International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 09
May 2016

# Providing Authorized Access Control on Encrypted Data and Indirect Trust for Cloud Storage Systems

**[1] P.S. Abhilash & [2] P.Namrata**

[1]M.Tech,, Dept of CSE, Intell Engineering College,Anantapuramu,Affiliated to JNTUA, AP, India
[2]Assistant Professor, Dept of CSEIntell Engineering College,Anantapuramu,Affiliated to JNTUA, AP, India

***Abstract**: Cloud Service Provider (CSPs)offers Storage-as-a-Service as a paid facility that enables organizations to outsource their sensitive data to be stored on remote servers. In this paper, a cloud-based storage scheme is proposed that allows the data owner to benefit from the facilities offered by the CSP and enables indirectmutual trust between them. The proposed scheme has four important features: (i) it allows the owner to outsource sensitive data to a CSP, and perform full block-level dynamic operations on the outsourced data, i.e., block modification, insertion, deletion, and append, (ii) it ensures that authorized users (i.e., those who have the right to access the owner's file) receive the latest version of the outsourced data, (iii) it enables indirect mutual trust between the owner and the CSP, and (iv) it allows the owner to grant or revoke access to the outsourced data. The security issues of the proposed scheme has been discussed. And, its performance through theoretical analysis and a prototype implementation on Amazon cloud platform to evaluate storage, communication, and computation overheads has been justified.*

***Key words:**Outsourcing data; access control;mutual trust; dynamic environment.*

## I.INTRODUCTION

In current world, many organizations have shifted to cloud and some of them have been shifting to cloud based system. They are producing a large amount of sensitive data including personal information, electronic health records, and financial data. The local management of such huge amount of data is problematic and costly due to the requirements of high storage capacity and qualified personnel. Therefore, Cloud service providers (CSPs) offered Storage-as-a-Service as anemerging solution to mitigate the burden of large local data storage and reduce the maintenance cost by means of outsourcing data storage. Since the data owner physically releases sensitive data to a remote CSP, there are some concerns regarding confidentiality, integrity, and access control of the data. The confidentiality feature can be guaranteed by the owner via encrypting the data before outsourcing to remote cloud server. For verifying data integrity over cloud servers, researchers have proposed provable data possession technique to validate the intactness of data stored on remote sites. A number of PDP protocols have been presented to efficiently validate the integrity of data, e.g., Proof of retrievability was introduced as a stronger technique than PDP in the sense that the entire data file can be reconstructed from portions of the data that are reliably stored on the servers.

Main Contributions

- The implementation of a cloud-based storage scheme that has the following roles: (i) allowing a data owner to outsource the data to a CSP, and perform full at the block-

level operations more dynamically, i.e., it supports operations such as block modification, insertion, deletion, and append; (ii) ensuring the newness property, i.e., the authorized users receive the most recent version of the outsourced data; (iii) developing an indirect mutual trust between the data owner and the CSP since each party resides in a different trust domain; and (iv) enabling the access control for the outsourced data.

- Also the security features of the proposed scheme are discussed. Besides, its performance is justified through theoretical analysis and a prototype implementation on Amazon cloud platform to evaluate storage, communication, and computation overheads.

## II RELATED WORKS

A model for provable data possession (PDP) is introduced 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 system. Here two provably-secure PDP schemes that are more efficient than previous solutions are presented which even when compared with schemes that achieve weaker guarantees give proper performance. In particular, the overhead at the server is low (or even constant), as opposed to linear in the size of the data. Experiments using the present implementation verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation.[1]

Checking data possession in networked information systems such as those related to critical infrastructures (power facilities, airports, data vaults, defense systems, etc.) is a matter of crucial importance. Remote data possession checking protocols permit to check that a remote server can access an uncorrupted file in such a way that the verifier does not need to know beforehand the entire file that is being verified.Unfortunately, current protocols only allow a limited number of successive verifications or are impractical from the computational point of view. In this paper, it has been presented that a new remote data possession checking protocol that: 1) it allows an unlimited number of file integrity verifications; 2) its maximum running time can be chosen at set-up time and traded off against storage at the verifier.[2]

