DATA STORAGE SECURITY USING AUDITING SCHEME

S.Harini[1], K.Pushpavathi[2]

[1]*Dept of Computer Science, School of Arts and Science, Vinayaka Mission's Research Foundation, Chennai, India.*
Email : harinishalu5@gmail.com

[2]*Associate Professor, Dept of Computer Science, School of Arts and Science, Vinayaka Mission's Research Foundation, Chennai, India.*
Email: pushpavathi@avit.ac.in

ABSTRACT

With cloud storage services, users can remotely store their data to the cloud and realize the data sharing with others. Remote data integrity auditing is proposed to guarantee the integrity of the data stored in the cloud. In some common cloud storage systems such as the Electronic Health Records (EHRs) system, the cloud file might contain some sensitive information. The sensitive information should not be exposed to others when the cloud file is shared. Encrypting the whole shared file can realize the sensitive information hiding, but will make this shared file unable to be used by others. How to realize data sharing with sensitive information hiding in remote data integrity auditing still has not been explored up to now. In order to address this problem, we propose a remote data integrity auditing scheme that realizes data sharing with sensitive information hiding in this paper. In this scheme, a sanitizer is used to sanitize the data blocks corresponding to the sensitive information of them file and transforms these data blocks signatures into valid ones for the sanitized file. These signatures are used to verify the integrity of the sanitized file in the phase of integrity auditing. As a result, our scheme makes the file stored in the cloud able to be shared and used by others on the condition that the sensitive information is hidden, while the remote data integrity auditing is still able to be efficiently executed. Meanwhile, the proposed scheme is based on identity-based cryptography, which simplifies the complicated certificate management. The security analysis and the performance evaluation show that the proposed scheme is secure and efficient.

Keywords: Data storage, privacy preserving, public auditability, cloud computing, delegation, batch verification, zero knowledge

1. INTRODUCTION

Cloud computing has been envisioned as the next generation information technology (IT) architecture for enterprises, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk [2]. As a disruptive technology with profound implications, cloud computing is transforming the very nature of how businesses use information technology. One fundamental aspect of this paradigm shifting is that data are being centralized or outsourced to the cloud. From users' perspective, including both individuals and IT enterprises, storing data remotely to the cloud in a flexible on-demand manner brings appealing benefits: relief of the burden for storage management, universal data access with location independence, and avoidance of capital expenditure on hardware, software, and personnel maintenances, etc., [3]. While cloud computing makes these advantages more appealing than ever, it also brings new and challenging security threats toward users' outsourced data. Since cloud service providers (CSP) are separate administrative entities, data outsourcing is actually relinquishing user's ultimate control over the fate of their data. As a result, the correctness of the data in the cloud is being put at risk due to the following reasons. First of all, although the infrastructures under the

cloud are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity [4]. Examples of outages and security breaches of noteworthy cloud services appear from time to time [5], [6], [7]. Second, there do exist various motivations for CSP to behave unfaithfully toward the cloud users regarding their outsourced data status. For examples, CSP might reclaim storage for monetary reasons by discarding data that have not been or are rarely accessed, or even hide data loss incidents to maintain a reputation [8], [9], [10]. In short, although outsourcing data to the cloud is economically attractive for long-term large-scale storage, it does not immediately offer any guarantee on data integrity and availability.

2.RELATED WORKS

To verify the integrity of the outsourced data, several cloud storage auditing schemes have been proposed one after another. Ateniese et al.[8] proposed provable data possession, which uses a random sampling strategy and homomorphic authenticator. Juels and Kaliski [9] proposed proof of retrievability (PoR), which supports integrity auditing and recovery of the outsourced data. Subsequently, Shacham and Waters [13] proposed a compact PoR based on BLS signature [14], which can support public integrity auditing. Furthermore, for cloud storage auditing, the security of the key is becoming increasingly important. Therefore, such as key-exposure resilience [15]–[18] and key escrow [19]–[21] have been proposed successively in the cloud storage auditing. After that, some cloud storage auditing schemes with privacypreserving have been proposed. Wang et al. [22] proposed a privacy-preserving public cloud storage auditing scheme, which can prevent the third-party auditor (TPA) from obtaining private data. Shen et al. [23] proposed a lightweight cloud storage auditing scheme based on third party medium, which assists the DO to generate authenticators while protecting data privacy. Subsequently, Zhao et al. [24] proposed a privacy-preserving cloud storage auditing scheme, which uses a security out-sourcing algorithm to assist the DO to generate authenticators. Anbuchelian et al. [25] proposed a privacy-preserving cloud storage auditing scheme based on a secure encryption hash algorithm, which uses this hash algorithm to split and encrypt data. Han et al. [26] proposed a lightweight privacy-preserving cloud storage auditing, which does not need bilinear pairing operations in the auditing phase. The above-mentioned schemes have complex certificate management problems because they are based on public key infrastructure. Then, Yu et al. [27] proposed an identity-based cloud storage auditing scheme, which can prevent the auditor from accessing the DO's private data. The data sharing is one important service provided by cloud storage. Wang et al. [28] proposed a cloud storage auditing scheme for data sharing. In this scheme, the DO's identity privacy can be protected through a ring signature. However, [28] cannot track the DO's real identity. Subsequently, Yang et al. [29] proposed a cloud storage auditing scheme for data sharing, which can track the DO's identity. Fu et al. [30] proposed a cloud storage auditing scheme, which used a homomorphic verifiable group signature to share data. Subsequently, other cloud storage auditing schemes for data sharing based on group signatures were successively proposed [31]–[34]. Wu et al. [35] proposed an efficient threshold privacy-preserving cloud storage auditing scheme. This scheme does not rely on group signatures or ring signatures, so the tag generation efficiency is more efficient. In the cloud storage auditing scheme of data sharing, the issue of user revocation has always been the focus of research. In 2018,

