El-Booz et al. EURASIP Journal on Information Security (2016) 2016:13
DOI 10.1186/s13635-016-0037-0

EURASIP Journal on Information Security

RESEARCH Open Access

A secure cloud storage system combining time-based one-time password and automatic blocker protocol

Sheren A. El-Booz*, Gamal Attiya and Nawal El-Fishawy

Abstract

Cloud storages in cloud data centers can be used for enterprises and individuals to store and access their data remotely anywhere anytime without any additional burden. By data outsourcing, users can be relieved from the burden of local data storage and maintenance. However, the major problem of cloud data storage is security. Moreover, cloud users must be able to use the cloud storage just like the local storage, without worrying about the need to verify the data integrity and data consistency. Some researchers have been conducted with the aid of a third party auditor (TPA) to verify the data stored in the cloud and be sure that it is not tampered. However, the TPA is leased by the provider, and after a time, a cloud service provider may contract with the TPA to conceal the loss of data from the user to prevent the defamation. This paper presents a novel secure cloud storage system to ensure the protection of organizations' data from the cloud provider, the third party auditor, and some users who may use their old accounts to access the data stored on the cloud. The proposed system enhances the authentication level of security by using two authentication techniques; time-based one-time password (TOTP) for cloud users verification and automatic blocker protocol (ABP) to fully protect the system from unauthorized third party auditor. The experimental results demonstrate the effectiveness and efficiency of the proposed system when auditing shared data integrity.

Keywords: Cloud computing, Privacy preserving, Public auditability, Third party auditor (TPA), One-time password (OTP), Automatic blocker protocol (ABP)

1 Introduction

Cloud computing has been envisioned as the next-generation of distributed/utility computing [1]. It is defined as a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [2]. The National Institute of Standards and Technology (NIST) defines cloud computing by five essential characteristics, three service models, and four deployment models [2]. The essential characteristics are on-demand self-service, location-independent resource pooling, broad network access, rapid resource elasticity, and measured service. The main three service models are software as a service (SAAS), platform as a service (PAAS), and infrastructure as a service (IAAS). The deployment models include private cloud, public cloud, community cloud, and hybrid cloud.

Nowadays, cloud-computing paradigm can offer any conceivable form of services, such as computational resources for high performance computing applications, web services, social networking, and telecommunications services. In addition, cloud storage in data centers can be useful for users to store and access their data remotely anywhere anytime without any additional burden [3]. However, the major problem of cloud data storage is security. Therefore, cloud data centers should have some mechanisms able to specify storage correctness and integrity of data stored on a cloud.

* Correspondence: eng.sheren1975@gmail.com
Computer Science and Engineering Department, Faculty of Electronic Engineering, Menoufia University, Menouf, Egypt

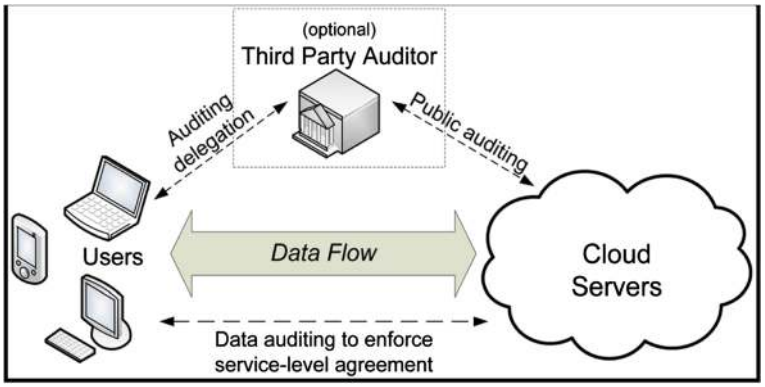

Fig. 1 Architecture of the cloud storage environment [2, 5]

Several methods have been put forward to tackle the issue of privacy preserving. Some researchers have been conducted with the aid of third party auditor (TPA) to verify the data stored in the cloud and be sure that it is not tampered [4–7]. The TPA can perform the auditing on behalf of a user and provide the audit report to the user. This technique is also useful for cloud service providers (CSP) to maintain its reputation by getting higher reliability, consistency, and data integrity ratings or certificates from TPA to improve their business on commercial point of view. However, the major problem that arises with this approach is that the TPA was leased by the provider, and after a time, the cloud service provider may contract with the TPA to conceal the loss of data from the user to prevent the defamation. As a result, the correctness of the data in the cloud storage is being put at risk.

This paper presents a novel secure cloud storage system to ensure high level of information confidentiality, availability, and integrity and to protect organizations' data from the cloud provider, the third party auditor, and some users who take advantage of their old accounts to access the data stored on the cloud. The proposed system increases the authentication level of security by using two authentication techniques; time-based one-time password (TOTP) for cloud users verification and automatic blocker protocol (ABP) to fully protect the system from unauthorized third party auditor. The experimental results demonstrate the effectiveness and efficiency of the proposed system when auditing shared data integrity.

The rest of this paper is organized as follows. Section 2 presents an overview of the related work. Section 3 introduces the general architecture of the privacy preserving public auditing system while Section 4 describes the TOTP algorithm. Section 5 describes the proposed system while Section 6 presents the system evaluation and implementation. Finally, Section 7 lists the concluding remarks.

