iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 Issue: V Month of publication: May 2021

DOI: https://doi.org/10.22214/ijraset.2021.34568

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue V May 2021- Available at www.ijraset.com

Privacy Preserving Public Auditing for Secure Cloud Storage

Revti R. Adel[1], Dr. V. M. Deshmukh[2]
[1]P.G Student, Department of Computer Science and Engineering, PRMITR&R , Badnera
[2]Associate Professor, Department of Computer Science and Engineering, PRMITR&R , Badnera

Abstract: Cloud computing is the on-demand availability of computer system resources, especially data storage (cloud storage) and computing power, without direct active management by the user. By using Cloud storage, users can access applications, services, software whenever they require over the internet. Users can put their data remotely to cloud storage and get benefit of on-demand services and application from the resources. The cloud must have to ensure data integrity and security of data of user. The issue about cloud storage is integrity and privacy of data of user can arise. To maintain to overkill this issue here, we are giving public auditing process for cloud storage that users can make use of a third-party auditor (TPA) to check the integrity of data. Not only verification of data integrity, the proposed system also supports data dynamics. The work that has been done in this line lacks data dynamics and true public audit ability. The auditing task monitors data modifications, insertions and deletions. The proposed system is capable of supporting public audit ability, data dynamics and Multiple TPA are used for the auditing process.

Keywords: Cloud Storage, Data Dynamics, Public Auditing, Privacy Preserving

I. INTRODUCTION

A. Motivation

Computing[1] is the on-demand availability of computer system resources, especially data storage (cloud storage) and computing power, without direct active management by the user. The term is generally used to describe data centres available to many users over the Internet. Large clouds, predominant today, often have functions distributed over multiple locations from central servers. If the connection to the user is relatively close, it may be designated an edge server. Clouds may be limited to a single organization or be available to many organizations (public cloud). Cloud computing relies on sharing of resources to achieve coherence and economies of scale. In today’s era, Cloud Computing is attaining more and more generosity in the intellectual and industrial community. It becomes an imitation for recognizing everywhere, well- located, on-demand network connection for all the configurable computing resources (i.e., networks, services, servers, applications). Mainly, users can abandon the maintaining of IT services to cloud service providers (CSP), who are apt in providing knowledge and maintaining the large amount of IT resources. Cloud computing causes many new security issues and challenges on assuring the integrity and privacy of users data in the cloud. To entice these issues, our work uses the concept of secret key which is based on symmetric key cryptography, in which it allows the TPA to execute the auditing without exacting the local copy of the user’s stored data and hence sharply analyse the transmission and reckoning overhead as related to the straightforward data auditing approaches. So to accommodate the encryption with examining, our protocol ensures that during the efficient auditing process, the TPA could not get any information about the data context stored within the cloud server. Cloud Computing offers the internet-based service and also the use of computer technologies. This is worth the money and more strong processors, well-organized with software as well as services computing architecture and are changing the data into data centres on large scale. The expanding and extensible network connections make it possible that users can now use high quality services from data and maintain remote on data centres. Cloud offers great help to the users for storing data since they don’t have to care about the problems of hardware. Although these internet-based online applications or services grant a huge volume of spaces for storage and modifiable computing resources. However, it simultaneously avoids the liability of local machines for the maintenance of data. As a result, on the one hand, users are concerned about their cloud service providers for the availability and integrity of their data; although the cloud services are much more effective and reputable than claimed computing devices. The chances of both internal and external threats still exit for data integrity. For example, from time to time, the dismount and data mishap incidents of indicative cloud storage services appear. On the other hand, users cannot retain the local copy of the commerce data; even there abide various incentives for Cloud Service provider (CSP) to act unfaithfully towards cloud users with respect to the status of their commerce data.