Storage outsourcing is a rising trend which prompts a number of interesting security issues, many of which have been extensively investigated in the past. However, Provable Data Possession (PDP) is a topic that has only recently appeared in the research literature. The main issue is how to frequently, efficiently and securely verify that a storage server is faithfully storing its client's (potentially very large) outsourced data. The storage server is assumed to be untrusted in terms of both security and reliability. (In other words, it might maliciously or accidentally erase

hosted data; it might also relegate it to slow or off-line storage.) The problem is exacerbated by the client being a small computing device with limited resources. Prior work has addressed this problem using either public key cryptography or requiring the client to outsource its data in encrypted form.In this paper, a highly efficient and provably secure PDP technique has been constructed based entirely on symmetric key cryptography, while not requiring any bulk encryption. Also, in contrast with its predecessors, our PDP technique allows outsourcing of dynamic data, i.e, it efficiently supports operations.[3]

Proposed system considers the problem of efficiently proving the integrity of data stored at untrusted servers. In the provable data possession (PDP) model, the client preprocesses the data and then sends it to an untrusted server for storage, while keeping a small amount of meta-data. The client later asks the server to prove that the stored data has not been tampered with or deleted (without downloading the actual data). However, the original PDP scheme applies only to static (or append-only) files. A definitional framework and efficient constructions for dynamic provable data possession (DPDP) has been presented, which extends the PDP model to support provable updates to stored data. A new version of authenticated dictionaries based on rank information is used. The price of dynamic updates is a performance change from O(1) to O(logn) (or $O(n^{\varepsilon}logn)$, for a file consisting of n blocks, while maintaining the same (or better, respectively) probability of misbehavior detection. The present experiments show that this slowdown is very low in practice (e.g. 415KB proof size and 30ms computational overhead for a 1GB file). It is

p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 09
May 2016

also shown how to apply the DPDP scheme to outsourced file systems and version control systems (e.g. CVS).[4]

Cloud Computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This unique paradigm brings about many new security challenges, which have not been well understood. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. In particular, the task of allowing a third party auditor (TPA) has been considered, on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud. The introduction of TPA eliminates the involvement of client through the auditing of whether his data stored in the cloud is indeed intact, which can be important in achieving economies of scale for Cloud Computing. The support for data dynamics via the most general forms of data operation, such as block modification, insertion and deletion, is also a significant step toward practicality, since services in Cloud Computing are not limited to archive or backup data only. While prior works on ensuring remote data integrity often lacks the support of either public verifiability or dynamic data operations, this paper achieves both. First identify the difficulties and potential security problems of direct extensions with fully dynamic data updates from prior works and then show how to construct an elegant verification scheme for seamless integration of these two salient features in our protocol design. In particular, to achieve efficient data dynamics, the Proof of Retrievability model [1] by manipulating the classic Merkle Hash Tree (MHT) construction for block tag

authenticationhas been improved. Extensive security and performance analysis show that the proposed scheme is highly efficient and provably secure.[5]

## III BACKGROUND

Commonly, traditional access control techniques assume the existence of the data owner and the storage servers in the same trust domain. This assumption, however, no longer holds when the data is outsourced to a remote CSP, which takes the full charge of the outsourced data management, and resides outside the trust domain of the data owner.The various security and privacy issues to be addressed in CSP:CSP needs to be safeguarded from a dishonest owner, who attempts to get illegal compensations by falsely claiming data corruption over cloud servers. This concern, if not properly handled, can cause the CSP to go out of business.

## IV PROPOSED SYSTEM

In this work, a scheme has been proposed that addresses important issues related to outsourcing the storage of data, namely dynamic data, newness, mutual trust, and access control.

The remotely stored data can be not only accessed by authorized users, but also updated and scaled by the owner. After updating, authorized users should receive the latest version of the data (newness property), i.e., a technique is required to detect whether the received data is stale. Mutual trust between the data owner and the CSP is another imperative issue, which is addressed in the proposed scheme.