2 Related work

The notion of public auditability has been proposed in the context of ensuring remotely stored data integrity

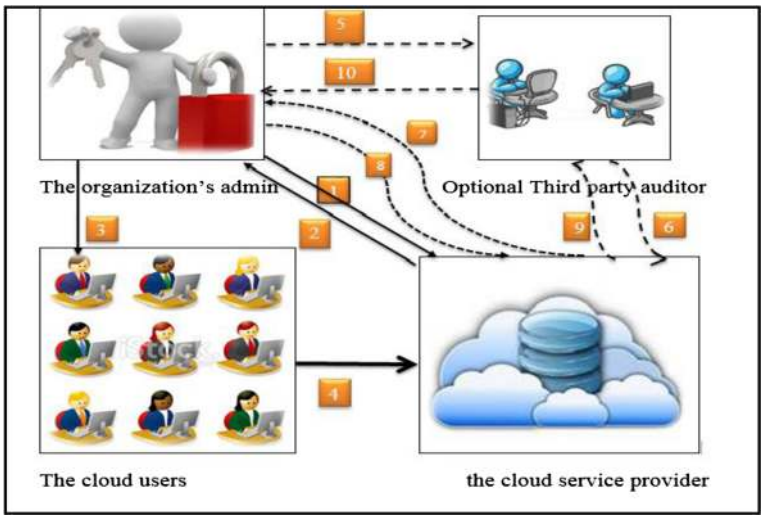

Fig. 2 Proposed system architecture

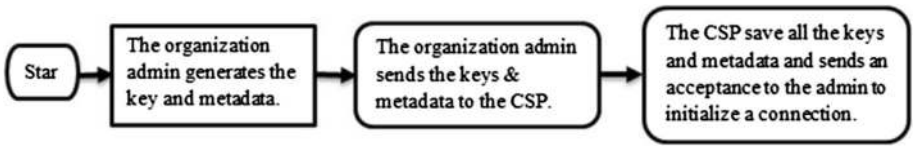

Fig. 3 Setup phase

under different system and security models. In [8], a public auditability model called provable data possession (PDP) is presented for ensuring possession of files on untrusted storages. The PDP model employees the RSA-based homomorphic authenticators for data auditing. By using the PDP model, public auditing is achieved, but that model only supports static data. In subsequent work, the authors in [9] present partially dynamic version of the PDP model. But, the system imposes a priori bound on the number of queries and does not support fully dynamic data operations, that is, it only allows very basic block operations with limited functionality, and block insertions cannot be supported. The PDP model presented in [8] is extended in [10] to support provable updates to stored data files using rank-based authenticated skip lists. The scheme is essentially a fully dynamic version of the PDP solution. To support updates, especially for block insertion, the extended PDP eliminates the index information in the tag computation in the PDP model and employ authenticated skip list data

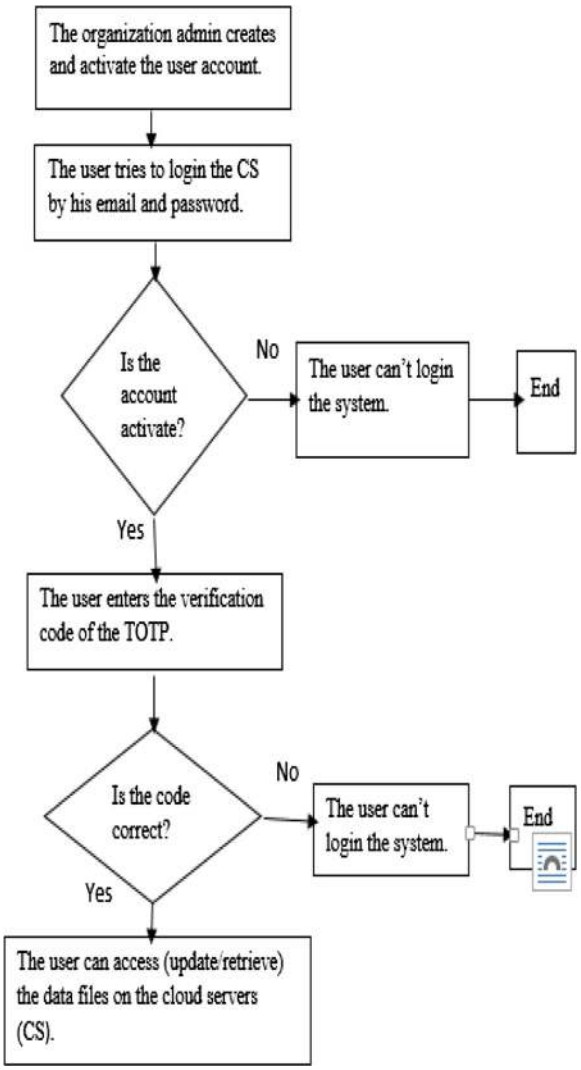

Fig. 4 Data access phase

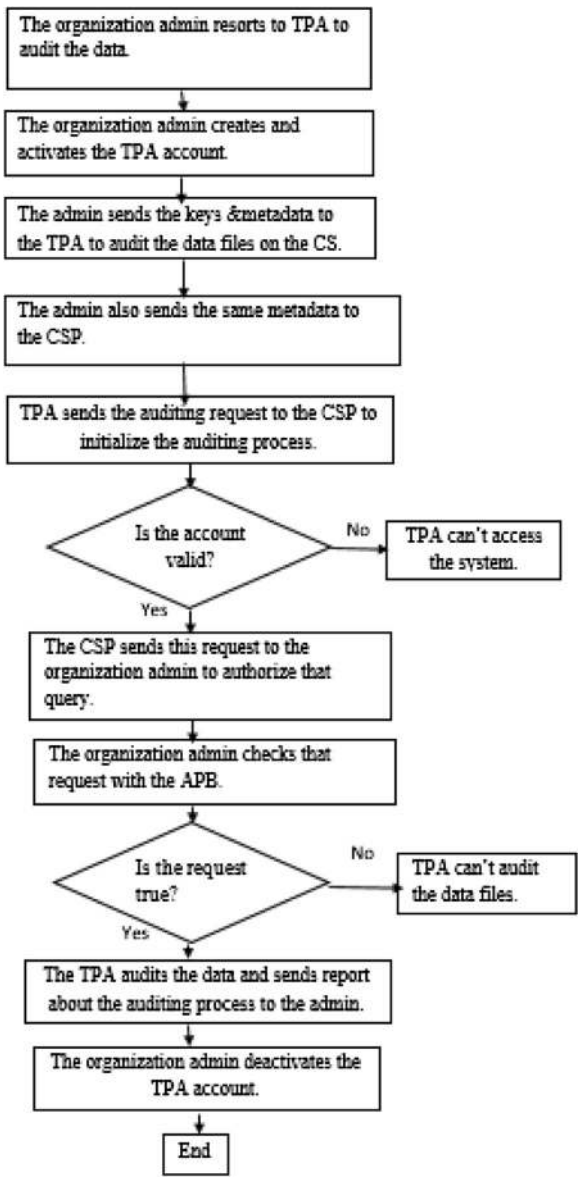

Fig. 5 Data audit phase

structure to authenticate the tag information of challenged or updated blocks first before the verification procedure. However, the efficiency of the extended PDP scheme remains unclear.