Our proposed work is one of the first few in this field to deliberate on storage security of distributed data storage in Cloud Computing. Third Party Auditor (TPA) For an organization, it is important that, cloud which allows delving from single party audit, the commerce data to ensure data security and save the user's reckoning and data storage. It is essential to maintain the public auditing services for cloud data storage in the way those users can trust an autonomous third party auditor (TPA). On the behalf of users, TPA polls the integrity of the data within the cloud, and also maintains the users to poll the validity of data in the cloud. In addition, public auditing provides the external party to verify as well the correctness of already stored data across the external attacks to the user. However, these systems, as in [1] don't relate the privacy protection of the data. In cloud computing, this is the main disadvantage which induces the security of protocols. So the users who rely only on the TPA for security storage actually want that their data should be protected from the autonomous auditors. Means that a cloud service provider has indicative computation resource and storage space to maintain the users' data. It also has a knack in building and managing the distributed cloud storage server and ability to retain and handle live cloud computing systems. Those who put their huge data files into the cloud storage server can abate the clog of storage and data processing. It is essential for the user to store the data correctly and securely. Users should be equipped with certain security so they get the surety of data to be safe. As always, the Cloud service providers are online & inferred to have liberal storage capacity and reckoning power. The third party auditor is invariably online too. It makes every data access be in control. The Fig. 1 shows how third party auditor (TPA) it consist three different entities: the cloud user, the cloud server (CS) and the third party auditor (TPA) As shown in fig .1.

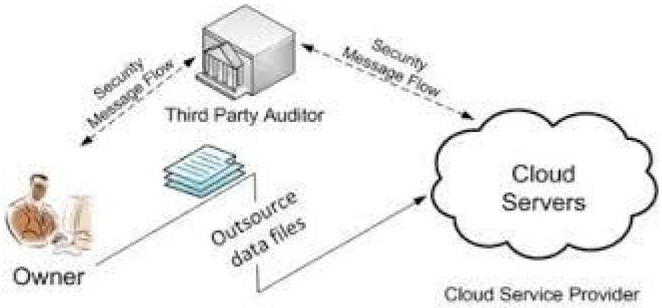

Fig. 1 : Third party auditor

The cloud user is the one who has a large number of data files that are stored in the cloud; the cloud server is the one who provides the data storage service like resources, software to the user. The cloud server is managed by a cloud service provider; the third-party auditor is the one who has a belief to access the cloud storage service for the benefit of the user whenever the user requests for data access. The TPA has capabilities and competence that the user does not have. They can also interact with a cloud server to access the stored data for different purposes in a different style. Every time it is not possible for the user to check the data which is stored on a cloud server that arrives online, burden to the user. So that's why to reduce online burden and maintain this integrity of cloud users may resort to TPA.

B. Objective

The proposed system aims to achieve the following objectives:

1) *Public Auditing:* The public verifiers (TPA) are capable to publicly certifying the integrity of the data to be shared away from retrieving the whole data within the cloud.
2) *Correctness:* The public verifiers (TPA) are adept at correctly verifying the integrity of data to be shared.
3) *Unforgeability:* Only the user within the group can have authority to generate valid verification (i.e., signatures) on shared data.
4) *Identity Privacy:* The public verifier (TPA) cannot distinguish the identity of the user on each block which shared data during the process of auditing.

II. LITERATURE SURVEY

Ateniese et al. [4] are the first to consider public audit ability in their defined "provable data possession" (PDP) model for ensuring possession of files on untrusted storages. In their scheme, utilize RSA based homomorphic tags for auditing outsourced data, thus public audit ability is achieved. However, Ateniese et al. do not consider the case of dynamic data storage, and the direct extension of their scheme from static data storage to dynamic case may suffer design and security problems.

Ateniese et al. [5] propose a dynamic version of the prior PDP scheme. However, the system imposes a priori bound on the number of queries and does not support fully dynamic data operations, i.e., it only allows very basic block operations with limited functionality, and block insertions cannot be supported.

Wang et al. [6] consider dynamic data storage in a distributed scenario, and the proposed challenge-response protocol can both determine the data correctness and locate possible errors. Similar to [5], they only consider partial support for dynamic data operation.

