Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH Open Access

Privacy preserving model: a new scheme for auditing cloud stakeholders

Abdul Razaque[1] and Syed S. Rizvi[2*]

Abstract

The Cloud computing paradigm provides numerous attractive services to customers such as the provision of the on-demand self-service, usage-based pricing, ubiquitous network access, transference of risk, and location independent resource sharing. However, the security of cloud computing, especially its data privacy, is a highly challengeable task. To address the data privacy issues, several mechanisms have been proposed that use the third party auditor (TPA) to ensure the integrity of outsourced data for the satisfaction of cloud users (CUs). However, the role of the TPA could be the potential security threat itself and can create new security vulnerabilities for the customer's data. Moreover, the cloud service providers (CSPs) and the CUs could also be the adversaries while deteriorating the stored private data. As a result, the objective of this research is twofold. Our first research goal is to analyze the data privacy-preserving issues by identifying unique privacy requirements and presenting a supportable solution that eliminates the possible threats towards data privacy. Our second research goal is to develop the privacy-preserving model (PPM) to audit all the stakeholders in order to provide a relatively secure cloud computing environment. Specifically, the proposed model ensures the quality of service (QoS) of cloud services and detects potential malicious insiders in CSPs and TPAs. Furthermore, our proposed model provides a methodology to audit a TPA for minimizing any potential insider threats. In addition, CUs can use the proposed model to periodically audit the CSPs using the TPA to ensure the integrity of the outsourced data. For demonstrating and validating the performance, the proposed PPM is programmed in C++ and tested on GreenCloud with NS2 by applying merging processes. The experimental results help to identify the effectiveness, operational efficiency, and reliability of the CSPs. In addition, the results demonstrate the successful rate of handling the negative role of the TPA and determining the TPA's malicious insider detection capabilities.

Keywords: Cloud computing, Privacy preserving model, Third party auditor, Cloud service provider, Cloud user, Authentication

Introduction

Cloud computing is an emerging IT environment that has significantly transformed everyone's vision of computing infrastructure, development models, and software distribution. Cloud computing is anticipated as the next generation high-tech paradigm for tomorrow's promise [1]. It provides several utilities as revolutionary gigantic paradigms where clients can remotely store valuable and confidential information as to avail from on-demand high quality computing resources [2]. While data outsourcing reduces the burden on the cloud users (CUs) from local storage and management, it brings several open problems related to the security and privacy of customer's outsourced data. On the other hand, cloud computing eradicates their physical control of data reliability and security, which can be addressed through the cooperation of three parties: the cloud service provider (CSP), the third party auditor (TPA) and the CU. Cloud computing has always been referred as virtualization of an existing server or data center. Subsequently, cloud computing is acknowledged as virtualization of existing data or data centers, providing multipurpose application support and enormous utility to remotely available users or clients [3]. This phenomenon leads to the cloud acting as a service, where services are provided upon request based on subscription or pay-per-use [4]. The cloud computing environment stores the

* Correspondence: srizvi@psu.edu
[2]Department of Information Sciences and Technology, Pennsylvania State University, Altoona, PA 16601, USA
Full list of author information is available at the end of the article

© The Author(s). 2017 **Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

valuable information and offers attractive user applications with reliable service support [5]. With the emergence of new technology, new categories of clouds and services are introduced such as supercomputing as service (SCaaS) and high-performance computing as a service (HPCaaS).

From a security perspective, dealing with large amounts of data is a challenge. The security of CSPs has been investigated thoroughly from a storage standpoint [6–10]. Two highly sought CSP features that guarantee privacy protection are data availability and integrity. Since the CUs do not have physical access to the outsourced data, it raises the question of data privacy protection in cloud computing, particularly for users with very limited computing resources. Moreover, there are several other factors regarding the CSPs corruption that can deceitfully smash the CUs outsourced data [11]. For instance, a CSP can attempt to sustain its reputation by hiding the security incidents about the customer's lost data [12]. Cloud services can be financially advantageous; however, there is no guarantee that the stored data will be secure and available at all times. If this continues to be an issue and is not thoroughly examined, the cloud computing environment may never reach its full potential.

The massive amount of outsourced data, along with the CU's limited resources, can present a daunting challenge for auditors when examining a cloud service [13]. One solution to this problem is to maintain a very low level of cloud storage overhead using minimal data retrieval operations. Although we face these challenges, it is still of utmost importance to develop trust between the CUs and CSPs.

This is where TPAs could assist in guaranteeing the CU's data privacy. Through a fair and impartial auditing process as well as the preservation of the CU's computational resources, a higher standard could be set for trust in cloud services. This auditing process could also help in improving the QoS provided by cloud-based platforms and resources. In the field of data security and privacy, the possibility of an insider threat can not be avoided. This raises the concern that a TPA could be malicious. The TPA will have free and open access to the CU's data and the cloud services, which leaves the CU vulnerable to attacks. Therefore, to circumvent potential financial losses or insider threats, a strict check and balance process over the TPA performance should exist. To address all of these issues, there is a need for a privacy-preserving model (PPM) that can provide a mechanism to authenticate all cloud stakeholders (i.e., CSP, TPA and CU) in order to safeguard the cloud computing environment.

To address privacy concerns, researchers have introduced models to ensure data correctness and privacy protection using protocols across multiple peers and servers [14–17]. Many of these proposed protocols support public certifiable remote integrity checking process [18–20]. However, without proper implementation, public certifiable auditing would perpetrate CUs a false perception that their data were undamaged in the CSP's data-centers.

The first privacy-preserving public auditing using blind technique was proposed in [21]. In [21], the verifier disables the TPA from detecting the file blocks by disguising the proof with some randomness. Subsequently, authors in [22] show the exploitation of the vulnerabilities originating from [18] when specific file blocks possess low entropy as well as allowing CUs to audit the TPA themselves to ensure an honest auditing process. Secure auditable models have been proposed in [23–26] to ensure integrity of outsourced data.

Although these computing paradigms introduce the audibility process, but fail to address the security concerns of the three parties (i.e., CSP, CU, TPA). To address these issues from a security perspective, this paper presents a novel data PPM. Our work is one of the unique data privacy-preserving contributions with a focus on auditing the three entities to reduce the trust deficit between the cloud stakeholders and improve the reliability of the outsourced data. Our proposed scheme provides the capability and a complete methodology to keep checks and balances between each entity in order to minimize data corruption, preserve data privacy, and restrict the misuse of resources.

Our proposed model ensures that the TPA does not have an access to the stored data by assigning session keys for each auditing task. Once the auditing process is completed, the assigned session keys will be expired and returned to the pool. In addition, the CU uses the strong authentication mechanism (e.g., triple Data Encryption Algorithm and SHA-256) to protect its outsourced data from potential attacks. The aggregation and derivative properties of our model help the three stakeholders to maintain strong authentication processes. A CU sets its priorities, QoS requirements, and anticipate timeframe for the completion of each task within the provided services. If the CU is not satisfied with the agreed requirements, the model immediately would enable the CU to refer the issue of incorrectness and inaccuracy of the paid services to the CSP. This feature of our proposed scheme not only improves the CUs experience but also warns the CSP to keep updating the cloud services and maintaining the agreed QoS according to the service level agreement (SLA).

Research contributions

Our research contributions are as follows:

- We provide the bounded-time interval by using session keys for each individual auditing service, preventing malicious insiders from accessing the client's confidential data stored in cloud servers.
- We embed the mod derivation process to fully secure the encrypted-message mechanism for authentication purpose. This feature does not provide any opportunity for the TPA to become an adversary, while handling the entire auditing process of each individual CU.
- Our proposed model not only ensures that the allocated resources are correctively delivered to the CU but also looks for the malicious behavior of clients to protect the data of other CUs stored in cloud servers. Thus, our proposed scheme addresses the security issues related to cloud multitenancy.
- We demonstrate and validate the data privacy-preserving through various experiments. Our simulation results demonstrate the effectiveness, reliability, and operational efficiency of CSPs. In addition, the results show the success rate for controlling the malicious activities of TPA and validating the TPA's malicious insider detection capabilities.

Adversary model

Our primary contribution in this research work is the proposed PPM that provides separate mechanism for auditing each cloud entity. An illustration of the proposed model is shown in Fig. 1. The model begins with a CU obtaining desired services from a CSP for storing the data, as shown in step 1a of Fig. 1. In response, the CSP delivers the desired cloud services to the requesting CU after setting up the necessary SLA (step 1b of Fig. 1). On the other hand, the CU wants to ensure the privacy preservation of its outsourced data. Therefore, it provides the details of the obtained services to the TPA, as shown in step 2 of Fig. 1. To audit the services provided to the CU, the CSP issues the key to the TPA for each auditing session (step 3 of Fig. 1). Once the session key is assigned to the TPA, the TPA starts the auditing process, as shown in step 4 of Fig. 1. The CSP checks the integrity of the TPA to determine whether the assigned keys are in use for further processes or not (step 5 of Fig. 1). If a TPA attempts to use any session key for obtaining the customer's confidential information, this cannot be done since all issued sessions keys are only effective for a specific period of time for a given auditing session. Finally, the TPA provides the audit report to the CSP and the CU respectively, as shown in steps 6a and 6b of Fig. 1.