In [11], Wang et al. Consider dynamic data storage in a distributed scenario. They proposed a protocol able to determine both the data correctness and locate possible errors. But, like [9], the authors only consider partial support for dynamic data operation. So, in their subsequent work [12], they proposed a system based on a combination between BLS-based homomorphic authenticator and MHT. That combination would support public auditability for fully dynamic data. In [13], a new scheme called proof of retrievability (PoR) is proposed. In this scheme, both spot-checking and error correcting codes are used to ensure both possession and retrievability of data files on the cloud storage system. However, like [9], the number of queries a client can perform is also a fixed priori, and the introduction of precomputed sentinels prevents the development of realizing dynamic data updates. In [14], an improved PoR scheme is designed using publicly verifiable homomorphic authenticators built from BLS signatures. Full proofs of security in the security model defined in [13] are doing. The proofs can be aggregated into a small authenticator value, and public retrievability is achieved. Although the enhancement achieves the purpose, still, the authors only consider static data files. In [15], the authors

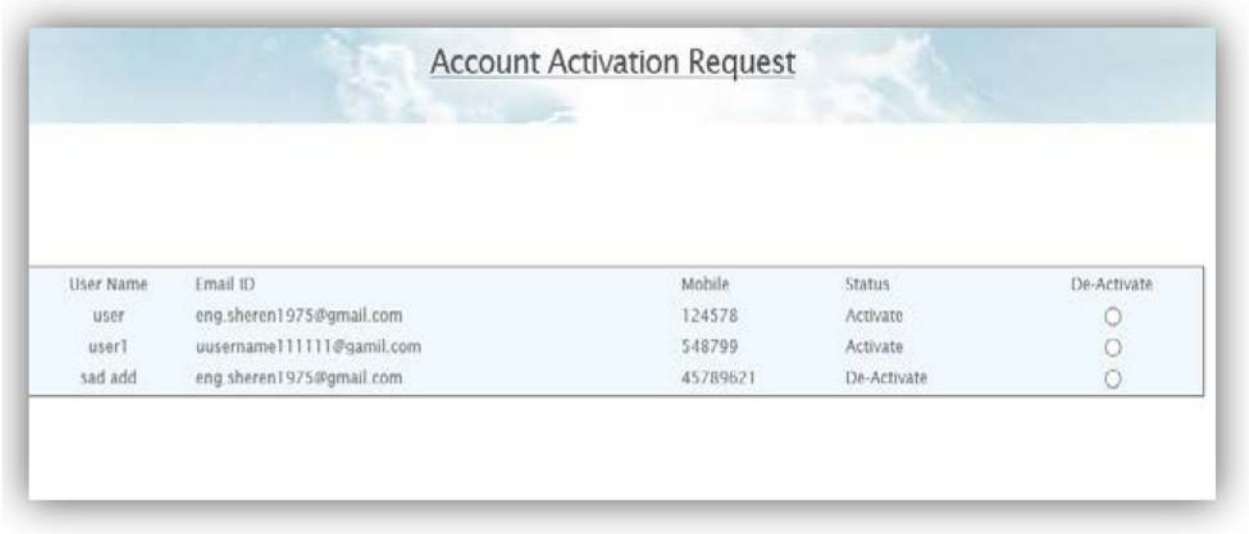

Fig. 6 Activation of users accounts by the admin

introduce the concept of TPA to reduce online burden and keeps data integrity and privacy preserve. An improved technique of verifying data integrity on cloud by utilizing the concept of TPA is introduced in [16]. In [17], the authors approved that involving the TPA may associate additional risk to the confidentiality of data.

3 Privacy preserving public auditing

As shown in Fig. 1, the general architecture of the cloud storage environment consists of three entities [6, 7].

- Cloud server (CS): an entity has significant storage space and computation resources. The cloud server is managed by cloud service provider (CSP) to provide data storage service to anyone wants to store data in the cloud.
- Cloud user (CU): a person who has huge amount of data files to be stored in the cloud server.
- Third party auditor (TPA): the one who has expertise and capabilities that users may not have and is trusted to assess and expose risk of cloud storage services on behalf of the users upon request.

In cloud data storage, a user stores his data through a CSP into a set of cloud servers, which are running in a simultaneous, cooperated, and distributed manner. The user relies on the CS for cloud data storage and maintenance. Thereafter, for various application purposes, the user may dynamically interact with the CS via CSP to access and retrieve/update the stored data. As the user no longer possesses his data locally, it is of critical importance for the user to ensure that his data are being correctly stored and maintained, that is, the user should be equipped with security means so that he can make continuous correctness assurance (to enforce cloud storage service-level agreement) of his stored data even without

Fig. 7 Access verification by the TOTP

Fig. 8 The TPA requests permission by the CSP

the existence of local copies. To verify the correctness and integrity of data stored in the cloud servers, auditing process should be done. The user may delegate the data auditing tasks to an optional trusted TPA of their respective choices for ensuring the storage security of the outsourced data, while hoping to keep his data private from the TPA. The privacy preserving public auditing scheme proceeds as follows:

1. The cloud users initialize the connection by generating the keys and metadata
2. Sending these keys to the CSP
3. Sending the key and metadata to the TPA to audit the outsourced data on the CS

If a cloud user wants to upload data to the cloud storage, the user waits until the CSP sends a permission to start uploading the data. On the other hand, if the user wants to check the correctness of data on the cloud, the user resorts to the TPA who has expertise to audit the data upon request from the user. The TPA sends a query to the CSP to audit data, and then the CSP responds to the TPA to permit auditing data [6, 7]. The TPA will do the auditing for storage correctness and integrity of data. Here, the TPA must efficiently audit data without bringing any changes to the original data. In addition, any possible leakage of user's outsourced data towards TPA through the auditing protocol should be prohibited.