Juels et al. [7] describe a “proof of retrievability” (PoR) model, where spot-checking and error correcting codes are used to ensure both “possession” and “retrievability” of data files on archive service systems. Specifically, some special blocks called “sentinels” are randomly embedded into the data file F for detection purpose, and F is further encrypted to protect the positions of these special blocks. However, like [5], 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.

Shacham et al. [8] design an improved PoR scheme with full proofs of security in the security model defined in [7]. They use publicly verifiable homomorphic authenticators built from BLS signatures , based on which the proofs can be aggregated into a small authenticator value, and public retrievability is achieved. Still, the authors only consider static data files.

Erway et al. [9] was the first to explore constructions for dynamic provable data possession. They extend the PDP model in [4] 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, they eliminate the index information in the “tag” computation in Ateniese's PDP model [4] and employ authenticated skip list data structure to authenticate the tag information of challenged or updated blocks first before the verification procedure. However, the efficiency of their scheme remains unclear.

Shan et al.[11] introduce TPA concept to maintain data integrity and preserve privacy. It reduces online burden and keeps the privacy preserve. Chen et al.[10] gives mechanism for auditing the correctness of data with multiple server.

Ateniese et al. [13] are the first to consider public audit ability in their defined “provable data possession” (PDP) model for ensuring possession of files on untrusted storages. In their scheme, utilize RSA based homomorphic tags for auditing outsourced data, thus public audit ability is achieved. However, Ateniese et al. do not consider the case of dynamic data storage, and the direct extension of their scheme from static data storage to dynamic case may suffer design and security problems.

In their subsequent work [14], Ateniese et al. propose a dynamic version of the prior PDP scheme. However, the system imposes a priori bound on the number of queries and does not support fully dynamic data operations, i.e., it only allows very basic block operations with limited functionality, and block insertions cannot be supported.

In [15], Wang et al. consider dynamic data storage in a distributed scenario, and the proposed challenge-response protocol can both determine the data correctness and locate possible errors. Similar to [7,14], they only consider partial support for dynamic data operation.

Juels et al. [16] describe a “proof of retrievability” (PoR) model, where spot-checking and error correcting codes are used to ensure both “possession” and “retrievability” of data files on archive service systems. Specifically, some special blocks called “sentinels” are randomly embedded into the data file F for detection purpose, and F is further encrypted to protect the positions of these special blocks. However, like [14], 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.

Shacham et al. [17] design an improved POR scheme with full proofs of security in the security model defined in [16]. They use publicly verifiable homomorphic authenticators built from BLS signatures , based on which the proofs can be aggregated into a small authenticator value, and public retrievability is achieved. Still, the authors only consider static data files.

Erway et al. [18] was the first to explore constructions for dynamic provable data possession. They extend the PDP model in [13] 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, they eliminate the index information in the “tag” computation in Ateniese's PDP model [13] and employ authenticated skip list data structure to authenticate the tag information of challenged or updated blocks first before the verification procedure. However, the efficiency of their scheme remains unclear.Shan et al.[19] introduce TPA concept to maintain data integrity and preserve privacy. It reduces online burden and keeps the privacy preserve. Chen et al.[8,20] gives mechanism for auditing the correctness of data with multiple server Frenz et al.[22] introduce a new strategy ,an Oblivious out-sourced storage which is based on Oblivious RAM technique. This idea used to conceal user access pattern and preserve the identity. The existing schemes[21] aim at providing integrity verification for different data storage systems, the problem of supporting both public audit ability and data dynamics has not been fully addressed.

How to achieve a secure and efficient design to seamlessly integrate these two important components for data storage service remains an open challenging task in Cloud Computing. Two basic solutions (i.e., the MAC-based and signature based schemes) for realizing data audit ability and discuss their demerits in supporting public audit ability and data dynamics. Secondly, generalize the support of data dynamics to both proof of retrievability (PoR) and provable data possession (PDP) models and discuss the impact of dynamic data operations on the overall system efficiency both. In particular, emphasize that while dynamic data updates can be performed efficiently in PDP models more efficient protocols need to be designed for the update of the encoded files in PoR models.

