INCLUSIVE AUDIT IN IDENTITY BASED INFORMATION OUTSOURCING WITH CLOUD COMPUTING ENVIRONMENT

Prof. K. Chandra Prabha[1], Miss. Kamali. R[2], Asst. Prof. Rajanandhini. M[3]

[1]*Head of the Department, Computer Application Department, Alagappa Chettiar Government College of engineering & Technology*

[2]*P.G. Student, Computer Application Department, Alagappa Chettiar Government College of engineering & Technology*

[3]*Guide, Asst. Prof., Computer Application Department, Alagappa Chettiar Government College of engineering & Technology*

Abstract - **The cloud storage system provides convenient file storage and sharing services for distributed clients. In order to solve the integrity, we present identity-based data outsourcing (IBDO), outsourcing and original auditing concerns about outsourced documents, The program is equipped with an ideal feature that facilitates existing recommendations to protect outsourcing data. First of all, our IBDO plan Allows the user to authorize the dedicated agent to upload the data to the cloud storage server (for example, the company can authorize) Some employees upload files to the company's cloud account in a controlled manner. The agent is identified and authorized Identifiable identity eliminates complex certificate management in conventional secure distributed computing systems. Second, Our IBDO program is a comprehensive audit that our program not only allows for formal integrity audits in existing programs Used to protect the outsourced data, but also allows the review of external data sources, types and consistency information.**

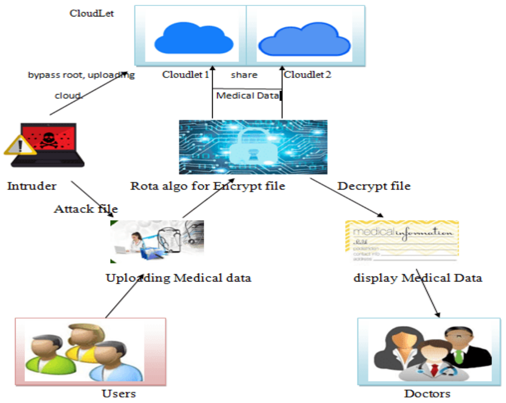
Index Terms— **Cloud storage, data outsourcing, proof of storage, remote integrity proof, public auditing.**

I. INTRODUCTION

Cloud platform provides powerful storage services to individuals and organizations. It brings great benefits of allowing on-the-move access to the outsourced files, simultaneously relieves file-owners from complicated local storage management and maintenance. However, some security concerns may impede users to use cloud storage. Among them, the *integrity* of outsourced files is considered as a main obstacle, since the users will lose physical control of their files after outsourced to a cloud storage server maintained by some cloud service provider (CSP). Thus, the file-owners may worry about whether their files have been tampered with, especially for those of importance. Considerable efforts have been made to address this issue. Among existing proposals, *provable data possession* (PDP) is a promising approach in proof of storage (PoS). With PDP, the file-owner only needs to retain a small amount of parameters of outsourced files and a secret key. To check whether or not the outsourced files are kept intact, the file owner or an auditor can challenge the cloud server with low communication overheads and computation costs. If some part of the file has been altered or deleted, for example, due to random hardware failures, the cloud storage server would not be able to prove the data integrity to convince the clients. We observe two critical issues not well addressed in existing proposals. First, most schemes lack a controlled way of delegable outsourcing. One may note that many cloud storage systems (e.g., Amazon, Dropbox, Google Cloud storage) allow the account owner to generate signed URLs using which any other designated entity can upload, and modify content on behalf of the user. However, in this scenario, the delegator cannot validate whether or not the authorized one has uploaded the file as specified or verify whether or not the uploaded file has been kept intact. Hence, the delegator has to fully trust the delegates and the cloud server. In fact, the file-owner may not only need to authorize some others to generate files and upload to a cloud, but also need to verifiably guarantee that the uploaded files have been kept unchanged. For instance, in *Electronic Health Systems* (EHS) when consulting a doctor, the patient needs to delegate her doctor to generate *electronic health records* (EHRs) and store them at a remote