Although the privacy preserving public auditing system achieves a good level of data security, the problem arises with such a system is that the cloud service provider may contract with the third party auditor to conceal the loss of data from the user to prevent the defamation. In this case, the user might lose his data because he did not know anything about the deal between the TPA and the CSP. In fact, this

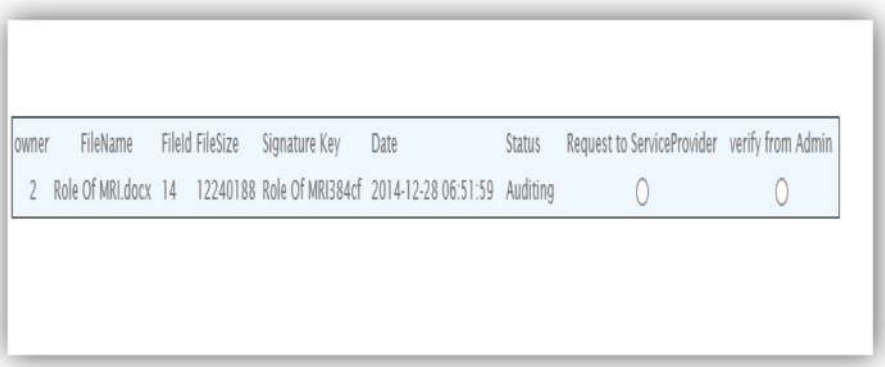

Fig. 9 The APB authenticates the TPA request

Table 1 Test results

Penetrate test	Recent system [22]	Result	Test period/days	Proposed system	Results	Test period/days
Test 1: SQL injection	No SQL injects	Intact	3 h	No SQL injects	Intact	1 day
Test 2: Broken authentication	The authentication is broken due to the usage of the static password	Not intact	2 days	The authentication not broken due to usage TOTP at users' side and APB at TPA side	Intact	3, 5 days
Test 3: Sensitive data exposure	The outsourced data encrypted using DES technique	Not intact	3 days	The outsourced data encrypted using AES technique	Intact	3, 5 days
Test 4: CSRF	The sessions managed using HTTP	Not intact	1 day	The sessions managed using HTTPS	Intact	3, 5 days
Test 5: Invalidated redirects and forwards	The spider found that some token is not assured	Not intact	day	The tokens between the users and CSP is assured	Intact	3, 5 days

problem appears recently and most of the data owners suffer from it.

4 Time-based one-time password algorithm

TOTP algorithm is an algorithm that computes a one-time password from a shared secret key and the current time. It has been adopted as Internet Engineering Task Force standard RFC 6238 [18]. The TOTP combines a secret key with the current timestamp using a cryptographic hash function to generate a one-time password. In a typical two-factor authentication application, user authentication proceeds as follows: a user will enter username and password into a website or other server, generate a one-time password for the server using TOTP running locally on a smartphone or other device, and type that password into the server as well. The server will then verify the entered one-time password. Both the server and the data owner compute the token, then the server checks if the token supplied by the data owner matches the locally generated token. Then the session between them is opened and securely the users can access the system.

According to RFC 6238 [18], the TOTP is based on HOTP with time stamp replacing the incrementing counter. The reference implementation of the HOTP algorithm is as follows:

1 K is a secret key.
2 C is a counter.
3 HMAC (K, C) = SHA1 (K ⊕ 0x5c5c … ‖ SHA1 (K ⊕ 0x3636 … ‖ C)) is a HMAC calculated with SHA cryptographic technique.
4 HOTP (K, C) = Truncate (HMAC (K, C)) & 0x7FFFFFFF.

The current time stamp has turned into an integer time-counter (TC) that depends on two parameters; the start of an epoch (T0) and the time step (TS). TC calculated as:

$$\mathrm{TC} = (\text{time now} - \text{time}(\mathrm{TO}))/\mathrm{TS}. \quad (1)$$

The TOTP is computed as follows:

$$\begin{aligned} \mathrm{TOTP} &= \mathrm{HOTP}(\text{secretkey}(\mathrm{K}), \mathrm{TC}) \\ \text{TOTP value} &= \mathrm{TOTP} \mod 10^{d} \end{aligned} \quad (2)$$

Where, d is the desired number of digits of the one-time password, according to RFC6238 [18] reference implementation.

5 Proposed system

The proposed system improves the authentication level of security by using two authentication techniques; TOTP [18] to authenticate the users and ABP [19] to authenticate the TPA. Figure 2 shows the general framework/architecture of the proposed system. The proposed system consists of four entities:

1. Organization admin: an entity, who has huge amount of data to be stored in the cloud, can be either enterprise or individual customers. The admin has all the privileges over the users and the third party auditors.
2. Cloud user (CU): the user that can access (update or retrieve) the data on the cloud under supervision of the organization admin.
3. Third party auditor (TPA): the one who may rent upon the request from the admin to audit the data stored on the cloud.
4. Cloud service provider (CSP): the one who can manage the cloud servers that have a large storage space available for any organization wants to store their data.