III. SYSTEM ANALYSIS

A. Existing System

The existing system introduced a Privacy Preserving in TPA for Secure Cloud using the secured encryption method. The TPA is capable of audit the shared data's integrity without accessing the whole data. Also, it cannot distinguish who the user on each block is. For improving the efficiency, reliability and effectiveness of certifying multiple auditing tasks, we also further drag out our mechanism to support batch auditing.

B. Hardware and Software Requirements

1) Minimum Hardware Requirement

a) System: Core i5 1.80 GHz Processor

b) Hard Disk: 500 GB.

c) Ram: 4 GB.

2) Software Requirement

a) Operating System: Windows 7

b) Technology Used: PHP

c) Database Used : Mysql

IV. SYSTEM ANALYSIS

A. Proposed System Architecture

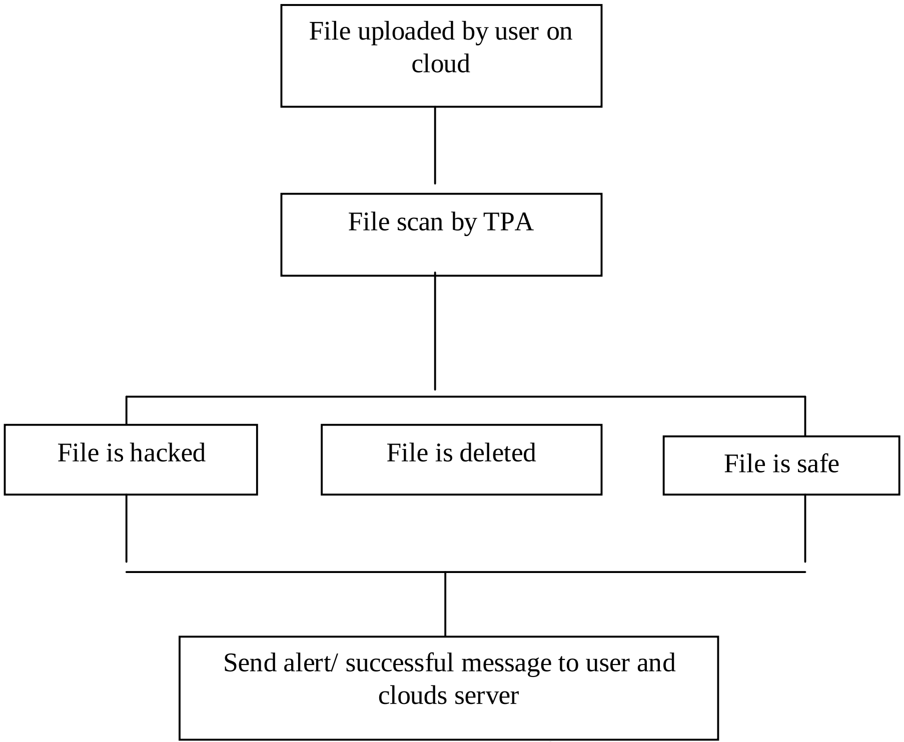
File uploaded by user on cloud
File scan by TPA
File is hacked
File is deleted
File is safe
Send alert/ successful message to user and clouds server