Zhang et al. [36] proposed a cloud storage auditing scheme for data sharing, which reduces the cost of revoking data users. Then, Chang and Wu [37] proposed an efficient user revocation scheme with oblivious transfer and stateless lazy reencryption. However, the DO's sensitive information can be accessed in the above-mentioned cloud storage auditing schemes for data sharing. In 2018, Shen et al. [11] proposed a cloud storage auditing scheme for data sharing based on sanitizable signature [12], which can support the hiding of the DO's sensitive information. However, any users can access the sharing data in the scheme [11], which may cause the data abuse. Also, this scheme needs a secure channel between the DO and the sanitizer

3.EXISTING SYSTEM

Existing access controls in cloud are centralized in nature. All other schemes use attribute based encryption (ABE). The scheme uses a symmetric key approach and does not support authentication. The schemes do not support authentication as well. Earlier work provides privacy preserving authenticated access control in cloud. However, the authors take a centralized approach where a single key distribution center (KDC) distributes secret keys and attributes to all users. Unfortunately, a single KDC is not only a single point of failure but difficult to maintain because of the large number of users that are supported in a cloud environment. We, therefore, emphasize that clouds should take a decentralized approach while distributing secret keys and attributes to users. It is also quite natural for clouds to have many KDCs in different locations in the world.

DISADVANTAGES

- Provides privacy preserving authenticated access control in cloud. However, the authors take a centralized approach where a single key distribution center (KDC) distributes secret keys and attributes to all users. Unfortunately, a single KDC is not only a single point of failure but difficult to maintain because of the large number of users that are supported in a cloud environment
- Key revocation not possible

4. PROPOSED SYSTEM

Proposed a decentralized approach, their technique does not authenticate users, who want to remain anonymous while accessing the cloud. Proposed distributed access control mechanism in clouds. However, the scheme did not provide user authentication. The other drawback was that a user can create and store a file and other users can only read the file. Write access was not permitted to users other than the creator. In the preliminary version of this paper, we extend our previous work with added features which enables to authenticate the validity of the message without revealing the identity of the user who has stored information in the cloud. In this version we also address user revocation. We use attribute based signature scheme to achieve authenticity and privacy. our scheme is resistant to replay attacks, in which a user can replace fresh data with stale data from a previous write, even if it no longer has valid claim policy. This is an important property because a user, revoked of its attributes, might no longer be able to write to the cloud.

ADVANTAGES

- Our scheme is robust and decentralized; most of the others are centralized. Our scheme also supports privacy preserving authentication, which is not supported by others. Most of the schemes do not support user revocation, which our scheme does.
- We compare the computation and communication costs incurred by the users

and clouds and show that our distributed approach has comparable costs to centralized approaches. The most expensive operations involving pairings and is done by the cloud. If we compare the computation load of user during read we see that our scheme has comparable costs.

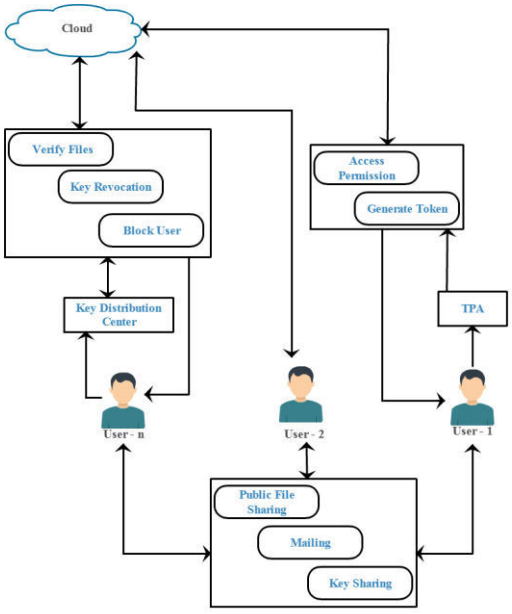

Fig.1. Overall Architecture of Proposed System

5.MODULE DESCRIPTION

5.1) Service Request to TPA:

User will send request to Third Party Authenticator (TPA) for registration. The client made request to the key manager for the public key, which will be generated according to the policy associated with the file. Different policies for files, public key also differs. But for same public key for same policy will be generated. Then the client generates a private key by combining the username, password and security credentials. Then the file is encrypted with the public key and private key and forwarded to the cloud.