A mechanism is introduced to determine the dishonest party, i.e., misbehavior from any side is detected and the responsible party is identified.

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 09
May 2016

Last but not least, the access control is considered, which allows the owner to grant or revoke access rights to the outsourced data.

ADVANTAGES OF PROPOSED SYSTEM

- It allows a data owner to outsource the data to a CSP, and perform full dynamic operations at the block-level, i.e., it supports operations such as block modification, insertion, deletion, and append;
- It ensures the newness property, i.e., the authorized users receive the most recent version of the outsourced data;
- It establishes indirect mutual trust between the data owner and the CSP since each party resides in a different trust domain; and
- It enforces the access control for the out sourced data.

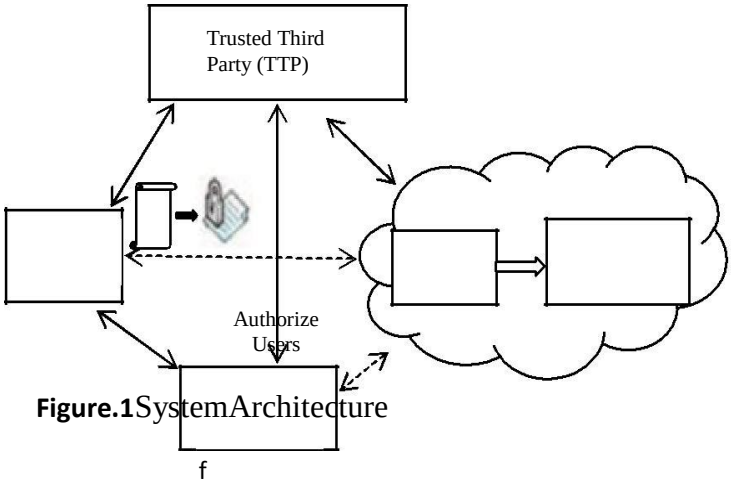


**Figure.1**SystemArchitecture

MODULES DESCRIPTION:

(a) Data Owner Module

In this module, the data owner module has been developed, where A data owner that can be an organization generating sensitive data to be stored in the cloud and made available

for controlled external use. First, the data owner has to register with the cloud service provider, to store their data in Cloud Server. After Registering, the data owners gets credential login access using their perspective username and password. The data owner then can upload their files in it. The details of uploaded files are also listed in the separate menu. All the uploaded files are encrypted securely.

(b) Cloud Service Provider Module

In this module the Cloud Service Provider is developed. CSP who manages cloud servers and provides paid storage space on its infrastructure to store the owner's files and make them available for authorized users. All the files uploaded by the data owner are saved in Cloud Server managed by the cloud service providers. It has been considered that, the CSP is untrusted, and thus the confidentiality and integrity of data in the cloud may be at risk. For economic incentives and maintaining a reputation, the CSP may hide data loss, or reclaim storage by discarding data that has not been or is rarely accessed.

(c) Authorized Users Module

In this module, the authorized user module has been developed, where the authorized user is a set of owner's clients who have the right to access the remote data. Also the system modelconsidered; On the other hand, a data owner and authorized users may collude and falsely accuse the CSP to get a certain amount of reimbursement. They may dishonestly claim that data integrity over cloud servers has been violated, or the CSP has returned a stale file that does not match the most recent modifications issued by the owner.

(d) Trusted Third Party (TTP) Module

In this module, the TTP, a trusted third party (TTP) has been developed, an entity who is trusted by all other system components, and has capabilities to detect/specify dishonest parties. In this module TTP has monitors the data owners file by verifying the data owner's file and stored the file in a database.Also ttp checks the CSP(CLOUD SERVICE PROVIDER),and find out whether the csp is authorized one or not.

## V CONCLUSION

In this paper, a cloud-based storage scheme has been proposed which supports outsourcing of dynamic data, where the owner is capable of not only archiving and accessing the data stored by the CSP, but also updating and scaling this data on the remote servers. The proposed scheme enables the authorized users to ensure that they are receiving the most recent version of the outsourced data. Moreover, in case of dispute regarding data integrity newness, a TTP is able to determine the dishonest party. The data owner enforces access control for the outsourced data by combining three cryptographic techniques: broadcast encryption, lazy revocation, and key rotation. The security features of the proposed scheme has been studied.