5.1 Methodology of the proposed system

The entities in the proposed system conduct according to the following sequence:

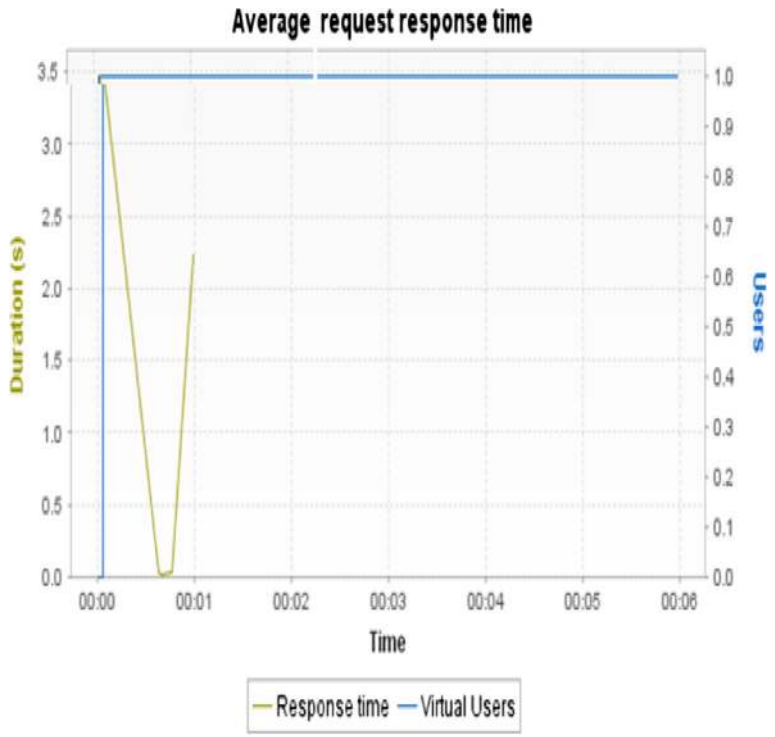

Fig. 10 Average request time for 10 users

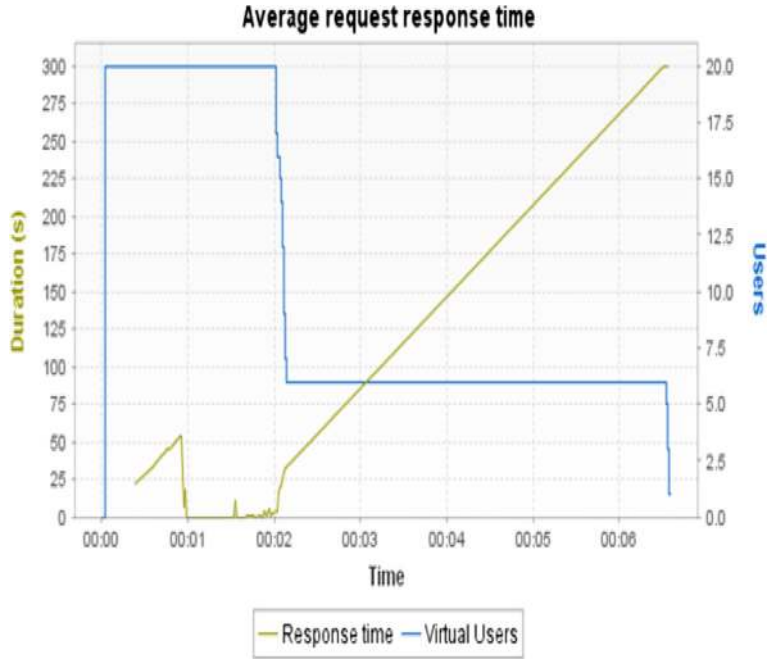

Fig. 11 Average request time for 20 users

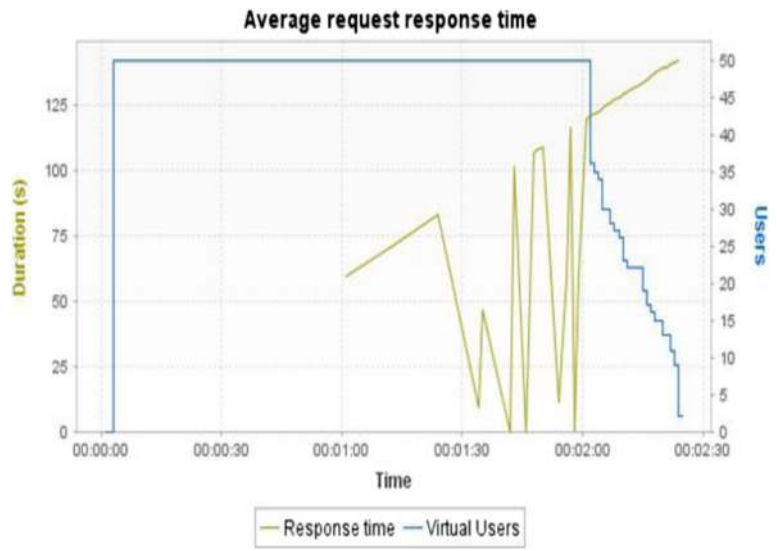

Fig. 12 Average request time for 50 users

1 The organization admin first initializes the setup scheme using KEYGEN [11, 12] algorithm to generate the keys and metadata and then sends them to the CSP.
2 The CSP replies to the request from the admin, by using the SINGEN [11, 12] algorithm, to accept the set up scheme. So, a connection initializes between the admin and the CSP. However, before outsourcing data to the CS, data is encrypted by a powerful encryption technique called advanced encryption system (AES).
3 To achieve the information confidentiality, integrity, and availability, according to the CIA triad for the information security, the user must