5.2) TPA Policy Creation:

TPA provides the rules and regulations to be followed by Creator, Reader and Writer. The client can download the file after completion of the authentication process. As the public key maintained by the key manager, the client request the key manager for public key. The authenticated client can get the public key. Then the client can decrypt the file with the public key and the private key. The users credentials were stored in the client itself. During download the file the cloud will authenticate the user whether the user is valid to download the file.

5.3) User File Upload or file creator:

File creator after getting proper authentication uploads his files in the cloud. The policy of a file may be revoked under the request by the client, when expiring the time period of the contract or completely move the files from one cloud to another cloud environment. When any of the above criteria exists the policy will be revoked and the key manager will completely removes the public key of the associated file. So no one recover the control key of a revoked file in future. For this reason we can say the file is assuredly deleted.

5.4) KDC Key generation

Key Distribution Centers which are decentralized generate different keys to different types of user after getting tokens from users. To recover the file, the client must request the key manager to generate the public key. For that the client must be authenticated. The attribute based encryption standard is used for file access which is authenticated via an attribute associated with the file. With file access control the file downloaded from the cloud will be in the format of read only or write supported. Each user has associated with policies for each file. So the right user will access the right file.

5.5) Key revocation

Ability to limit and control the access to host systems and applications via communication links. To achieve, access must b ect policies with the files.

Whenever there is misbehaviour detected upon a user his key is revoked and user can neither use or re-enter the cloud environment.

5.6) Cloud Admin:

Cloud admin has the list of key distribution centers and TPA. Admin sets the norms to be followed by TPA and KDC. It monitors the key generation policies and informs abnormal behaviours.