Fig. 2 : Architecture diagram of working system

B. Flowchart

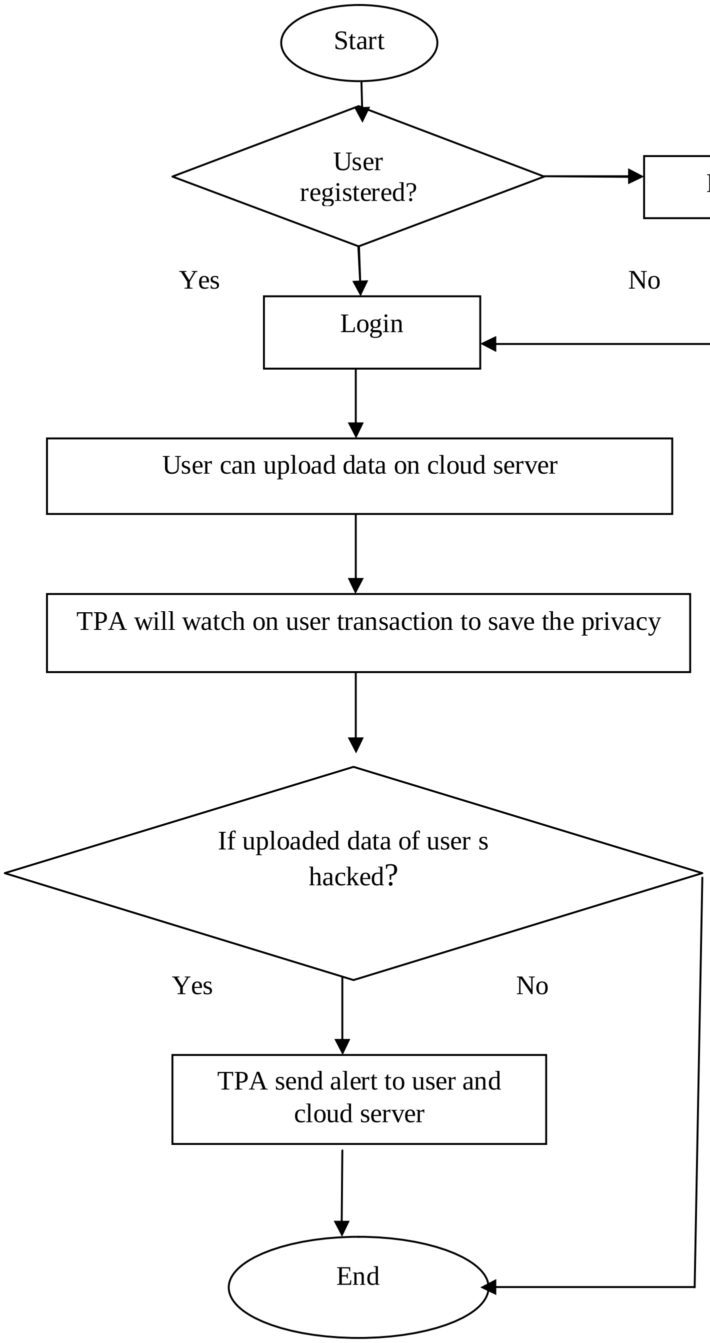
Start
User registered?
Registration
Yes
No
Login
User can upload data on cloud server
TPA will watch on user transaction to save the privacy
If uploaded data of user s hacked?
Yes
No
TPA send alert to user and cloud server
End

Fig. 3 : Flow Chart

C. Proposed Work

The following shows the steps of the implementation.

1) Step 1. Start
2) Step 2. Initialize user login to the dashboard
3) Step 3. Scan user uploaded files
4) Step 4. Check the original file uploaded status
5) Step 5. Check file current status
6) Step 6. Compare original file status with current file status
7) Step 7. If match is found in both files statuses then the file has no harm

8) Step 8. If mismatch is found, the file is misused or corrupt
9) Step 9. If no file is found, the intruder may delete the file from the server.
10) Step 10. According to file status, give status input to the user.
11) Step 11. Stop.

D. For Encryption purpose I used RSA Algorithm

1) *RSA Algorithm:* RSA (Rivest–Shamir–Adleman) is a public-key cryptosystem that is widely used for secure data transmission. It is also one of the oldest. The acronym RSA comes from the surnames of Ron Rivest, Adi Shamir, and Leonard Adleman, who publicly described the algorithm in 1977. An equivalent system was developed secretly, in 1973 at GCHQ (the British signals intelligence agency), by the English mathematician Clifford Cocks. That system was declassified in 1997.