A malicious third party auditor (TPA)

A cloud environment involves two main entities: one is the service provider (i.e., the CSP) and the others are the service utilizers (i.e., the CUs). Both of these entities interact with each other using various tools and technologies (e.g., databases, networks, virtualization, operating systems, transaction management, resource scheduling, concurrency control, load balancing, and memory management). Not only the CUs utilize the services offered by the CSPs but also often outsource their sensitive data to the cloud servers. The use of various technologies and the fact that the customer's data is not in-house bring numerous security challenges. Among those security challenges, encryption and data integrity is ranked as one of the top concerns of most of the CUs by the research community [5]. To address the issue of

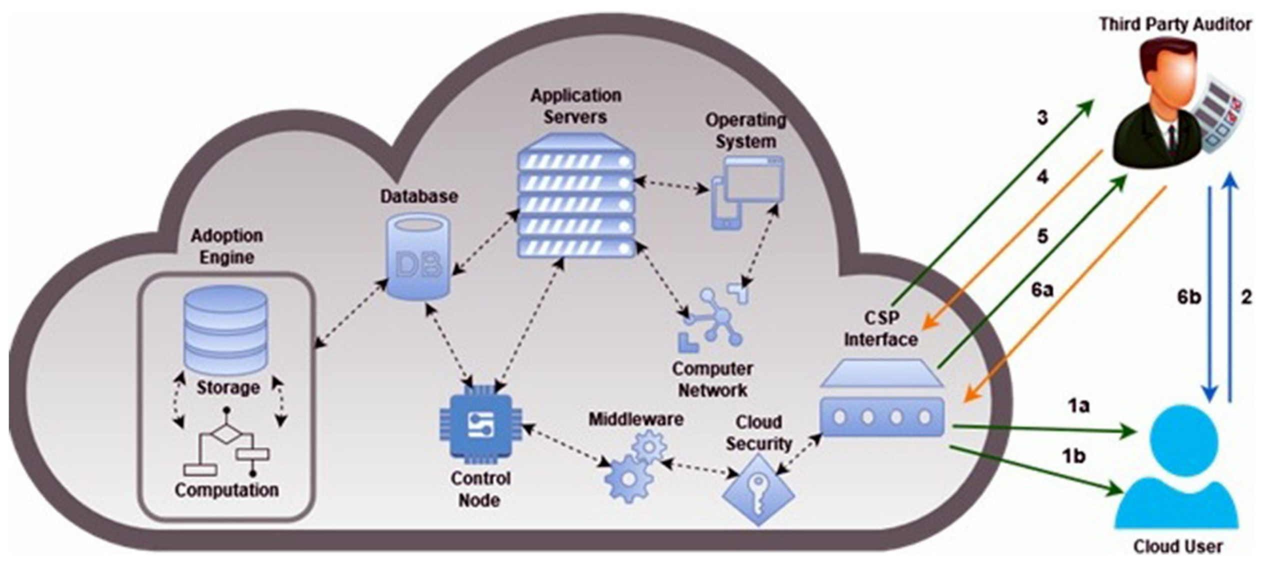

Fig. 1 Privacy-preserving service oriented architecture for cloud computing

data integrity and confidentiality, the use of a trusted TPA has been proposed by several researchers [11, 18]. It has been shown that the TPA can be very effective in performing several resource consuming tasks (such as checking the integrity of outsourced data and managing the encryption keys etc.) on behalf of CUs. Although the use of the TPA reduces significant computational burden on CUs, the possibility of malicious insiders at the TPA cannot be ignored. Therefore, there is clearly a need of a method that can be used to detect any malicious activities perform by the TPA. Since TPA servers as a proxy between the CUs and the CSPs, its integrity should be checked based on cloud services and customer's data. An illustration of the three entities (CUs, TPA, and CSP) is shown in Fig. 1.

A malicious cloud service provider (CSP)

To ensure the data confidentiality of customer's data, most of the existing work relies on the encryption-based schemes where the encrypted information can only be accessed by the entity (e.g., CU, CSP, and the TPA) that possess the encryption keys. Since CSPs may need to perform frequent computations on the customer's date for the offered services (e.g., data searches, modifications, additions, deletions and insertions), it is considered as a suitable candidate for holding and managing the encryption keys. In this way, the CU does not have to manage and assign the encryption key for each computation services provided by the CSPs. This also avoids the unnecessary delay that may cause due to the sharing/transmission of encryption keys between the CU and the CSP. However, it is unrealistic to assume that all CSPs are trustworthy. They can hide a data loss/leakage incident from CUs to maintain their high reputation. For instance, Byzantine failures, server conspiring attack, malicious data alteration, are some examples that may result a loss/leakage security incident. In a worst case scenario, a malicious CSP or malicious insiders at the CSP can exploit their own privileges by misusing the encryption keys to compromise the confidentiality of the customer's outsourced data, modify or even delete sensitive information without the knowledge of CUs. These malicious insiders can be categorized in two types. The first type is involved in debasing the stored CUs data files from individual servers. Once a server is compromised, a malicious insider can read and modify the contents of the customer's outsourced data. The second type of malicious insiders can compromise multiple data servers by taking advantage of multitenancy and collocation features of cloud environment. In either case, the confidentiality and the integrity of outsourced customer data is at high risk.

SLA violation by cloud user (CU)

In addition to the CSP, a CU can turn into a malicious entity by purposefully violating the terms and conditions of the SLA. Once a CU receives services from a CSP, it can sublet the services to other third-party organizations or individuals, which can raise serious security concerns and bring numerous management issues. Moreover, the subletting of cloud services to third-parties can slow down the service delivery process from the CSP site. All such malicious activities can severely affect the reputation of a CSP and can result into significant business loss. This clearly demands a periodic audit of CUs to detect any potential violations of SLA. For an unbiased and fair audit of CUs, a trusted TPA should be considered as the best candidate for this task.

Proposed privacy preserving model

We develop a new triangular data PPM to authenticate all the stakeholders (i.e., CU, CSP, and a TPA), as shown in Fig. 2. This model aims to ensure the integrity of the CU's data stored in the cloud data center, which can be retrieved on-demand at any time. Recent work has focused more on evaluating the reliability of the CSP in terms of its security and data privacy measures as well

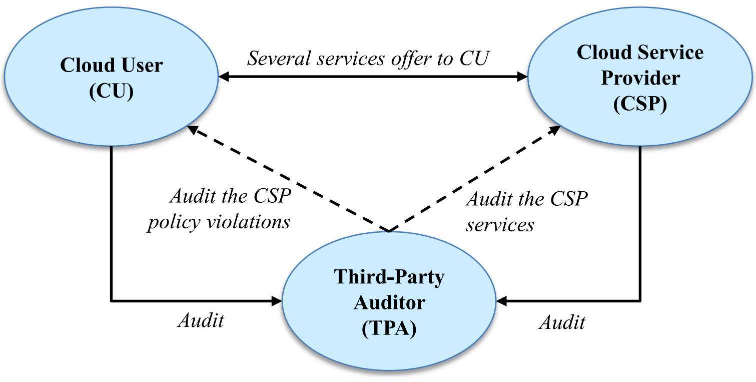
Cloud User (CU)
Several services offer to CU
Cloud Service Provider (CSP)
Audit the CSP policy violations
Audit the CSP services
Third-Party Auditor (TPA)
Audit
Audit

Fig. 2 A basic Cloud Computing Privacy-Preserving Authenticating Model

as its compliance with its SLA. However, little work has been done to evaluate the reliability of the CU and the TPA. Therefore, our proposed model evaluates the CU's integrity in terms of their ability to not violate any of the agreed upon rules defined and set by the CSP in the SLA. Furthermore, the TPA audits the services provided to the CU and ensures the TPA's integrity (i.e., the TPA is not disclosing the CU's contents from the information obtained through the auditing process).

The CU and the CSP provide a mechanism to audit the TPA. Thus, the CU verifies whether the TPA performs the assigned auditing within the given specification and the time frame. Furthermore, the CSP also verifies whether the TPA performs its auditing tasks using the assigned time-released session keys.

The proposed model performs the following functionalities. For each functionality of the cloud stakeholder, we provide a mathematical model to derive closed-form expressions. These functionalities are as follows:

- Service Types
- Malicious Insider and TPA
- CU Authentication for TPA through CSP
- CSP Authentication Process

Service types

This section discusses the types of services provided to the CU and derives a closed-form expression to reflect the correct delivery of cloud services. Let us assume that 'N' is the number of CUs in a given computing environment where each individual cloud customer (n_i) can obtain a maximum of K number of services from a CSP. The total number of CUs can be expressed as follows: $N = n_1, n_2, n_3, \ldots, n_n$.

A CU can access the cloud services that are within the SLA, since services beyond the SLA are restricted. Thus, the CSP can define security limitations 'L' for assigned services (K) such that each offered service (k_i) can have one or more limitations (i.e., L_i where $i = 1$ to K). Therefore, the security limitations (L) can be expanded up to the total number of services (K) offered by a CSP. This relationship can be expressed as follows:

$$L = [L_1,\ L_2, L_3, \ldots,\ L_K] \tag{1}$$

In our proposed model, CUs are allowed to define their own priorities, quality of service (QoS) requirements, and anticipate timeframes for the completion of several tasks within the organization. On the other hand, CUs are required to meet these specified requirements within the obtained services and the allotted timeframe defined in the SLA (i.e., allowed time that a CU can use services until the service-contract expires). We illustrate the allocated service time for the CU to use the cloud resources as follows:

$$\{A^x\ (k), k \in (1, 2, 3, \ldots,\ K)\} \tag{2}$$

where 'A^x' is the amount paid for each service for specific period of time, 'x' is the user that can access the service and 'k' is the type of service.

Let us consider that each service involves an arbitrary data set $Z = \{Z_1, Z_2, \ldots, Z_n\}$ that can be selected independently. Each query for every member of the data set is denoted by $Q_i = (q_i, f_i)$; for $i = 0,1,\ldots,\ n$ where $q_i \subseteq [n]$ is the sub set of the data-set member. Where f_i: specifies the function (e.g., max or sum). Thus, each query has an answer in the form defined as: $\Delta \mathrm{q}_i = f_i(q_i)$. This can be applied to the subset of the data set entry as follows: $\{Z_j \mid j \in q_i\}$.