EHRs center maintained by a CSP [7]. In another typical scenario of cloud-aided office applications, a group of engineers in different places may fulfill a task in cooperation. The group leader can create a cloud storage account and authorize the members with secret warrants. The behavior of the group members and the cloud server should be verifiable. Second, existing PoS-like schemes, including PDP and *Proofs of Retrievability* (PoR) [8], do not support data log related auditing in the process of data possession proof. The logs are critical in addressing disputes in practice. For example, when the patient and doctor in EHS get involved medical disputes, it would be helpful if some specific information such as outsourcer, type and generating time of the outsourced EHRs are auditable. However, there exist no PoS-like schemes that can allow validation of this important information in a multi-user setting. It brings great benefits of allowing on-the-move access to the outsourced files, simultaneously relieves file-owners from complicated local storage management and maintenance. However, some security concerns may impede users to use cloud storage.Among them, the integrity of outsourced files is considered as a main obstacle , since the users will lose physical control of their files after outsourced to a cloud storage server maintained by some cloud service provider (CSP). Thus, the file-owners may worry about whether their files have been tampered with, especially for those of importance. In fact, the file-owner may not only need to authorize some others to generate files and upload to a cloud, but also need to verifiably guarantee that the uploaded files have been kept unchanged.

II.SYSTEM ARCHITECTURE

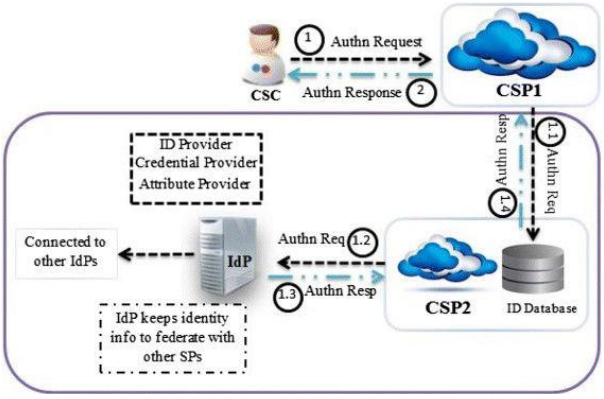


The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user.

Efficient and intelligent output design improves the system's relationship to help user decision-making. One may note that many cloud storage systems (e.g., Amazon, Drop box, Google Cloud storage) allow the account owner to generate signed URLs using which any other designated entity can upload, and modify content on behalf of the user. However, in this scenario, the delegator cannot validate whether or not the authorized one has uploaded the file as specified or verify whether or not the uploaded file has been kept intact. PoS-like schemes, including PDP and Proofs of Irretrievability (PoR), do not support data log related auditing in the process of data possession proof. We investigated proofs of storage in cloud in a multi-user setting. We introduced the notion of identity based data outsourcing and proposed a secure IBDO scheme. It allows the file-owner to delegate her outsourcing capability to proxies. Only the authorized proxy can process and outsource the file on behalf of the file-owner. Both the file origin and file integrity can be verified by a public auditor. The identity-based feature and the comprehensive auditing feature make our scheme. Although cloud computing promises lower costs, rapid scaling, easier maintenance, and service availability anywhere, anytime, a key challenge is how to ensure and build confidence that the cloud can handle user data securely. A recent Microsoft survey found that "58 percent of the public and 86 percent of business leaders are excited about the possibilities of cloud computing. But more than 90 percent of them are worried about security, availability, and privacy of their data as it rests in the cloud."1 This tension makes sense: users want to maintain control of their data, but they also want to benefit from the rich services that application developers can provide using that data. So far, the cloud offers little platform-level support or standardization for user data protection beyond data encryption at rest, most likely because doing so is nontrivial. Protecting user data while enabling rich computation requires both specialized expertise and resources that might not be readily available to most application developers. Cloud storage and synchronization services let users access their digital content any time, from anywhere, and with any device—smartphone, tablet, or desktop PC. It has become common for people to outsource their data to the cloud without being concerned about how the storage is managed. At the same time, the rapid adoption of portable equipment and the growing integration of computation into consumer products have brought mobile and pervasive computing into the mainstream, with cloud storage and synchronization services widely used across mobile devices.