The RSA algorithm is an asymmetric cryptography algorithm; this means that it uses a public key and a private key (i.e., two different, mathematically linked keys). As their names suggest, a public key is shared publicly, while a private key is secret and must not be shared with anyone.

The RSA algorithm is named after those who invented it in 1978: Ron Rivest, Adi Shamir, and Leonard Adleman.

The following illustration highlights how asymmetric cryptography works:

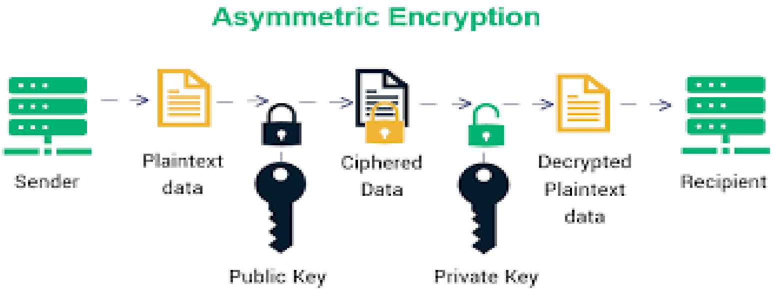

Fig. 4: Asymmetric Encryption

E. How it works

The RSA algorithm ensures that the keys, in the above illustration, are as secure as possible. The following steps highlight how it works:

F. Generating the Keys

1) Select two large prime numbers, x and y. The prime numbers need to be large so that they will be difficult for someone to figure out.
2) Calculate n = x * y.
3) Calculate the totient function; $\phi(n)=(x-1)(y-1)$.
4) Select an integer e, such that e is co-prime to $\phi(n)$ and)$1<e<\phi(n)$. The pair of numbers (n,e) makes up the public key.

G. Encryption

Given a plaintext P, represented as a number, the cipher text C is calculated as:

C = P^{e}mod n.

C. Decryption

Using the private key (n, d) the plaintext can be found using:

P = C^{d} mod n

V. CONCLUSIONS

This research introduced a secure cloud storage base platform in which a user can perform some operations like uploading a new file, access the uploaded file, change the file contents, drop the file and so on. Suppose user uploaded the file on Google cloud ,and an attacker hack that data, the privacy will break. To avoid such problems, this research introduced a concept of public auditing. the TPA is capable to audit the shared data's integrity without accessing the whole data. Also, it cannot distinguish who the user on each block is. For improving the efficiency, reliability and effectiveness of certifying multiple auditing tasks, we also further drag out our mechanism to support batch auditing.

VI. ACKNOWLEDGMENT

It is a matter of great pleasure by getting the opportunity of highlighting a fraction of knowledge we acquired during our technical education through these assignments. This would not be possible without the guidance and help of many people. This is the only page where we have an opportunity of expressing our emotion and gratitude from the care of my heart. These assignments would not have been successful without enlightened ideas timely suggestions and our most respected guide. Dr. V. M. Deshmukh without her best guidance this would have been an impossible task to complete. Being on the same line, we all express our deep sense of gratitude to our heads of the Department Dr. G. R. Bamnote sir for this most valuable guidance provided by him. Last but not least, we would like to express our thankfulness to the teaching and non-teaching staff, our friends and all my well - wishers.

REFERENCES

[1] https://en.wikipedia.org/wiki/Cloud_computing

[2] C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy- Preserving Public Auditing for Storage Security in Cloud Computing,” Proc. IEEE INFOCOM ‟10, Mar. 2010.

[3] P. Mell and T. Grance, “Draft NIST Working Definition of Cloud Computing,” http://csrc.nist.gov/groups/SNS/cloud- computing/index.html, June 2009.

[4] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable Data Possession at Untrusted Stores,” Proc. 14th ACM Conf. Computer and Comm. Security (CCS ‟07), pp. 598-609, 2007.