Theorem 1 *In the proposed PPM, CUs authorize the TPA to audit the obtained services by providing the few samples of services.*

Proof Let us prove that the paid amount for provided services to CUs are correctly stored on the CSP's server. Taking this into account, the confirmation process can be done as follows:

$$A_{dit} = \sum_{i=0}^{n} (\mathrm{A}^x) \equiv \sum_{j=1}^{n} j(k_1) + j(k_2), \ldots, j(k_n) \underset{\approx}{\subsetneq} T_s \tag{3}$$

Equation (3) shows the total amount paid against all the services that are signed in the contract. The TPA compares the contracted services with the samples provided by CUs. The TPA based comparison can be shown in (4) as follows:

$$C_{sp} = \sum_{n=0}^{\infty} n\left(N\right) \cong C_{id} \left\{ \sum_{j=1}^{n} j(k_1) + j(k_2), \ldots, j(k_n) \right\} \underset{\approx}{\subsetneq} T_s \tag{4}$$

In Eq. (4), the CSP confirms the CU's identity with its record. If the CUs are legitimate, the TPA is allowed to compare the samples with the original services. This matching process can be shown in (5) as follows:

$$A_{dit} = \sum_{n=0}^{\infty} n(\omega) \cong\ T_s \tag{5}$$

In Eq. (5), the TPA initiates the matching process of signed contracts with the samples on record. If the samples match with the signed contracted services, the provided service types are considered as legitimate. Furthermore, the matching process helps increase the trust level of CSPs. The functional service type behavior is depicted in Fig. 3 and its used notations are described in Table 1.

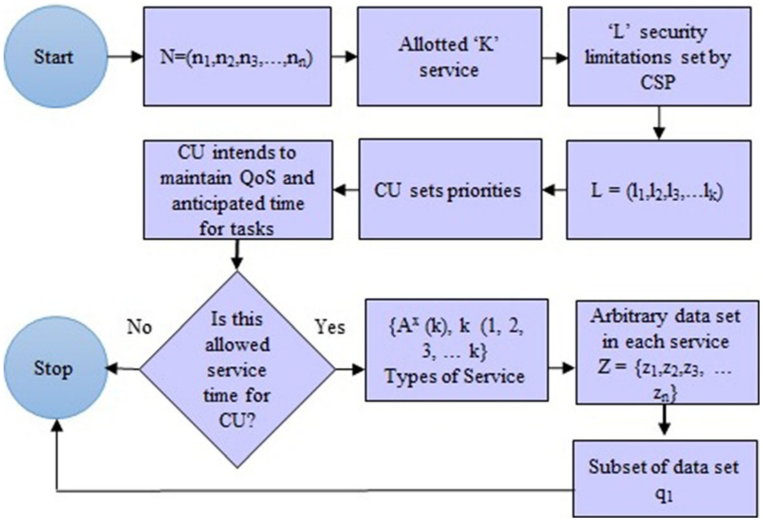
Start
N=(n1,n2,n3,…,nn)
Allotted 'K' service
'L' security limitations set by CSP
L = (l1,l2,l3,…lk)
CU sets priorities
CU intends to maintain QoS and anticipated time for tasks
Is this allowed service time for CU?
No
Yes
Stop
{Ax (k), k (1, 2, 3, … k} Types of Service
Arbitrary data set in each service Z = {z1,z2,z3, … zn}
Subset of data set q1

Fig. 3 Functional behavior of service-type to ensure the privacy-preserving authentication

Malicious insider and TPA

Let us assume that queries and their answers are protected by 'γ^{Φ}' for each entry 'Z_j' of malicious insiders which is bounded by the time interval of 'T_I'. An attacker can attempt to generate malicious entry 'Z_j' to capture the query and its protected answers during the bounded time interval. However, our objective is to prevent an attacker from exploiting the privacy of the protected 'γ^{Φ}' queries and answers. The method for protecting the information is to use the bounded-time with the session keys for each individual auditing-service. The CSP releases the session keys with the bounded-time for each auditing service. Once the auditing service is completed, the session keys will be returned and will not be regenerated until the CSP releases another poll of session keys.

Table 1 System parameters and definitions for service type

Notations	Description
A^x	Amount paid for each service for a particular period
f_i	Specifies the function (e.g., max or sum).
k	Service type
C_{sp}	Cloud service provider
L	Security limitations
N	The number of cloud users
$\Delta q_i = f_i(q_i)$	Correct answer of each query stored in a cluster
$Q_i = (q_i, f_i)$	Each query for every member of data set
x	The user that access the service
$n(\omega)$	Contracted services
$Z = \{Z_1, Z_2, \ldots, Z_n\}$	Arbitrary data set
$\{Z_j \mid j \in q_i\}$.	Subset of data-set entry
A_{dit}	Third party auditor
C_{id}	Identity of cloud users

In addition, the malicious insider could be a team member of the TPA or a CSP. As a result, their illicit attempt is not successful $Z_j \notin T_I$ to exploit the protected data $Z_j \in T_I$. Therefore, the sequence of queries and their answers are protected 'γ^{Φ}' from entry 'Z_j' due to bounded-time interval 'T_I'. These protected queries and answers can be written as follows:

$$\gamma^{\Phi}, j, T_I \{(q_1, q_2, \ldots, q_n), (a_1, a_2, \ldots, a_n)\} = \begin{cases} 1 \ \textit{if} \ 1(1+\gamma^{\Phi}) \leq \dfrac{\beta\,(Z_j \in T_I \,|\sum_{i=0}^{n} f_i(q_i) \ = a_i}{\beta(Z_j \in T_I)} \leq (1+\gamma^{\Phi}) \\ 0 \quad \textit{Otherwise} \end{cases} \tag{6}$$

where β: prediction of malicious insider.

Let $Z = \{Z_1, Z_2, \ldots, Z_n\}$ be the data set where Z_i is selected separately. This complies with the following statement as: (γ^{Φ}, β).

- The malicious insider poses the query as $Q_m = (q_m, f_m)$.
- The TPA decides whether to permit the query of a malicious attacker 'Q_m' or not. The TPA responds with the following expression $\Delta q_a = f_a(q_a)$.If Q_m is permitted, Δq_a is rejected, otherwise.
- Malicious insider is successful if $\beta = f_a(q_a)$

The malicious insider and TPA process is depicted in Fig. 4. Table 2 lists all the parameters used for malicious insiders and TPA modules.

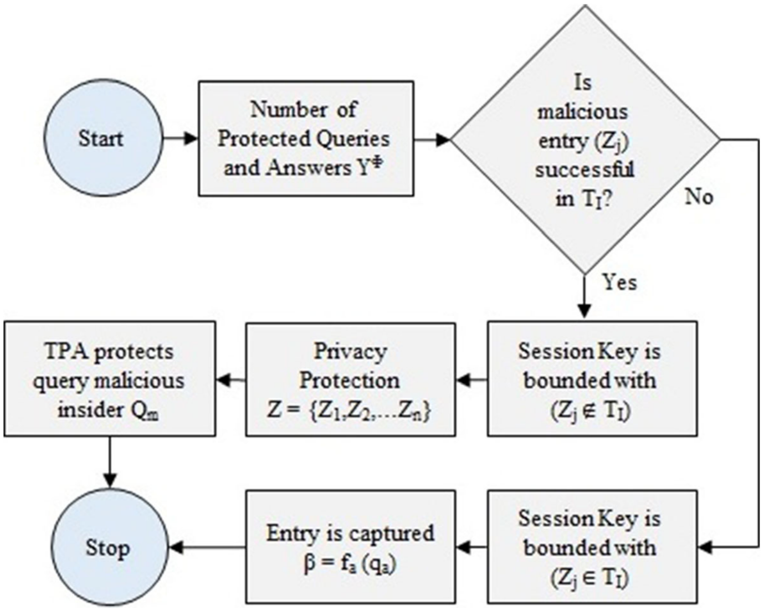
Start
Number of Protected Queries and Answers Y^Φ
Is malicious entry (Zj) successful in TI?
No
Yes
Session Key is bounded with (Zj ∉ TI)
Privacy Protection Z = {Z1,Z2,…Zn}
TPA protects query malicious insider Qm
Stop
Session Key is bounded with (Zj ∈ TI)
Entry is captured β = fa (qa)

Fig. 4 Working process of malicious insider and TPA for guaranteeing the privacy-preserving authentication

CU authentication for TPA through CSP

Let us assume that 'φ' is the outsourced data file of the CU that consists of 'n' number of blocks: $n = \{b_1, b_2, b_3, \ldots, b_n\}$. The CU chooses the authentication message $M_a = \left(\iiint_{\varpi=0}^{n}\Delta s_{\varpi} + log_2 \mathrm{K}'^{n}\right)$ using a secure and strong encryption scheme (e.g., AES Algorithm with SHA-256) that is modeled as $'\iiint_{\varpi=0}^{n}\Delta s_{\varpi}'$ with '$log_2\mathrm{K}'^{n}$' size of the message. The chosen authenticated message by the CU is illustrated as follows:

$$M_a = \left[\left(\iiint_{\varpi=0}^{n}\Delta s_{\varpi} + \left(log_2\dot{\mathrm{K}}\right)^{n}\right)\{\,\dot{\mathrm{x}}.\dot{\mathrm{y}}(\nabla\beta\delta)\}\right] \tag{7}$$

where ϖ: random number assigning the initial value to product, $(\nabla\beta\delta)$: combination of two randomly chosen values consists of variable lengths and $\dot{\mathrm{x}}.\dot{\mathrm{y}}$: Quotient function that is used as a mod function. The detail of authenticated message with the encryption process is given as follows: Replacing the mod function $\dot{\mathrm{x}}.\dot{\mathrm{y}}\cdot$:

