ning, S. D. C. di Vimercati, and P. F. Syverson, Eds., pp. 584–597, 2007.

[8] S. J. Thomas Schwarz and E. L. Miller, "Store, forget, and check: Using algebraic signatures to check remotely administered storage," in *Proceedings of the 26th IEEE Internationa Conference on Distributed Computing Systems (ICDCS '06)*, p. 12, July 2006.

[9] D. L. G. Filho and P. S. L. M. Barreto, "Demonstrating data possession and uncheatable data transfer," IACR Cryptology ePrint archive, 2006.

[10] F. Sebé, J. Domingo-Ferrer, A. Martínez-Ballesté, Y. Deswarte, and J.-J. Quisquater, "Efficient remote data possession checking in critical information infrastructures," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 8, pp. 1034–1038, 2008.

[11] G. Yamamoto, S. Oda, and K. Aoki, "Fast integrity for large data," in *Proceedings of the ECRYPT Workshop Software Performance Enhancement for Encryption and Decryption*, pp. 21–32, June 2007.

[12] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan, "Auditing to keep online storage services honest," in *Proceedings of the 11th USENIX Workshop Hot Topics in Operating Systems (HOTOS '07)*, G. C. Hunt, Ed., 2007.

[13] 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. 598–610, ACM, New York, NY, USA, November 2007.

[14] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 5, pp. 847–859, 2011.

[15] H. Shacham and B. Waters, "Compact proofs of retrievability," in *Advances in Cryptology: Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security, Melbourne, Australia, December 2008*, vol. 5350 of *Lecture Notes in Computer Science*, pp. 90–107, Springer, Berlin, Germany, 2008.

[16] C. Wang, S. S. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, 2013.

[17] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 12, pp. 2231–2244, 2012.

[18] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in *Proceedings of the 4th International Conference on Security and Privacy in Communication Networks (SecureComm '08)*, pp. 1–10, ACM, Istanbul, Turky, September 2008.

[19] C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS '09)*, pp. 213–222, New York, NY, USA, November 2009.

[20] Y. Zhu, G.-J. Ahn, H. Hu, S. S. Yau, H. G. An, and C.-J. Hu, "Dynamic audit services for outsourced storages in clouds," *IEEE Transactions on Services Computing*, vol. 6, no. 2, pp. 227–238, 2013.

[21] H. Wang, "Proxy provable data possession in public clouds," *IEEE Transactions on Services Computing*, vol. 6, no. 4, pp. 551–559, 2013.

[22] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 9, pp. 1717–1726, 2013.

[23] H. Wang, "Identity-based distributed provable data possession in multi-cloud storage," *IEEE Transactions on Services Computing*, 2014.

[24] X. Chen, X. Huang, J. Li, J. Ma, W. Lou, and D. S. Wong, "New algorithms for secure outsourcing of large-scale systems of linear equations," *IEEE Transactions on Information and Forensics Security*, vol. 10, no. 1, pp. 69–78, 2015.

[25] D. Zhao, H. Peng, L. Li, and Y. Yang, "A secure and effective anonymous authentication scheme for roaming service in global mobility networks," *Wireless Personal Communications*, vol. 78, no. 1, pp. 247–269, 2014.

[26] J. Liu, Z. Zhang, X. Chen, and K. S. Kwak, "Certificateless remote anonymous authentication schemes for wireless body area networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 2, pp. 332–342, 2014.