6.RESULTS

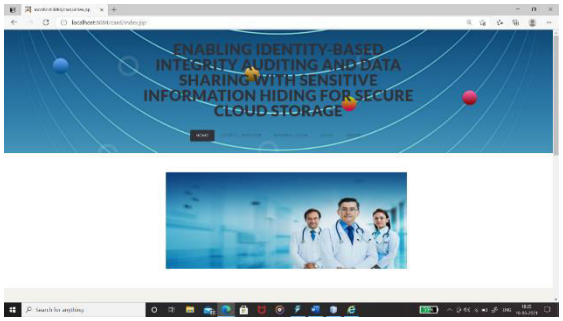

Fig 2 : Home Page

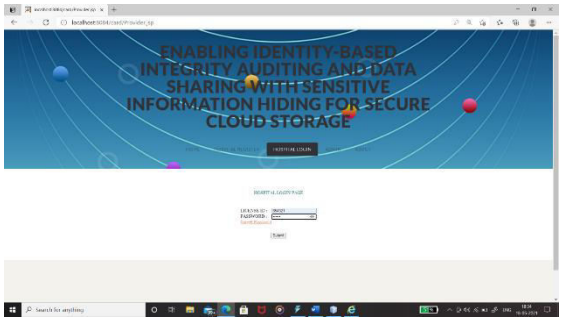

Fig 3 :Hospital Login Page

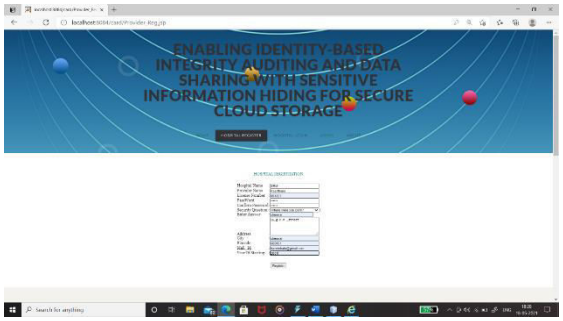

Fig 4 : Hospital Register page

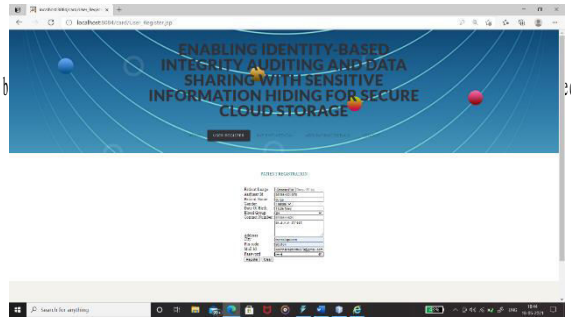

Fig 5 : User Register Page

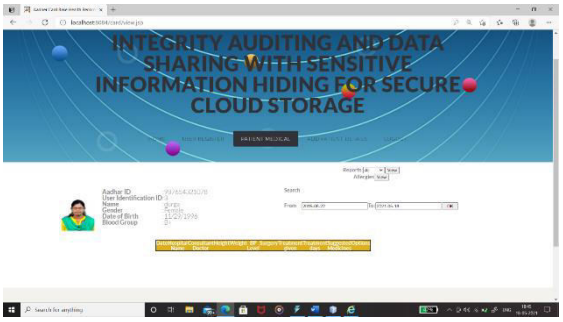

Fig 6 : Paitent Medical Details

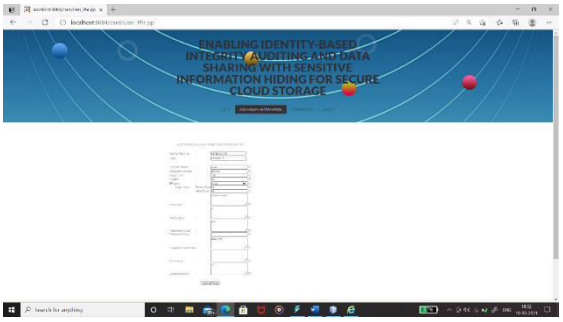

Fig 7 :Patient Health Information Details

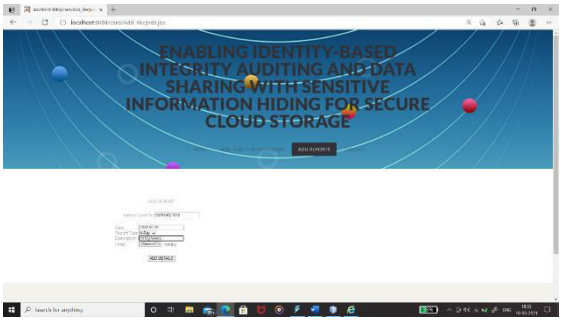

Fig 8 : Add Medical Report

7. CONCLUSION

We proposed an identity-based data integrity auditing scheme for secure cloud storage, which supports data sharing with sensitive information hiding. In our scheme, the file stored in the cloud can be shared and used by others on the

condition that the sensitive information of the file is protected. Besides, the remote data integrity auditing is still able to be efficiently executed. The security proof and the experimental analysis demonstrate that the proposed scheme achieves desirable security and efficiency.

REFERENCES

[1] C. Sivapragash, S. R. Thilaga, and S. S. Kumar, "Advanced cloud computing in smart power grid," in Proc. IET Chennai 3rd Int. Sustain. Energy Intell. Syst., 2014, pp. 356–361.

[2] C. Sivapragash, S. Padmanaban, H. Eklas, J. B. Holmnielsen, and R. Hemalatha, "Location-based optimized service selection for data management with cloud computing in smart grids," Energies, vol. 12, no. 23, 2019, Art. no. 4517.

[3] S. Kumar and C. Sivapragash, "Time orient traffic estimation approach to improve performance of smart grids," J. Comput. Theor. Nanosci., vol. 13, no. 8, pp. 5037–5045, 2016.

[4] D. G. Chandra and R. S. Bhadoria, "Role of G-cloud in citizen centric governance," in Proc. IEEE Int. Conf. Parallel Distrib. Grid Comput., 2012, pp. 44–48.

[5] K. Ren, C. Wang, and Q. Wang, "Security challenges for the public cloud," IEEE Internet Comput., vol. 16, no. 1, pp. 69–73, Jan./Feb. 2012.

[6] K. S. Jadon, R. S. Bhadoria, and G. S. Tomar, "A review on costing issues in big data analytics," in Proc. Int. Conf. Comput. Intell. Commun. Netw., 2015, pp. 727–730.

[7] R. S. Bhadoria, "Security architecture for cloud computing," Handbook of Research on Securing Cloud-Based Databases With Biometric Applications. Hershey, PA, USA: IGI Global, 2015.

[8] G. Ateniese et al., "Provable data possession at untrusted stores," in Proc. 14th ACM Conf. Comput. Commun. Secur., 2007, pp. 598–609.

[9] A. Juels and B. S. Kaliski Jr, "Pors: Proofs of retrievability for large files," in Proc. 14th ACM Conf. Comput. Commun. Secur., 2007, pp. 584–597.

[10] K. Liang et al., "A DFA-based functional proxy re-encryption scheme for secure public cloud data sharing," IEEE Trans. Inf. Forensics Secur., vol. 9, no. 10, pp. 1667–1680, Oct. 2014.