We introduce a model for provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking supports large data sets in widely-distributed storage systems.

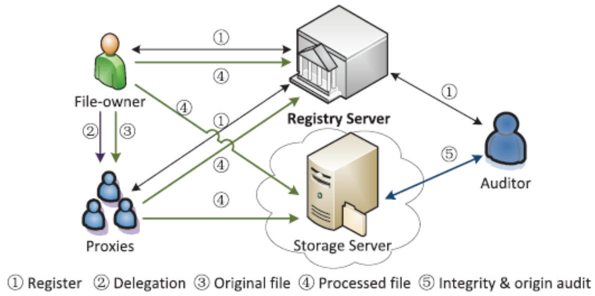
III. WORKING PROCEDURE



To address the above issues for securing outsourced data in clouds, this paper proposes an *identity-based data outsourcing* (IBDO) system in a multi-user setting. Compared to existing PoS like proposals, our scheme has the following distinguishing features.

- **Identity-based outsourcing.** A user and her authorized proxies can securely outsource files to a remote cloud server which is not fully trustable, while any unauthorized ones cannot outsource files on behalf of the user. The cloud clients, including the file-owners, proxies and auditors, are recognized with their identities, which avoids the usage of complicated cryptographic certificates. This delegate mechanism allows our scheme to be efficiently deployed in a multi-user setting.
- **Comprehensive auditing.** Our IBDO scheme achieves a strong auditing mechanism. The integrity of outsourced files can be efficiently verified by an auditor, even if the files might be outsourced by different clients. Also, the information about the origin, type and consistence of outsourced files can be publicly audited. Similar to existing publicly auditable schemes, the comprehensive auditability has advantages to allow a public common auditor to audit files owned by different users, and in case of disputes, the auditor can run the auditing protocol to provide convincing judicial witnesses without requiring disputing parties to be corporative.
- **Strong security guarantee.** Our IBDO scheme achieves strong security in the sense that: (1) it can detect any unauthorized modification on the outsourced files and (2) it can detect any misuse/abuse of the delegations/authorizations. These security properties are formally proved against active colluding attackers. To the best of our knowledge, this is the first scheme that simultaneously achieves both goals. A thorough comparison of our scheme with several related schemes is shown in Table I in terms of delegated data outsourcing, certificate-freeness, data origin auditing, data consistence validation and public verifiability. We also conduct extensive experiments on our proposed IBDO scheme and make comparisons with Shacham and Waters' (SW) PoR scheme. Both theoretical analyses and experimental results confirm that the IBDO proposal provides resilient security properties without incurring any significant performance penalties.

The architecture of our IBDO system is an IBDO system consists of five types of entities, that is, file owners, proxies, auditors, registry server, and storage server. Generally, the file-owners, proxies and auditors are cloud clients. The registry server is a trusted party responsible for setting up the system and responding to the clients ‘registration, and also allows the registered clients to store the public parameters of outsourced files. The cloud storage server provides storage services to the registered clients for storing outsourced files. In real-world applications, an organization buys storage services from some CSP, and the IT department of the organization can play the role of a registry server.