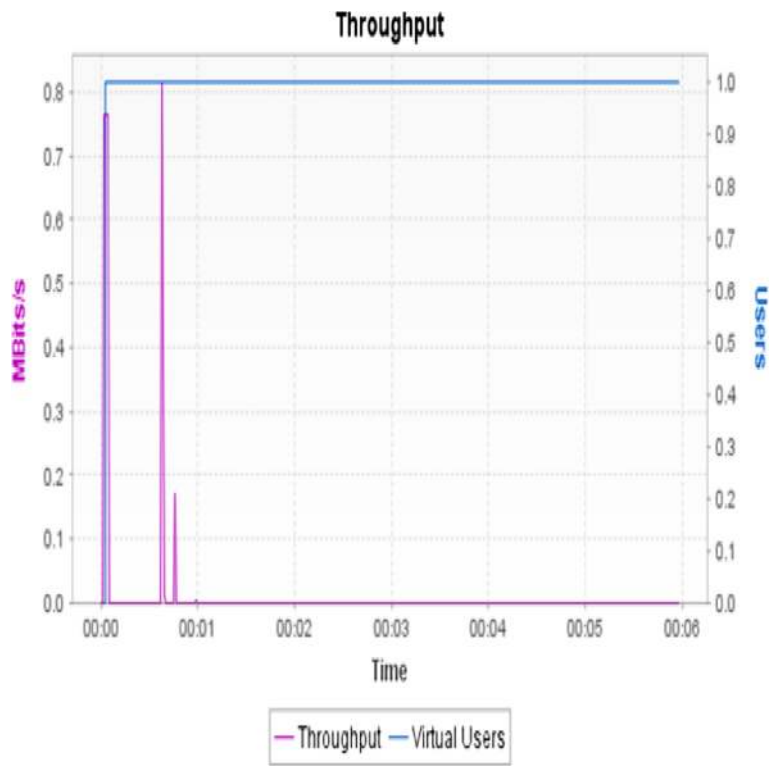

Fig. 13 System throughput for 10 users

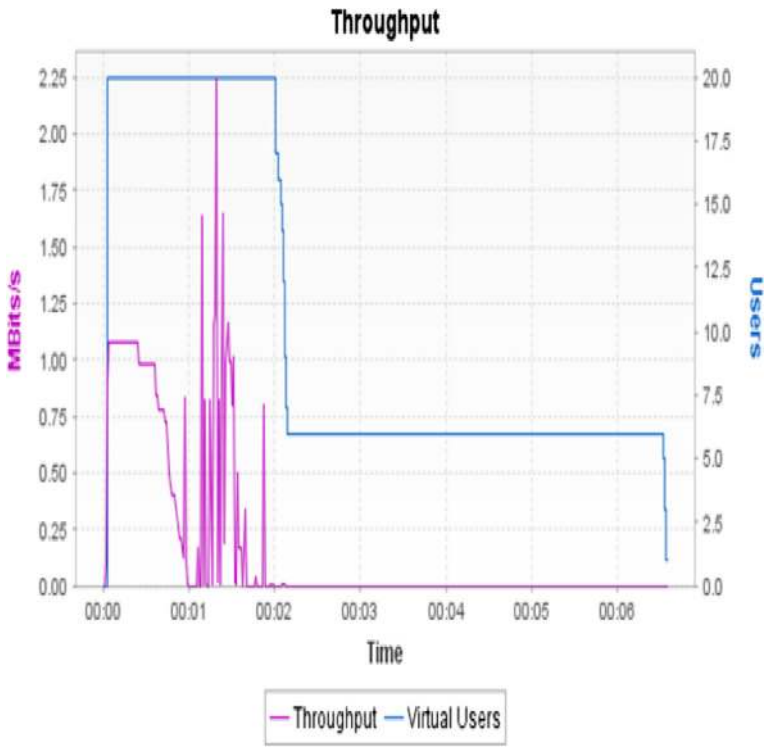

Fig. 14 System throughput for 20 users

have an account (emails and password) to access stored data. In this system, more restrictions upon these accounts are done by the admin to avoid data access by pre-activated accounts, where, the admin is the only one that can activate or not the accounts.

4 The activated users' accounts can login by using the two stages authentication technique; username with password and the TOTP that is permitted for one session between the user and the cloud server.
5 If the organization admin wants to audit the outsourced data on the cloud server, he resorts to

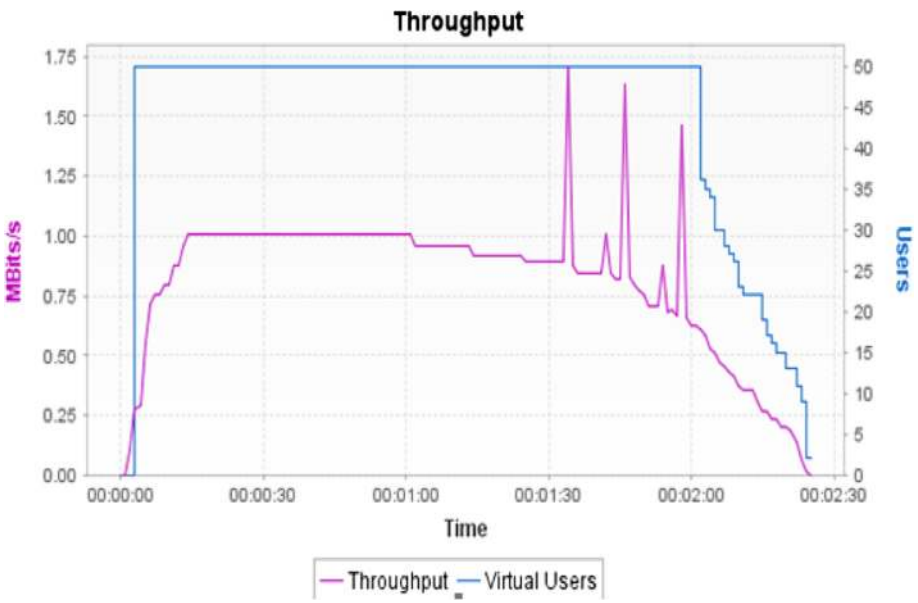

Fig. 15 System throughput for 50 users

the TPA who has the expertise to audit the data. However, the TPA must have an account in the system. This account must also activate from the organization admin. If the TPA account is activated from the organization admin, then the secret key and metadata would send to the TPA to audit the outsourced data on the CS; otherwise, the TPA cannot access the system.

6 TPA with the secret key and metadata sends the auditing request to the CSP to initialize the auditing process.

7 The CSP sends a query about the auditing process to the organization admin to authorize that query and TPA metadata by using the APB.