Table 2 System Parameters and Definitions for malicious insider and TPA

Notations	Description
$Q_m = (q_m, f_m)$	Query posed by a malicious insider
Q_m	Malicious attacker
T_I	Bounded time interval
(γ^{Φ}, β).	Prediction of the malicious attacker against protected data (queries and answers)
γ^{Φ}	Protected data comprising of query and answers
Z_j	The malicious insider entry
$Z_j \notin T_I$	Unsuccessful attempt
$Z_j \in T_I$	Successful attempt
$\Delta q_a = f_a(q_a)$	Response from TPA for malicious insider to authenticate itself
$\beta = f_a(q_a)$	Successful prediction of malicious insider

$$M_a = \left[\left(\iiint_{\varpi=0}^{n}\Delta s_{\varpi} + \left(log_2\dot{\mathrm{K}}\right)^{n}\right)\{f(z)(\nabla\beta\delta)\}\right]$$

Differentiation of $f(z)$ is required to model the mod derivation

$$M_a = \left[\left(\iiint_{\varpi=0}^{n}\Delta s_{\varpi} + \left(log_2\dot{\mathrm{K}}\right)^{n}\right)\left\{\,\frac{s(z)}{t(z)}(\nabla\beta\delta)\right\}\right]$$

Determining the equivalency of mod derivation to support encryption process yields:

$$M_a = \left[\left(\iiint_{\varpi=0}^{n}\Delta s_{\varpi} + \left(log_2\dot{\mathrm{K}}\right)^{n}\right)\left\{\frac{\Delta d}{\Delta dz}\,\frac{s(z)}{t(z)}(\nabla\beta\delta)\right\}\right]$$

Applying the product rule for mod derivation to secure the encryption process yields the following expression:

$$M_a = \left[\left(\iiint_{\varpi=0}^{n}\Delta s_{\varpi} + \left(log_2\dot{\mathrm{K}}\right)^{n}\right)\right.$$
$$\left.\left\{t(z).\frac{\Delta d}{\Delta dz}[f(z)] + \frac{\Delta d}{\Delta dz}[t(z)].f(z).(\nabla\beta\delta)\right\}\right]$$

Simplifying the process to support the mod derivation and to show the division with two randomly generated numbers $(\nabla\beta\delta)$ for encryption is given as:

$$M_a = \left[\left(\iiint_{\varpi=0}^{n} \Delta s_{\varpi} + \left(log_2 \acute{\mathrm{K}}\right)^n\right)\left\{\frac{\frac{\Delta d}{\Delta dz}[s(z)] - \frac{\Delta d}{\Delta dz}[t(z)].f(z)}{t(z)}.(\nabla\beta\delta)\right\}\right]$$

Once again, the differentiation of *f*(*z*) is required to hide the encryption and the length of the data file. This yields the following expression:

$$M_a = \left[\left(\iiint_{\varpi=0}^{n} \Delta s_{\varpi} + \left(log_2 \acute{\mathrm{K}}\right)^n\right)\left\{\frac{\frac{\Delta d}{\Delta dz}[s(z)] - \frac{\Delta d}{\Delta dz}[t(z)].\frac{s(z)}{t(z)}}{t(z)}.(\nabla\beta\delta)\right\}\right]$$

To show the complete message encryption to protect the CU from a malicious insider, we derive the following equation:

$$M_a = \left[\begin{array}{c}\left(\iiint_{\varpi=0}^{n} \Delta s_{\varpi} + \left(log_2 \acute{\mathrm{K}}\right)^n\right) \\ \left\{\dfrac{\dfrac{\Delta d}{\Delta dz}[s(z)].t(z) - \dfrac{\Delta d}{\Delta dz}[t(z)].s(z)}{[t(z)]^2}.(\nabla\beta\delta)\right\}\end{array}\right] \tag{8}$$

If a user is the legitimate CU, then a copy of $(\nabla\beta\delta)$ is generated at the server of a CSP. The CSP is required to identify the generated authentication message of a CU such that $M_a = \left(\iiint_{\varpi=0}^{n} \Delta s_{\varpi} + log_2 \acute{\mathrm{K}}^n\right)$. The authentication message is generated by using a calculated aggregated authenticator to validate whether the message was generated by a legitimate CU. The resultant message is given as follows:

$$\acute{\mathrm{M}}_a = \prod_{k=1}^{n} (\nabla\beta\delta)A_k \leq = \sum_{n=0}^{\infty} (\nabla\beta\delta)^N \tag{9}$$

where $(\nabla\beta\delta)^N$ represents the aggregated authenticator value.

Once the CU's authentication message matches with the aggregated authenticator of the CSP, the CU is considered as a legitimate client within the CSP's domain. The TPA can then obtain the secret key from the CSP for each authentication message of the CU to initiate the auditing process. Specifically, the TPA requires session keys for comparison of each audit. Thus, it will be harder for the TPA to expose the outsourced data file 'ф' of a CU once the auditing process is completed. If the TPA attempts to exploit the outsourced data file of a CU after the auditing process, it needs a new session key that is not assigned to the TPA by the CU. However, the TPA may experience the problem because the auditing process will be limited if auditing is required more than one time.

All possible secret keys should be able to overcome this drawback in advance. Once all fixed secret keys are exhausted, the CU can then retrieve and publish the data, which complies with the privacy-preserving requirements depicted in Fig. 5. Table 3 lists all the parameters used in the TPA module.

CSP authentication process

The CSP's integrity is of paramount importance for correct delivery of the services. The CSP's responsibility is to maximize the guarantee for providing shared resources to the customers. The CSP provides numerous resources, which are not only shared by multiple CUs, but also dynamically reallocated. This allocation of resources needs to be authenticated for the CU's satisfaction. Thus, the TPA not only audits the CU but also makes sure that the assigned service is correctly provided as per specification of the CU.

The auditing process begins when the TPA generates a sample "check message" $'C_m'$ against each provided service to the CU to confirm the provided service by the CSP. The TPA chooses a random value $'\forall\partial'$ for total received services $T_s = \{s_1, s_2, s_3, \ldots, s_n\}$ for each service provided as quantified in Eq. (1). Each service has different characteristics that are attributed by $'s_f'$. Thus, $s_f \in T_s$. The TPA checks some features of service by sending a "check message" $C_m = \{s_f, \ldots \forall\partial\}$ $s_f \in T_s$ to the CSPs server. Upon receiving the "check message", the server generates a response against "check message" to guarantee the storage of data correctly. Therefore, the server also selects the random number $\varpi \leftarrow \mathbb{R}_g$ and computes its value such as: $\forall\partial = (\rho, \ell)^{\varpi} \in \mathbb{R}_g$. The following parameters should be noted: s_f: features of each service, ρ: the service returned by cloud server, ℓ: order-set for the features of service, and $\mathbb{R}_g$: Random generator.

Let us assume that ρ^* represents the combination of sampled blocks that are specified in $'C_m'$, Thus, sampled check blocks can be described as:

$$\rho^* = \sum_{s_f \in T_s}^{s_n} \forall\partial L_k \tag{10}$$

In response to the sampled blocks, the server computes the requested service $'\rho'$ for the TPA to prove its integrity and ensure that the CU is correctly provided the requested services. To satisfy the TPAs requested query, the requested service should be delivered to the TPA using full encryption along with the sampled blocks. This compares the sampled blocks of requested service with the original service provided to the CU and can be expressed as follows:

$$\rho = \varpi + \sum_{s_f \in T_s}^{s_n} \forall\partial L_k + \left[\left(\iiint_{\varpi=0}^{n} \Delta s_{\varpi} + \left(log_2 \acute{\mathrm{K}}\right)^n\right)\right] \times \left\{\frac{\frac{\Delta d}{\Delta dz}[s(z)].t(z) - \frac{\Delta d}{\Delta dz}[t(z)].s(z)}{t(z)}.(\nabla\beta\delta)\right\} \tag{11}$$

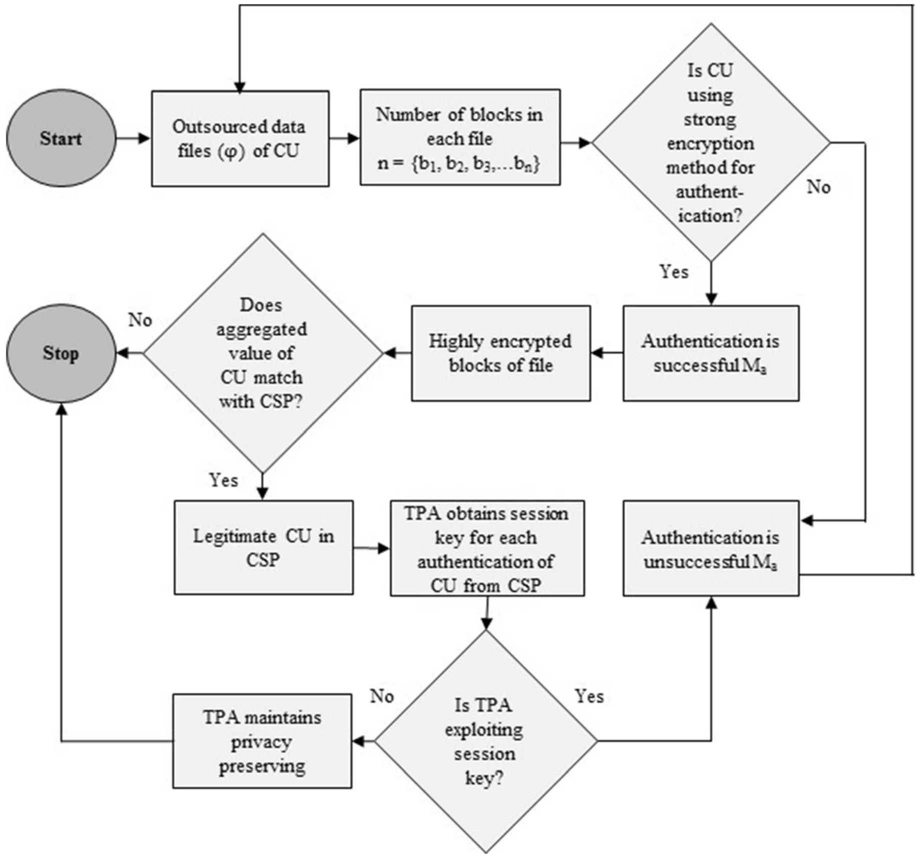
Start
Outsourced data files (φ) of CU
Number of blocks in each file n = {b1, b2, b3,…bn}
Is CU using strong encryption method for authentication?
No
Yes
Authentication is successful M3
Highly encrypted blocks of file
Does aggregated value of CU match with CSP?
No
Stop
Yes
Legitimate CU in CSP
TPA obtains session key for each authentication of CU from CSP
Authentication is unsuccessful M3
Is TPA exploiting session key?
No
Yes
TPA maintains privacy preserving