File-owner
Registry Server
Auditor
Proxies
Storage Server
①
②
③
④
⑤
① Register ② Delegation ③ Original file ④ Processed file ⑤ Integrity & origin audit

The file-owner and her authorized proxies can outsource files to the cloud server. Specifically, on behalf of the owner, the authorized proxy processes the file, sends the processed results to the storage server, and uploads the corresponding public parameters of the file to the registry server. Neither the file-owner nor the proxy is required to store the original file or the processed file locally. The duty of the auditor is to check the integrity of outsourced files and their origin-like general log information by interacting with the cloud storage server without retrieving the entire file. Adversary Model and Security Goals An IBDO system confronts two types of active attacks. The cloud client may impersonate others, specifically, she may impersonate an owner or another authorized proxy, or abuse delegation, and in this way she can process a file and outsource it to the storage server in an unwanted way. On the other hand, a malicious storage server may modify or even remove the outsourced files (for example, for saving storage space or due to hardware failures), especially for the rarely accessed files. Taking into account the above realistic attacks, a secure IBDO system should satisfy the following requirements:• Dedicated delegation: A delegation issued by a file-owner can only be used by the specific authorized proxy to outsource specified files in a designated way. Even the authorized proxy cannot abuse it to outsource unspecified files, and multiple proxies cannot cooperatively deduce valid delegation for a new warrant to outsource an unspecified file.• Comprehensive auditing: Not only the integrity of the outsourced file, but also the log information about the origin, type and consistence of the outsourced files should be verifiable by the auditors.

The integrity auditing ensures that the outsourced files have been kept intact; the other general log information auditing ensures that the file has been outsourced in the designated way. With comprehensive auditing, an IBDO system can provide convincing judicial witnesses to address disputes. The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement. We summarize the computation costs of each algorithm and protocol in Table II, which shows a comparison between our scheme and Shacham and Waters' publicly verifiable PoR scheme over bilinear groups [9]. The costs of file tag generation and verification are not counted as they are determined by the specific signature scheme *S* chosen at the processing file phase. In the table, M and E denote one multiplication and one exponentiation in *G*, respectively; similarly, MT and ET respective denote one multiplication and one exponentiation in GT ; Mq and Aq represent one multiplication and one addition in Zq, respectively; P denotes one bilinear pairing evaluation $\hat{e} : G \times G \rightarrow GT$. We do not differentiate hash valuations of $H1$, $H2$ or $H3$, and denote them commonly as H. Since both $g1$ and $g2$ are public parameters in our IBDO scheme, $\hat{e}(g1, g2)$ can be pre-computed and looks also as a public value. As known that exponentiations and pairings are more time-consuming compared to the other ones, they would essentially determine the efficiency of these two schemes. In both schemes, all verification cases at user side take only constant pairings, that is, when verifying a private key issued by registry server, validating a delegation generated by a file owner, or checking a proof in a round of (comprehensive) auditing. All other phases do not involve pairing evaluations. For processing a file *M* with sectors $\{mi, j\}r\times c$, SW scheme takes $(rc + r)$ exponentiations, rc multiplications and r hash evaluations. Our scheme only incurs one more time-consuming exponentiation, plus r efficient multiplications. Thus, both SW scheme and ours enjoy the same efficiency level for processing the same file. When performing a round of auditing protocol on an outsourced file, both schemes require the same number (i.e., |*I* |) of exponentiations in group *G* at cloud storage server side. While at auditor side, SW scheme takes *(*|*I*| + *c)* exponentiations in group *G* and two pairings to check a proof, and our scheme would take another four pairings in order to auditing the corresponding delegation.