[5] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, “Scalable and Efficient Provable Data Possession,” Proc. Int‟l Conf. Security and Privacy in Comm. Networks (SecureComm ‟08), pp. 1-10, 2008.

[6] C. Wang, Q. Wang, K. Ren, and W. Lou, “Towards Secure and Dependable Storage Services in Cloud Computing,” IEEE Trans. Service Computing, vol. 5, no. 2, 220-232, Apr.-June 2012.

[7] A. Juels and J. Burton, S. Kaliski, “PORs: Proofs of Retrievability for Large Files,” Proc. ACM Conf. Computer and Comm. Security (CCS ‟07), pp. 584-597, Oct. 2007.

[8] D. Boneh, B. Lynn, and H. Shacham, “Short Signatures from The Weil Pairing,” J. Cryptology, vol. 17, no. 4, pp. 297- 319, 2004.

[9] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia,“Dynamic Provable Data Possession,” Proc. 16th ACM Conf.Computer and Comm. Security (CCS ‟09), 2009.

[10] B. Chen, R. Curtmola, G. Ateniese, and R. Burns, “Remote Data Checking for Network Coding-based Distributed Storage Systems,” in Proc. ACM Cloud Computing Security Workshop (CCSW), 2010, pp.31– 42.

[11] M.A. Shah, M. Baker, J.C. Mogul, and R. Swaminathan, “Auditing to Keep Online Storage Services Honest,” Proc. 11th USENIX Workshop Hot Topics in Operating Systems(HotOS ‟07), pp. 1-6, 2007.

[12] https://en.wikipedia.org/wiki/Merkle_tree.

[13] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable Data Possession at Untrusted Stores,” Proc. 14th ACM Conf. Computer and Comm. Security (CCS ‟07), pp. 598-609, 2007.

[14] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, “Scalable and Efficient Provable Data Possession,” Proc. Int‟l Conf. Security and Privacy in Comm. Networks (SecureComm ‟08), pp. 1-10, 2008.

[15] C. Wang, Q. Wang, K. Ren, and W. Lou, “Towards Secure and Dependable Storage Services in Cloud Computing,” IEEE Trans. Service Computing, vol. 5, no. 2, 220-232, Apr.-June 2012.

[16] A. Juels and J. Burton, S. Kaliski, “PORs: Proofs of Retrievability for Large Files,” Proc. ACM Conf. Computer and Comm. Security (CCS ‟07), pp. 584-597, Oct. 2007.

[17] D. Boneh, B. Lynn, and H. Shacham, “Short Signatures from The Weil Pairing,” J. Cryptology, vol. 17, no. 4, pp. 297- 319, 2004.

[18] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia,“Dynamic Provable Data Possession,” Proc. 16th ACM Conf.Computer and Comm. Security (CCS ‟09), 2009.

[19] M.A. Shah, M. Baker, J.C. Mogul, and R. Swaminathan, “Auditing to Keep Online Storage Services Honest,” Proc. 11th USENIX Workshop Hot Topics in Operating Systems(HotOS ‟07), pp. 1-6, 2007.

[20] B. Chen, R. Curtmola, G. Ateniese, and R. Burns, “Remote Data Checking for Network Coding-based Distributed Storage Systems,” in Proc. ACM Cloud Computing Security Workshop (CCSW), 2010, pp.31– 42.

[21] Salve Bhagyashri , Prof. Y.B.Gurav, “Privacy-Preserving Public Auditing For Secure Cloud Storage,” IOSR Journal of Computer Engineering, Volume 16, Issue 4, Ver. III (Jul – Aug. 2014), PP 33-38.

[22] M. Franz, P. Williams, B. Carbunar, S. Katzenbeisser, and R. Sion, “Oblivious Outsourced Storage with Delegation,” in Proc. Financial Cryptography and Data Security Conference (FC), 2011, pp. 127– 140.