Fig. 5 Authentication of CU to maintain the privacy preserving process

If the CSP fails to provide the proper services according to the SLA, it may attempt to launch different kinds of attacks such as a forge attack, replay attack, distributed denial-of-service attack, etc.

To expose the attacks that are launched by a CSP, we examine if the "check message" 'C_m' or its data tag '$\boldsymbol{\aleph}_t$' is bastardized. If they are corrupt, the TPA's verification request that determines the correctness of the service assigned to the CU cannot be processed. As a result, the server impersonates the original TPA's "check message" including its data tag '$\boldsymbol{\aleph}_t$', and replaces them with the fake "check message" 'F_m' and fake tag 'F_t''. The impersonated message '$\nexists\,\mathbb{R}_m$' forwarded to the TPA with encryption can be defined as:

$$\nexists\mathbb{R}_m = \prod_{i=0}^{n}{}'\mathrm{O}\left(F_m,\ F_t\right)^{\mathrm{D}b_i} + \left[\left(\iiint_{\varpi=0}^{n}\Delta s_{\varpi} + \left(log_2\acute{\mathrm{K}}\right)^n\right)\right] \times \left\{\frac{\frac{\Delta d}{\Delta dz}\left[s(z)\right].t(z)-\frac{\Delta d}{\Delta dz}\left[t(z)\right].s(z)}{t(z)}.(\nabla\beta\delta)\right\} \tag{12}$$

where, 'O: Output check-message, and $\mathrm{D}b_i$: Blocks of data sent in check-message.

Once the CSP forwards an encrypted impersonated message '$\nexists\,\mathbb{R}_m$' to the TPA, the TPA compares the CSP's message with (C_m, $\boldsymbol{\aleph}_t$) to help determine a CSP's illegitimate action 'Ʀ' that is compared and shown in Fig. 6.

If $Ʀ = \left[\left\{\prod_{i=0}^{n}{}'\mathrm{O}(F_m, F_t)^{Db_i}\right\} \equiv \left(\varpi + \sum_{S_f\in T_s}^{S_n}\forall\partial L_k\right)\right]$, the CSP provides legitimate service(s) to the CU and fulfils the data-privacy requirements. On the other hand, if the above expression is not equivalent, it implies that the user data confidentiality is compromised and the CSP provides unstandardized services to the CU. Table 4 shows the parameters used for authenticating CSPs.

Table 3 System Parameters and Definitions of CU module for TPA

Notations	Description
$(\nabla\beta\delta)^N$	Aggregated authenticator value
$n = \{b_1, b_2, b_3, \ldots, b_n\}$	Number of data blocks
$\log_2\acute{K}^n$	Size of the encrypted message
$\iiint_{\varpi=0}^{n}\Delta s_{\varpi}$	Secure and robust encryption method
$\dot{x}.\dot{y}\cdot$	Quotient function that is used as mod
φ	Outsourced data file of CU
ϖ	Random number

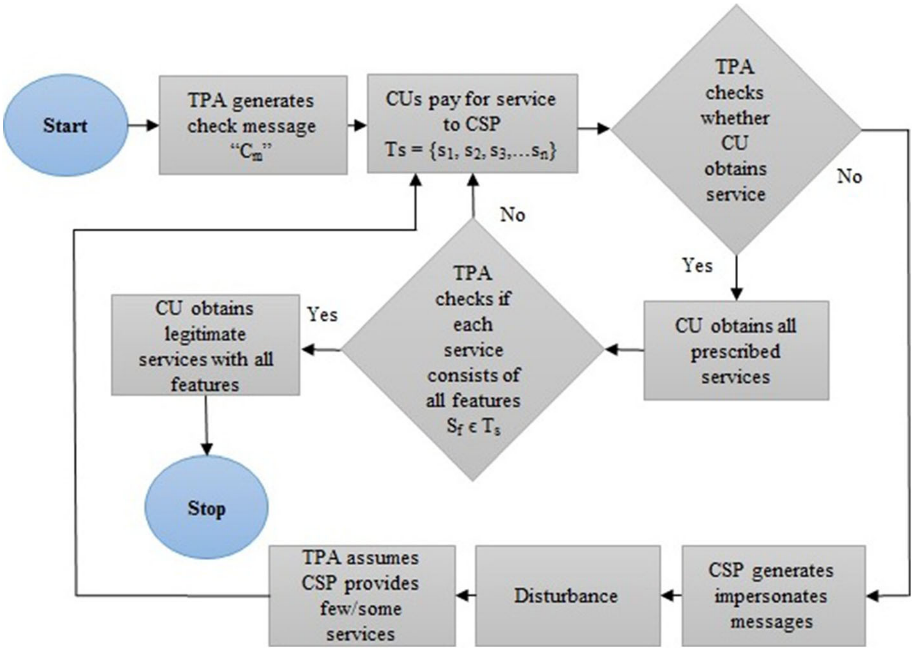
Start
TPA generates check message "C_m"
CUs pay for service to CSP $T_s = \{s_1, s_2, s_3, \ldots s_n\}$
TPA checks whether CU obtains service
No
Yes
CU obtains all prescribed services
TPA checks if each service consists of all features $S_f \in T_s$
No
Yes
CU obtains legitimate services with all features
Stop
CSP generates impersonates messages
Disturbance
TPA assumes CSP provides few/some services

Fig. 6 Identifying the authentication process of CSP for fulfillment of data-privacy requirement

Experiments and performance evaluation

We test the performance of the proposed privacy-preserving model (PPM) involving the three entities (i.e., CSP, CU, and TPA). The proposed model is programmed in C++ and tested on the GreenCloud simulator, the extension of the Network Simulator 2 (NS2). The GreenCloud and NS2 are installed using the merging process. A 2.8 GHz Pentium Dual Core CPU with 5 GB RAM powered computer using a default Ubuntu 12.04 OS is used to run the experiments. The test machine uses a 64-bit version of Windows 8. The GreenCloud presents the repeatable and controllable environment to show the realistic behavior. We test the proposed model for different scenarios to demonstrate the validity of the three entities. Specifically, the following parameters are observed:

- Effectiveness of CSP
- Operational efficiency of CSP
- TPA malicious attempts and successful rate
- Reliability of CSP versus number of auditing

Effectiveness of cloud service provider (CSP)

In the first scenario, we investigate the effectiveness of the CSP based on the provided services to CUs. The effectiveness of the proposed PPM was evidenced in the obtained results which are relatively similar to the realistic environment. The PPM measured the performance of the CSP for ideal (PPM-I), expected (PPM-E), and worst-case (PPM-W) scenarios, as depicted in Figs. 7 and 8. We analyze the effectiveness of the CSP by increasing the number of provided services to the CU. As the number of services to CU is increased over the time, a slight decrease is observed in the effectiveness of the CSP. In all three cases, the proposed PPM demonstrates the realistic behavior of the CSP. The effectiveness of the CSP in the expected case was the reconfirmation that the PPM-E was not entirely approaching the theoretical maximum of the PPM-I. Similarly, in the case of PPM-W, the PPM-E performance was more realistic for the effectiveness of the CSP.

It was validated that the proposed model is well-ordered and remained bounded by a different number of services for most of the simulation time. Hence, it can be implied that the proposed PPM system remains stable in determining the effectiveness of the CSP. The reason for the performance degradation of the CSP was mainly the involvement of some impairing factors such as malicious insiders and outsiders that added into the system as more services are delivered by the CSP. If the role of the malicious adversary is entirely neglected, the effectiveness of the CSP could be much better. However, the possibility of all potential attacks on the CSP cannot be disregarded; otherwise, the effectiveness of a CSP in a realistic environment could not be analyzed in an organized way.