IV. CONCLUSION

In this paper, we investigated proofs of storage in cloud in a multi-user setting. We introduced the notion of identity based data outsourcing and proposed a secure IBDO scheme. It allows the file-owner to delegate her outsourcing capability to proxies. Only the authorized proxy can process and outsource the file on behalf of the file-owner. Both the file origin and file integrity can be verified by a public auditor. The identity based feature and the comprehensive auditing feature make our scheme advantageous over existing PDP/PoR schemes. Security analyses and experimental results show that the proposed scheme is secure and has comparable performance as the SW scheme. We investigated proofs of storage in cloud in a multi-user setting. We introduced the notion of identity based data outsourcing and proposed a secure IBDO scheme. It allows the file-owner to delegate her outsourcing capability to proxies. Only the authorized proxy can process and outsource the file on behalf of the file-owner. Both the file origin and file integrity can be verified by a public auditor. The identity-based feature and the comprehensive auditing feature make our scheme advantageous over existing PDP/PoR schemes. Security analyses and experimental results show that the proposed scheme is secure and has comparable performance as the SW scheme. The identity-based feature and the comprehensive auditing feature make our scheme advantageous over existing PDP/PoR schemes. Security analyses and experimental results show that the proposed scheme is secure and has comparable performance as the SW scheme. The process of data possession proof. The logs are critical in addressing disputes in practice. When the patient and doctor in EHS get involved medical disputes, it would be helpful if some specific information such as outsourcer, type and generating time of the outsourced EHRs are auditable. Also, in both schemes, the time cost at the auditor side is larger than that at the cloud side, which is consistent with the theoretical analysis shown in Table II. Note that the former can be shared by several auditors in a multi-auditor setting.

V. ACKNOWLEDGEMENT

This research was supported by my guide and supervisors. I thank my faculties and friends who provided insight and expertise that greatly assisted the research, although they may not agree with all of the interpretations of this paper. I thank **Mrs. Rajanandhini. M,** Guide, Assistant Professor, and **Mrs. Chandra Prabha. K**, H.O.D, Alagappa Chettiar Government College of Engineering & Technology for commenting that greatly improved the manuscript.

REFERENCES

[1] D. Song, E. Shi, I. Fischer, and U. Shankar, "Cloud data protection for the masses," *Computer*, vol. 45, no. 1, pp. 39–45, Jan. 2012.

[2] C.-K. Chu, W.-T. Zhu, J. Han, J. K. Liu, J. Xu, and J. Zhou, "Security concerns in popular cloud storage services," *IEEE Pervasive Comput.*, vol. 12, no. 4, pp. 50–57, Oct. 2013.

[3] K. Yang and X. Jia, "Data storage auditing service in cloud computing:Challenges, methods and opportunities," *World Wide Web*, vol. 15, no. 4,pp. 409–428, 2012.

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

[5] J. Sun and Y. Fang, "Cross-domain data sharing in distributed electronic health record systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 6, pp. 754–764, Jun. 2010.

[6] J. Sun, X. Zhu, C. Zhang, and Y. Fang, "HCPP: Cryptography based secure EHR system for patient privacy and emergency healthcare," in *Proc. IEEE 31st Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2011, pp. 373–382.

[7] L. Guo, C. Zhang, J. Sun, and Y. Fang, "PAAS: A privacy-preserving attribute-based authentication system for eHealth networks," in *Proc. IEEE 32nd Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2012, pp. 224–233.

[8] A. Juels and B. S. Kaliski, Jr., “PORs: Proofs of retrievability for large files,” in *Proc. 14th ACM Conf. Comput. Commun. Secur.*, New York,NY, USA, 2007, pp. 584–597.

[9] H. Shacham and B. Waters, “Compact proofs of retrievability,” *J. Cryptol.*, vol. 26, no. 3, pp. 442–483, 2013.

[10] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, “Privacypreserving public auditing for secure cloud storage,” *IEEE Trans.Comput.*, vol. 62, no. 2, pp. 275–362, Feb. 2013.

[11] B. Wang, B. Li, and H. Li, “Panda: Public auditing for shared data with efficient user revocation in the cloud,” *IEEE Trans. Serv. Comput.*,vol. 8, no. 1, pp. 92–106, Feb. 2015.

[12] F. Chen, T. Xiang, Y. Yang, and S. S. M. Chow, “Secure cloud storage meets with secure network coding,” *IEEE Trans. Comput.*, vol. 65, no. 6, pp. 1936–1948, Jun. 2016.