8 If the APB is true, the admin sends the approval to the cloud service provider with the metadata; otherwise, the TPA cannot access the system.

9 The TPA has a report about the data auditing.

10 The TPA sends the report about the auditing process to the organization admin with the metadata, then the admin deactivates the TPA account, to prevent any deal may be happen between the CSP and the TPA to hide the data losses from the admin.

Briefly, the proposed public auditing system is constructed in three phases: setup, data access and audit.

Setup phase: The organization admin initializes the public and secret parameters of the system by executing KEYGEN algorithm and preprocesses the data by using SigGen to generate the verification metadata (the information about the connection, the users' accounts). The admin then stores the data at the cloud server, deletes its local copy, and publishes the verification metadata to TPA for later audit. As part of pre-processing, the user may alter the data file by expanding it or including additional metadata to be stored at server. Figure 3 shows the sequence of the setup phase.

Data access phase: The user that has a valid account activated by the admin is the only one that can login to the CS. However, to access stored data, the user must enter a verification code (TOTP) to get permission for one session between the user and the cloud server. Figure 4 shows the sequence of the data access phase.

Data audit phase: The organization admin resorts to the TPA who has the expertise to audit the data. This is done by activating the TPA account and sending the secret key and metadata (the information about the data files but not the data files itself) to the TPA to audit the outsourced data on the CS. The TPA issues an audit message to the cloud server to make sure that the cloud server has retained the data file properly at the time of the audit. The cloud server will derive a response message from a function of the stored data file F by executing GenProof [19]. Using the verification metadata, the TPA verifies the response via verify proof [19]. The TPA sends the report about the auditing process to the organization admin, then the admin deactivates the TPA account, to prevent any deal may be happen between the CSP and the TPA. Figure 5 shows the sequence of the audit phase.

The proposed system can serve any organization (i.e., trading companies and banking, commercial companies) to store their data on the cloud storage that set with the providers. The organization can rent a TPA for auditing process to prevent the contract between the TPA and the CSP for hiding data losses. The auditing process may be done any times upon the organization need. In addition, the proposed system can serve the organization to authenticate the users to access their system many times upon their registration in the system.

6 System implementation and evaluation

A. System implementation:
The proposed system is implemented by using the Java enterprise edition web application with tomcat server. In the proposed system, every organization has an admin to guarantee the data confidentiality, integrity, and availability. The admin generates the keys and metadata, initializes connection with the CS, and then stores the data at the cloud server. However, before outsourcing data to the CS, data is encrypted by using the advanced encryption system (AES). To access the data on the CS, the admin activates the users' accounts, as shown in Fig. 6. Then, the TOTP authenticates the users with the approval from the cloud service provider, as shown in Fig. 7. To audit the outsourced data, the admin delegates the TPA for the auditing process and uses the ABP to permit the activated TPA to audit the outsourced data, as shown in Fig. 8 and Fig. 9. If the TPA is permitted by the APB, the admin sends the metadata with the secret key to the TPA for auditing process. After auditing, the TPA sends the auditing report to the admin, then the admin deactivates the TPA account to ensure that the TPA cannot login the system again. These restrictions upon TPA enhance the secrecy of the system.

B. System evaluation:
Since the proposed system is developed to enhance the level of information confidentiality, availability, and integrity, therefore, to evaluate the proposed system, it should be tested against hacking the user credential ties, and the TPA auditing credential ties, and hacking the data transmitted between the cloud entities.
The proposed system is evaluated by using the penetration test program of the Open Web Application Security Project (OWASP) [20, 21]. The

OWASP is an online community dedicated to web application security. This program has 10 tests used to examine the security levels of any web application classifies the vulnerabilities on the web application. The OWASP top 10 tests are named as follows; 1—injection, 2—broken authentication and session management, 3—cross-site scripting (XSS), 4—insecure direct object references, 5—security misconfiguration, 6—sensitive data exposure, 7—missing function level access control, 8—cross-site request forgery (CSRF), 9—using components with known vulnerabilities, and 10—unvalidated redirects and forwards.

In this evaluation, the most five tests that runs upon the most recent privacy preserving systems are used to evaluate the proposed system. These tests include:

Test 1: SQL injection
All the interpreters in the system are examined, including external users, internal users, and administrators.
Test 2: Broken authentication and session management
Anonymous attackers attempt to steal the users' accounts or spoofing their passwords; this might be handled in our system using TOTP for users' login that is available for one session for a while between the user and CSP and APB used to authenticate the TPA.
Test 3: Sensitive data exposure
The one who can access our data is very important and dangerous, so we assume that the admin is the only one who allows the other interpreters in the system to access the data, and the data must be encrypted using AES encryption technique before it outsourced to CSP.
Test 4: Cross-site request forgery (CSRF)
If there was a request for our interpreters to steal their credential ties, so the interpreters like users must follow some more precautions after they log in to the system. We solve that problem using TOTP that is OTP which encrypted using SHA technique which considered as good security technique.
Test 5: invalidated redirects and forwards
Examines the web application against HTTP sessions, which examines if there were some vulnerability in our codes and how we manage sessions, if these sessions redirected in a true manner or not to assure the tokens between the interpreters and the server.

Table 1 summarizes the results of different tests. It presents a comparative study between results obtained by applying the proposed system and that obtained by applying the most recent existing system [22]. From the table, the test results of the recent privacy preserving system indicate that these systems have some vulnerability. However, when examining the proposed system, the novel secure system increases the data confidentiality and availability. Note that, in the proposed system, the main concern is the efficiency and effectiveness of the security technique regardless of the storage space or the time delay.

The second evaluation is done by testing the average request response time of the proposed system when different number of users login the system (i.e., 10 users, 20 users, 50 users) taking into account the users request time. Figures 10, 11, and 12 show the average request response time under 10, 20, and 50 users, respectively.