Operational efficiency of cloud service provider (CSP)

Both Figs. 9 and 10 demonstrate the operational efficiency of the CSP versus the number of users. The simulation results exhibit this by increasing the number of

Table 4 System Parameters and Definitions for the authentication process of CSP

Notations	Description
C_m	Check the message used by TPA to examine the correctness of contents stored on a cluster
Db_i	Blocks of data sent in check-message
(F_m, F_t)	Fake check message and fake tag generated by CSP if not provided the required service to CU
(γ^{Φ}, β).	Prediction of the malicious attacker against protected data (queries and answers)
γ^{Φ}	Protected data comprising of query and answers
Z_j	The malicious insider entry
$Z_j \notin T_l$	Unsuccessful attempt
$Z_j \in T_l$	Successful attempt
$\Delta q_a = f_a(q_a)$	Response from TPA for malicious insider to authenticate itself
$\beta = f_a(q_a)$	Successful prediction of malicious insider
s_f	The features of each service assigned to CU
$\varpi \leftarrow \mathbb{N}$	Generation of random number by TSP
ρ^*	Representing the combination of sampled blocks
$\mathbb{R}_g$	Random generator of TSP
$\nexists \mathbb{R}_m$	Impersonated message created by the server
'O	Output for check-message
$\aleph_t$	Data tag generated by TPA to validate the CSPs provided services to CU
R	Determining the CSPs illegitimate action
$\frac{s(z)}{t(z)}$	Derivative of two quotients
$\forall \partial$	TPA chooses a random value for checking the CSPs provided service
ℓ	Set of the features

CUs which decreases the operational efficiency for all three cases (i.e., PPM-I, PPM-E, and PPM-W), as shown in Fig. 10. The generated scenarios for determining the operational efficiency of the CSP is more realistic than expected for the proposed PPM. Therefore, we focus more on some of the limitations affecting the CSP such as rapid business changes, highly competitive markets, unpredictable economic environments, and dealing with more regulations. In accordance with our expectations, as the number of CUs increases (e.g., see Fig. 10), the overall operational efficiency for the three cases should fluctuate as compare to Fig. 9. However, the worst case scenario (PPM-W) profoundly affected the operational efficiency, as depicted in both Figs. 9 and 10.

In addition to an increase in the CUs and their respective services, the network and server delays also play a role in degrading the overall operational efficiency of the CSPs. The operational efficiency of the CSP could be improved, if we guarantee that all connections are protected prior to binding the cloud applications. For more accurate results, we need to know whether CUs are required to sign into a protected connection first, and then into the cloud application or if it is manageable and reachable ubiquitously.

TPA malicious attempts and successful rate

The primary challenge in protecting data privacy is to handle the malevolent role of the TPA. In the cloud storage system, the CUs host their data on the CSP's servers that can be accessed from anywhere. Due to data outsourcing, the TPA's fraudulence could damage the CSP's image. In addition, the confidential information of CUs could be exploited or leaked to adversaries. In Fig. 11, we demonstrate the malicious attempts of the TPA versus the successful malicious-detection of the proposed PPM.

While handling the malicious attempts of the TPA, different scenarios were generated for three different cases: ideal, expected, and worst. The PPM captured all the malicious attempts in an ideal case (PPM-I) and provided a success-rate reaching 100%. On the other hand, the PPM received the successful capturing rate of 96.6% and 88.3% for expected (PPM-E) and worst case (PPM-W) scenarios, respectively. The expected successful rate could be improved if a malicious insider did not help the TPA to gain access to sensitive information by recuperating the data blocks from a data proof. However, the malicious role of a CSP cannot be ignored, leaving the TPA's newly generated content keys useless against each CU's stored information. The simulation results are comparatively similar to a realistic environment and also confirm the correctness of the proposed PPM.

Reliability of cloud service provider versus number of auditing

The reliability of the CSP is a significant concern. When referring to the reliability of the CSP, it is mainly measured in terms of how secure the customer's data is at the data center and how securely the cloud services are delivered to the CUs. Figure 12 demonstrates the reliability of the CSP for different scenarios: an ideal, expected, and worst case. The primary goal of our experiment is to validate the PPM for various scenarios to evaluate the reliability ratio of the CSP versus the number of audits performed on the customer's data and the cloud services.

The simulation results show that the PPM-I produced 90.2% reliability of the CSP. However, in the expected and worst cases, the results show further decline in the reliability of the CSP as more auditing is done. Hence, the PPM-W confirmed the higher drop rate in the reliability of the CSP. Based on the number of auditing, the CSP will be able to maintain approximately 90.2% reliability for PPM-I, as shown in Fig. 12. In an expected

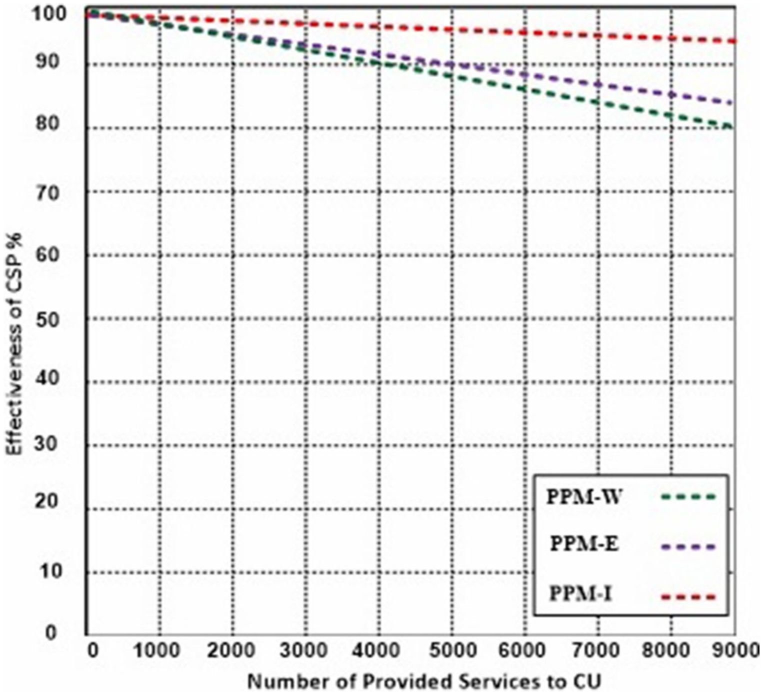

Fig. 7 Effectiveness of CSP VS relatively smaller number of services to Cloud Users

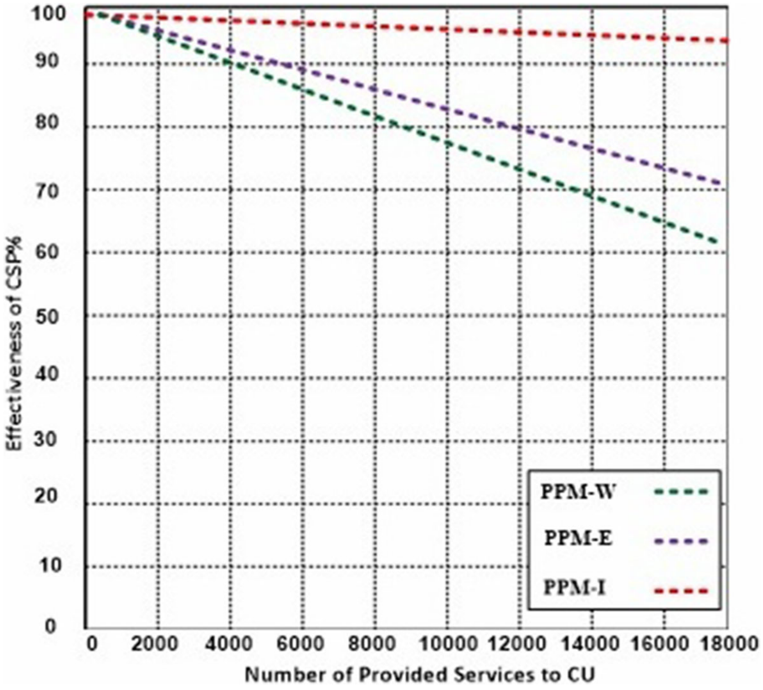

Fig. 8 Effectiveness of CSP VS relatively larger number of services to Cloud Users

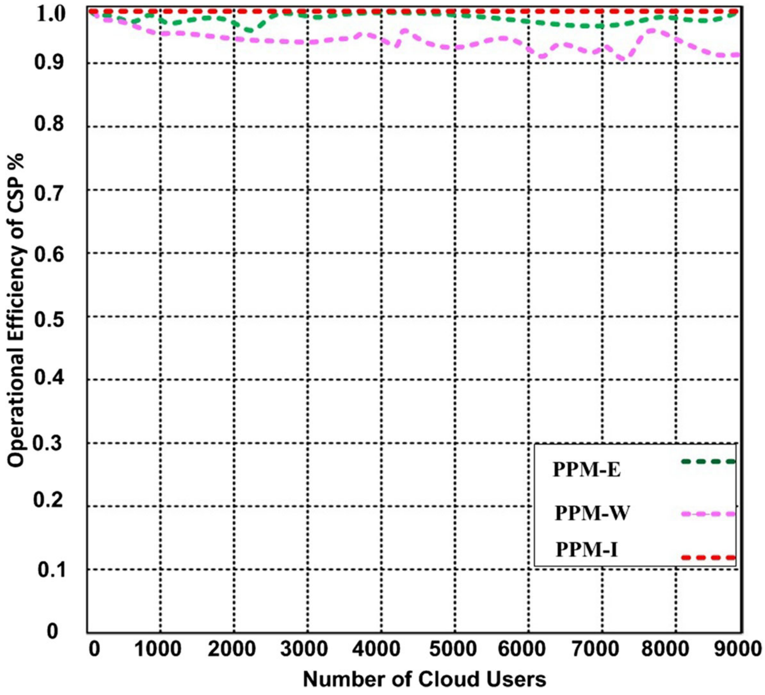

Fig. 9 Operational Efficiency VS number of cloud users (for a relatively smaller cloud network)