At present the overheads added by the present scheme are investigatedwhen they are incorporated into a cloud storage model for static data with only confidentiality requirement. The storage overhead is ≈ 0.4% of the outsourced data size, the communication overhead due to block-level dynamic changes on the data is ≈ 1% of the block size, and the communication overhead due to retrieving the data is ≈ 0.2% of the outsourced data size. For a large organization with 105 users, performing dynamic operations and enforcing access control add about 63 milliseconds ofoverhead. Therefore, important features of outsourcing data storage can be supported without excessive overheads in storage, communication and computation.

## VI REFERENCES

*[1]G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song," Provable data possession at untrusted stores" in Proceedings of the 14th ACM Conference on Computer and Communications Security, ser. CCS '07, 2007, pp. 598–609.*

*[2]F. Seb´e, J. Domingo-Ferrer, A. Martinez-Balleste, Y. Deswarte, and J.-J.Quisquater, "Efficient remote data possession checking in critical information infrastructures," IEEE Trans. on Knowl.And Data Eng., vol. 20, no. 8, 2008.*

*[3] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks, 2008, pp. 1–10.*

*[4] C. Erway, A. K¨upc¸ ¨u, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in Proceedings of the 16th ACM Conference on Computer and Communications Security, 2009, pp. 213–222.*

*[5] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing,"*

*[6] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MR-PDP: multiple-replica provable data possession," in 28th IEEE ICDCS, 2008, pp. 411–420.*

*[7] A. F. Barsoum and M. A. Hasan, "On verifying dynamic multiple data copies over cloud servers," Cryptology ePrint Archive, Report 2011/447, 2011, 2011.*

p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 09
May 2016

*[8] K. D. Bowers, A. Juels, and A. Oprea, "HAIL: a high-availability and integrity layer for cloud storage," in CCS '09: Proceedings of the 16th ACM conference on Computer and communications security. New York, NY, USA: ACM, 2009, pp. 187–198.*

*[9] Y. Dodis, S. Vadhan, and D. Wichs, "Proofs of Retrievability via hardness amplification," in Proceedings of the 6th Theory of Cryptography Conference on Theory of Cryptography, 2009.*

*[10] H. Shacham and B. Waters, "Compact proofs of retrievability," in ASIACRYPT '08, 2008, pp. 90–107.*

*[11] M. Backes, C. Cachin, and A. Oprea, "Secure key-updating for lazy revocation," in 11th European Symposium on Research in Computer Security, 2006, pp. 327–346.*

*[12] D. Boneh, C. Gentry, and B. Waters, "Collusion resistant broadcast encryption with short ciphertexts and private keys," in Advances in Cryptology - CRYPTO, 2005, pp. 258–275.*

*[13] W. Wang, Z. Li, R. Owens, and B. Bhargava, "Secure and efficient access to outsourced data," in Proceedings of the 2009 ACM workshop on Cloud computing security, 2009, pp. 55–66.*

*[14] K. E. Fu, "Group sharing and random access in cryptographic storage file systems," Master's thesis, MIT, Tech. Rep., 1999.*

*[15] R. A. Popa, J. R. Lorch, D. Molnar, H. J. Wang, and L. Zhuang, "Enabling security in cloud storage SLAs with cloudproof," in Proceedings of the 2011 USENIX conference, 2011.*

*[16] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in INFOCOM'10, 2010, pp. 534–542.*

*[17] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in CCS '06, 2006, pp. 89–98.*

*[18] S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Over-encryption: Management of access control evolution on outsourced data," in Proceedings of the 33rd International Conference on Very Large Data Bases. ACM, 2007, pp. 123–134.*

*[19] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," in NDSS, 2005.*

*[20] K. E. Fu, "Group sharing and random access in cryptographic storage file systems," Master's thesis, MIT, Tech. Rep., 1999.*