The third evaluation is done by testing the throughput (the number of megabites of data per second) of the proposed system when different number of users login the system (i.e., 10 users, 20 users, 50 users) taking into account the users request time. Figures 13, 14 ,and 15 show the system throughput of returned data in megabites by 10, 20, and 50 users, respectively.

As shown in Figures 10, 11, and 12, the average request response time increases for every increase in number of users requests. In addition, the proposed system performance or throughput (the number of megabits of data per second returned by the server) increases for every increase in the users numbers, as shown in Figures 13, 14, and 15.

7 Conclusions

This paper tackled the privacy preserving public auditing. A novel secure cloud storage system is proposed to ensure the protection of organizations' data from both the cloud provider and the third party auditor and from some users who take advantage of the old accounts to access the data stored on the cloud. The proposed system increases the authentication level of security by using two authentication techniques; time-based one-time password (TOTP) and automatic blocker protocol (ABP). In the proposed system, the data owner controls all the privileges to be sure that who can access the outsourced data on cloud storage servers. To increase security, user authentication is verified by two-factor authentication: the first is exercised with a username and password while the second is caused by the implementation of TOTP. The experimental results demonstrate the effectiveness and efficiency of the proposed system when auditing shared data integrity.

Authors' contributions

SE and GA carried out the main research of this work. NE participated in the design and methodology of the proposed system architecture. SE implemented the proposed system while GA and NE approved the system implementation. SE performed the experiments and performed the statistical analysis. SE has drafted the manuscript while GA and NE read and modified the manuscript. In addition, all the authors revised and approved the final manuscript.

Acknowledgements

The authors extend their thanks to the reviewers for the comments and suggestions that helped in improving the overall quality of this article.

Received: 25 January 2016 Accepted: 25 May 2016
Published online: 11 June 2016

References

1. SUCIU George, HALUNGA Simona, APOSTU Anca, VULPE Alexandru, TODORAN Gyorgy, Cloud computing as evolution of distributed computing—a case study for SlapOS distributed cloud computing platform. Informatica Economică **17**(4), 109–122 (2013)
2. P Mell, T Grance, *"The NIST Definition of Cloud Computing," National Institute of Standards and Technology, Information Technology Laboratory, October 7*, 2009. http://www.nist.gov/itl/cloud/
3. MA Sharkh, M Jammal, A Shami, A Ouda, Resource allocation in a network-based cloud computing environment: design challenges. IEEE Communications Magazine **51**(11), 46–52 (2013)
4. C Wang, Q Wang, K Ren, W Lou, Privacy-preserving public auditing for secure cloud storage. IEEE Transactions on Computers **62**(2), 1–12 (2013)
5. M. Venkatesh, M. R. Sumalatha and C. SelvaKumar, "Improving public auditability, data possession in data storage security for cloud computing," Proc. of the International Conference on Recent Trends in Information Technology (ICRTIT), pp. 463-467, 19-21 April 2012.
6. S Bhagyashri, YB Gurave, A survey on privacy preserving techniques for secure cloud storage. International Journal of Computer Science and Mobile Computing (IJCSMC) **3**(2), 675–680 (2014)
7. T Paigude, TA Chavan, A survey on privacy preserving public auditing for data storage security. International Journal of Computer Trends and Technology (IJCTT) **4**(3), 412–418 (2013)
8. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Paterson, and D. Song, "Provable data possession at untrusted stores", Proc. of the ACM Conference on Computer and Communications Security (CCS'07), pp. 598-610, October 29–November 2, 2007.
9. G. Ateniese, R. D. Pietro, L. v. Mancini, and G. Tsudik, "Scalable and efficient provable data possession", Proceedings of the 4th International Conference on Security and Privacy in Communication Networks, SecureComm, pp. 1–10, 2008.
10. C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession", Proc. of the 16th ACM conference on Computer and communications security (CCS), pp. 213-222, 2009.
11. C Wang, Q Wang, K Ren, W Lou, Privacy preserving public auditing for secure cloud storage. IEEE Transactions on Computers **62**(2), 362–375 (2011)
12. C Wang, Q Wang, K Ren, W Lou, Towards secure and dependable storage services in cloud computing. IEEE Trans. on Services Computing **5**(2), 220–232 (2012)
13. A. Juels, J. Burton, and S. Kaliski, "Proofs of retrievaliability for large files", Proceedings of the 14th ACM Conference on Computer and Communications Security (ccs), pp. 584-597, 2007.
14. H Shecham, B Wates, Compact proofs of retrievability. Advances in Cryptology-ASIACRYPT **5350**, 90–107 (2008)
15. M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan, "Auditing to keep online storage services honest," Proceedings of the 11th workshop on hot topics in operating systems (HotOS'07), 'HotOS', USENIX Association, pp. 1-6, 2007.
16. Q. Wang, C. Wang, J. Li, K. Ren and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing", Proc. 14th European Symp. Research in Computer Security (ESORICS '09), pp. 355-370, 2009.
17. P. Prasadreddy, T. Srinivasa and S.Phani, "A threat free architecture for privacy assurance in cloud computing" Proceedings of the IEEE World Congress on Services, pp. 564-568, Jul.4-9, 2011, USA. IEEE Xplore Press, DOI:10.1109/SERVICES.2011.11.
18. D. M'Raihi, S. Machani, M. Pei, J. Rydell, "TOTP: time-based one-time password algorithm", Request for Comments (RFC) 6238, July 13, 2011.
19. K. Kiran, K. Padmaj, and P. Radha, "Automatic protocol blocker for privacy-preserving public auditing in cloud computing", IJCST, Vol. 3, Issue. 1, Jan –March, pp. 33-36, 2012.
20. www.owasp.org/index.php/Top_10. Accessed on April 2015.
21. OWASP TESTING GUIDE, V4.0, www.owasp.org. Accessed on April 2015.
22. S Bhagyashri, YB Gurav, Privacy-preserving public auditing for secure cloud storage. IOSR Journal of Computer Engineering (IOSR-JCE) **16**(4), 33–38 (2014)