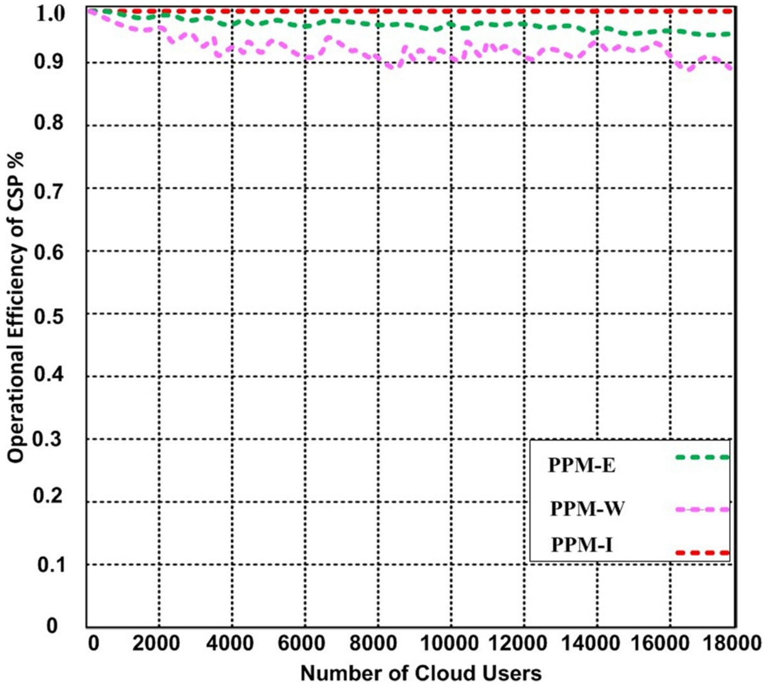

Fig. 10 Operational Efficiency VS number of cloud users (for a relatively larger cloud network)

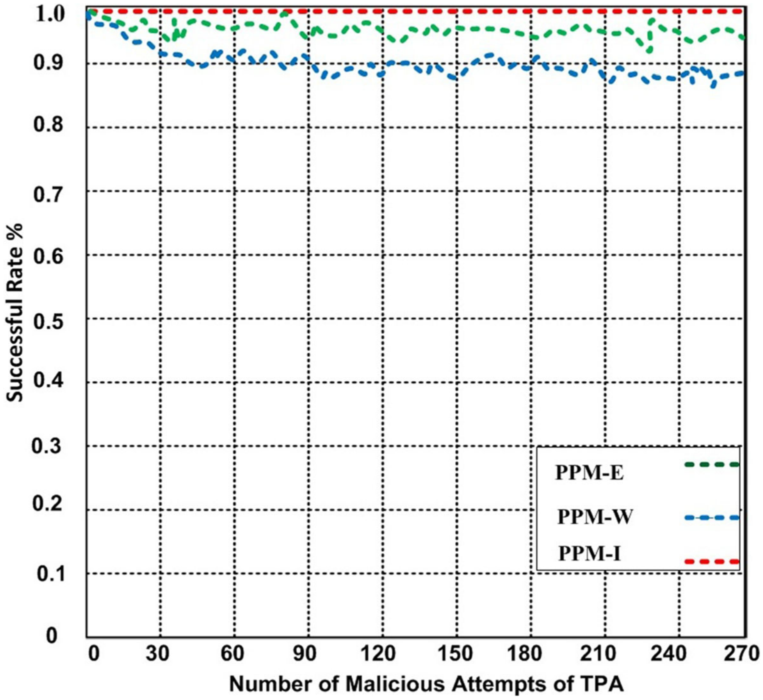

Fig. 11 Number of malicious attempts of TPA versus successful rate of detecting the malicious behavior

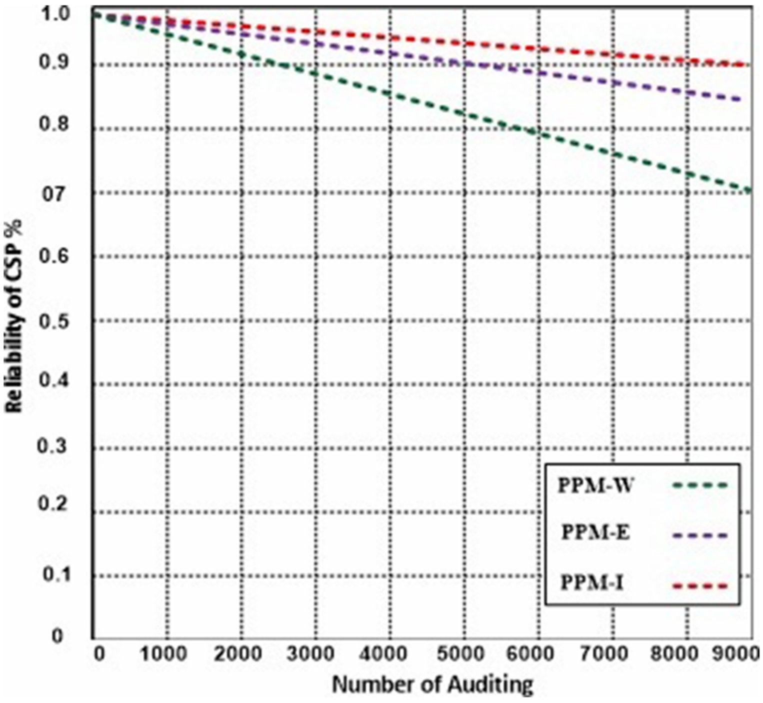

Fig. 12 Reliability of CSP on Number of Auditing

scenario (PPM-E), an approximately 85.2% reliability ratio was observed. In the third scenario, worse case (PPM-W), approximately 70% reliability of the CSP was noted. This simulated behavior implies that the PPM is capable of determining the reliability ratio of the CSP for all three cases. If a CSP is able to maintain an expected reliability ratio, the advantage of storing the confidential data with the CSP will greatly help in evolving the trust between CUs and service providers.

Related work

Most of the existing work [27–29] on the data privacy preservation is based on web services, which address the issues of data control and security in a cloud environment. This includes data access, data integrity, data recovery, data separation, data disposition, and data regulations [27]. Sengupta et al. [28] addressed the cloud security and privacy concerns such as data access, data compliance, and cloud hosted code. Rabai et al. [29] introduced a quantitative cloud security model which helps the subscribers and providers to measure the security risks related to the resources. The model assists the subscribers and providers to investigate and identify the security related issues. In addition, the model takes certain attributes into account when making the security decisions, such as economics, stakeholders, and heterogeneities. Yuhong et al. [30] proposed a new EnTrust framework which integrates the encryption and trust-based techniques to preserve the cloud's storage privacy. Specifically, the proposed framework contains three components: an encryption module, a trust evaluation module and a decision module.

Trusted third party protocols for securing the cloud computing environment were introduced in [31] and [32]. Zissis et al. [31] discussed the TPAs working principles and its security concerns. Thamizhselvan et al. [32] proposed a model that uses a third party security vendor that takes care of encryption and decryption of the data based on the CU's preferences. The public verifiability protocol without the use of a TPA was proposed in [33]. The protocol involves features that do not disclose any confidential information to the TPAs. The protocol demonstrated the accuracy and security parameters through formal analysis. Based on the experimental results, the authors claimed that the proposed protocol performed better in remote data cloud storage.

The trusted computing environment was proposed in [34], which involves the trusted computing module in the cloud computing environment. The trusted computing module focuses on the confidentiality, integrity, and the authentication. However, the module does not provide validation proof. A third party auditing protocol was introduced to keep online storage secure by encrypting the data before applying the hash functions using the symmetric-keys [25]. The auditor verifies the integrity of the received data and decrypts key at the server side. This proposed protocol experiences problems due to limited features that potentially exert an excessive overhead on the servers.

Rosa et al. [35] proposed a privacy-enhancement and trust-aware IdM mechanism which is based on the SAMLv2/ID-FF standards. The primary objective of this mechanism is to obtain an effective access control and identity management. A similar scheme was proposed by Siani et al. [36] to mitigate the data protection risks using a privacy management architecture. In their proposed architecture, the privacy manager is responsible to preserve the privacy for cloud computing technologies. Another privacy-preserving scheme was introduced in [37] for protecting the privacy of individuals in cloud computing environment. The authors attempt to induce the significance of privacy at all levels when designing, collecting, sharing, and processing the cloud's services. However, their research only focueses on the notion of privacy, but no proof was provided to handle the data privacy issues.

Cong et al. [38] proposed a privacy-preserving public auditing system for data storage security in cloud computing. Specifically, the proposed system introduced the homomorphic linear authenticator with the random masking to prevent a TPA in accessing the contents of the customer's outsourced data during the auditing process. The proposed system enables a TPA to perform multiple auditing tasks in a batch processing mode for improved efficiency.

Hassan et al. [39] highlighted the fast adoption deferring-reasons for cloud security. Addressing this issue, a comprehensive cloud security framework was introduced to resolve the deferring-reasons for cloud security. This framework involves the following modules: the policy integration module, access control, trust management, service management, authentication, heterogeneity management and identity management. Pelin et al. [40] proposed an authentication framework to detect the cloud computing entities such as cloud as user and services. The authors use the identity management module to address the privacy and identity issues. Recently, Abdul et al. [41] proposed a triangular data privacy-preserving scheme that supports public auditing with the capability of auditing all the key stakeholders for achieving optimal security in a cloud environment.

Conclusion

To guarantee data privacy in a cloud computing environment, it is essential to introduce a new scheme to authenticate the three cloud stakeholders (i.e., CSP, CU and TPA). Thus, the proposed triangle authentication process enables the three stakeholders to detect the negative role of each other. Another concern is how to design a privacy-preserving model to restrict the potential TPA vulnerabilities, control the malicious insider threats in CSPs, and determine the CUs deceitful role of distributing the obtained service to other clients. In this paper, we explored the integrity and privacy-related challenges among the three entities. To build a secure and efficient cloud computing environment, we extend and improve the existing CSP and TPA security models by leveraging the properties into a single triangular data privacy-preserving model to provide the auditing capability to all the key stakeholders. To support efficient and effective triangular auditing tasks, the scope of our privacy-preserving model is limited to: (a) guaranteeing the TPA's integrity, (b) administering the firm compliance of SLA by both the CSP and CUs, (c) authoring the exact use of allotted session keys for auditing the confidential data stored on the cloud's server, and (d) confirming the message authentication at the cloud service provider's side. The TPA audits the CSP to confirm the privacy of the CUs' outsourced data. The TPA also monitors the response provided by the CUs for the utilized services according to the SLA. Finally, an audit of the TPA is performed by both CSPs and CUs to reduce the probability of any possible malicious insider threats. To validate the correctness and soundness of the proposed work, an experimental analysis is conducted, which proves that the proposed PPM is highly efficient for preserving the data stored in the cloud computing environment.

Funding
This research project was not funded.

Authors' contributions
Both authors have equal contribution. Specifically, Dr. Razaque worked on Sections Proposed privacy preserving model and Experiment and performance evaluation. Dr. Rizvi worked on Sections Adversary model and Related work. Both authors read and approved the final manuscript.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Computer Science Department, New York Institute of Technology, New York, NY, USA. [2]Department of Information Sciences and Technology, Pennsylvania State University, Altoona, PA 16601, USA.

Received: 12 August 2016 Accepted: 7 February 2017
Published online: 27 March 2017

References

1. Rajkumar B, Yeo CH, Venugopal S, Broberg J, Brandic I (2009) Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Futur Gener Comput Syst 25(6):599–616
2. Peter M, Grance T (2011) The NIST definition of cloud computing
3. Sumant R, Eloff M, Smith E (2010) The management of security in cloud computing. In: IEEE Information Security for South Africa (ISSA)., pp 1–7
4. Yong C, Buyya R, Jiangchuan L (2014) Cloud computing. China Communications, 11(4) China Institute of Communications (CIC) and the IEEE Communications Society (IEEE ComSoc), USA.
5. Toosi AN, Calheiros RN, Buyya R (2014) Interconnected cloud computing environments: challenges, taxonomy, and survey. ACM Comput Surv 47(1):7:1–7:47
6. Qian W, Wang C, Li J, Ren K, Lou W (2009) Enabling public verifiability and data dynamics for storage security in cloud computing. In: Computer Security–ESORICS. Springer, Berlin Heidelberg, pp 355–370
7. Rampal S, Kumar S, Kumar SA (2012) Ensuring data storage security in cloud computing. International Journal of Engineering And Computer Science, 2319–7242
8. Cong W, Wang Q, Ren K, Cao N, Lou W (2012) Toward secure and dependable storage services in cloud computing. IEEE Trans Serv Comput 5(2):220–232
9. Francesc S, Domingo-Ferrer J, Martinez-Balleste A, Deswarte Y, Quisquater J-J (2008) Efficient remote data possession checking in critical information infrastructures. IEEE Trans Knowl Data Eng 20(8):1034–1038
10. Zhuo H, Yu N (2010) A multiple-replica remote data possession checking protocol with public verifiability. In: 2010 Second International Symposium on Data, Privacy and E-Commerce (ISDPE)., pp 84–89
11. Kun H, Xian M, Fu S, Liu J (2014) Securing the cloud storage audit service: defending against frame and collude attacks of third party auditor. IET Commun 8(12):2106–2113
12. Mehul SA, Swaminathan R, Baker M (2008) Privacy-preserving audit and extraction of digital contents. IACR Cryptology ePrint Archive, Report 2008/186, 2008, http://www.hpl.hp.com/techreports/2008/HPL-2008-32R1.pdf. Accessed 10 Aug 2016
13. Glenn B, Mogull R (2009) Security guidance for critical areas of focus in cloud computing. Cloud Security Alliance 2(1):1–76
14. Qingji Z, Xu S (2011) Fair and dynamic proofs of retrievability. In: Proceedings of the first ACM conference on Data and application security and privacy., pp 237–248
15. Ari J, Kaliski BS (2007) PORs: proofs of retrievability for large files. In: Proceedings of the 14th ACM Conference on Computer and Communications Security., pp 584–597
16. Yevgeniy D, Vadhan S, Wichs D (2009) Proofs of retrievability via hardness amplification. In: Theory of cryptography. Springer, Berlin Heidelberg, pp 109–127
17. Kevin BD, Juels A, Oprea A (2009) Proofs of retrievability: theory and implementation. In: Proceedings of the 2009 ACM Workshop on Cloud Computing Security., pp 43–54
18. Cong W, Ren K, Lou W, Li J (2010) Toward publicly auditable secure cloud data storage services. IEEE Netw 24(4):19–24
19. Mohammed SH, Al-Haidari F, Salah K (2011) Edos-shield-a two-steps mitigation technique against edos attacks in cloud computing. In: 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC)., pp 49–56
20. Hovav S, Waters B (2008) Compact proofs of retrievability. In: Advances in Cryptology-ASIACRYPT. Springer, Berlin Heidelberg, pp 90–107
21. Cong W, Wang Q, Ren K, Lou W (2010) Privacy-preserving public auditing for data storage security in cloud computing. In: IEEE INFOCOM., pp 1–9
22. Jia X (2011) Auditing the auditor: secure delegation of auditing operation over cloud storage. IACR Cryptology ePrint Archive, https://eprint.iacr.org/2011/304.pdf. Accessed 10 Aug 2016
23. Giuseppe A, Burns R, Curtmola R, Herring J, Kissner L, Peterson Z, Song D (2007) Provable data possession at untrusted stores. In: Proceedings of the 14th ACM Conference on Computer and Communications Security., pp 598–609
24. Bhagyashri S, Gurav YB (2014) Privacy-preserving public auditing for secure cloud storage. IOSR Journal of Computer Engineering (IOSR-JCE) 16(4):33–38
25. Mehul SA, Baker M, Mogul J, Swaminathan R (2007) Auditing to keep online storage services honest. In: Galen H (ed) Proceedings of the 11th USENIX Workshop on Hot Topics in Operating Systems (HOTOS'07). USENIX Association, Berkeley, p 6, Article 11
26. El-Booz SA, Attiya G, El-Fishawy N (2016) A secure cloud storage system combining time-based one-time password and automatic blocker protocol. EURASIP J Inf Secur 2016:13. doi:10.1186/s13635-016-0037-0
27. David C, Fatema K (2012) A privacy preserving authorisation system for the cloud. J Comput Syst Sci 78(5):1359–1373
28. Sengupta S, Kaulgud V, Sharma V (2013) Cloud computing security – trends and research directions. In: 2011 IEEE World Congress on Services, Washington DC, 2011., pp 524–531
29. Rabai LBA, Jouini M, Aissa AB, Mili A (2013) A cybersecurity model in cloud computing environment. Journal of Kind Saud University – Computer and Information Sciences 25(1):63–75
30. Yuhong L, Ryoo J, Rizvi S (2014) Ensuring data confidentiality in cloud computing: an encryption and trust-based solution. In: Proceedings of 23rd

IEEE Wireless and Optical Communication Conference (WOCC), Newark, NJ, May 2014., pp 1–6

31. Zissis D, Lekkas D (2012) Addressing cloud computing security issue. Futur Gener Comput Syst 28(2012):583–592
32. Thamizhselvan M, Raghuraman R, Gershon S, Victer Paul P (2015) A novel security model for cloud using trusted third party encryption. In: 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore., pp 1–5
33. Zhuo H, Zhong S, Yu N (2011) A privacy-preserving remote data integrity checking protocol with data dynamics and public verifiability. IEEE Trans Knowl Data Eng 23(9):1432–1437
34. Shen Z, Tong Q (2012) The security of cloud computing system enabled by trusted computing technology. In: 2nd International Conference on Signal Processing Systems (ICSPS), Dalian, 2010., pp V2-11–V2-15
35. Rosa S, Almenares F, Arias P, Díaz-Sánchez D, Marín A (2012) Enhancing privacy and dynamic federation in IdM for consumer cloud computing. IEEE Trans Consum Electron 58(1):95–103
36. Siani P, Shen Y, Mowbray M (2009) A privacy manager for cloud computing. In: Cloud computing. Springer, Berlin Heidelberg, pp 90–106
37. Jian W, Zhao Y, Jiang S, Le J (2009) Providing privacy preserving in cloud computing. IEEE International Conference on Test and Measurement (ICTM'09) 2:213–216
38. Cong W, Chow SM, Wang Q, Ren K, Lou W (2013) Privacy-preserving public auditing for secure cloud storage. IEEE Trans Comput 62(2):362–375
39. Hassan T, Joshi BD, Ahn GJ (2010) Securecloud: towards a comprehensive security framework for cloud computing environments. In: 2010 IEEE 34th Annual Computer Software and Applications Conference Workshops (COMPSACW)., pp 393–398
40. Pelin A, Bhargava B, Ranchal R, Singh N, Linderman M, Othmane LB, Lilien L (2010) An entity-centric approach for privacy and identity management in cloud computing. In: 2010 29th IEEE Symposium on Reliable Distributed Systems., pp 177–183
41. Razaque A, Rizvi S (2016) Triangular data privacy-preserving model for authenticating all key stakeholders in a cloud environment. Comput Secur 62:328–347

Submit your manuscript to a SpringerOpen journal and benefit from:

► Convenient online submission
► Rigorous peer review
► Immediate publication on acceptance
► Open access: articles freely available online
► High visibility within the field
► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com