TAXONOMY OF DATA INTEGRITY SCHEMES FOR CLOUD STORAGE SECURITY

A. AnushaPriya [1] and R. Gunasundari [2]

ABSTRACT

Cloud storage is a new cost-effective paradigm that aims at providing high availability, reliability, massive scalability and data sharing. This research article introduces about the cloud computing, several deployment models, privacy issues pertaining to cloud computing and attributes related to data integrity schemes. This paper reviewed around 61 research articles and grouped under respective topics pertaining to data integrity schemes in cloud computing arena. From the reviewed research article we aim to propose a cloud data storage security model that well suits for private cloud environment.

Keywords : Cloud computing, private cloud, Service Level Agreement (SLA), Cloud storage, Cloud service provider, Third party auditor,

1. INTRODUCTION

Cloud computing provides a flexible and cost-effective solution for many services through Internet [1]. It is considered, as a major Information Technology (IT) shift and a latest model of computing over pooled computing resources such as bandwidth, storage, servers, processing power, services, and applications. Today, this new model has gained tremendous popularity and is receiving a lot of attention from researchers in the academic and industrial communities. Essential characteristics of the cloud-computing model include on-demand self-service, rapid elasticity, resource pooling and broad network access. Cloud computing has gained a lot of popularity, which is mainly due to the following reasons, as discussed in [44]: (a) Cloud computing has eliminated the overhead of planning from the user, providing resources that are available on-demand, self-service, and the ability to scale according to requirements. (b) Cloud computing has eliminated up-front commitment by the end users. Pay-as-you-go model has allowed companies to start small and increase their computing resources only when needed.

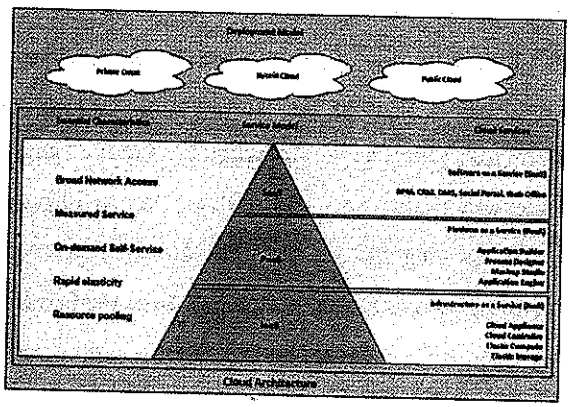

Figure 1. Cloud Architecture

The deployment models used by cloud computing are Private Cloud (internal datacenters of business or organizations which are not available for general public use) [44], Public Cloud (private cloud made

[1] Research Scholar, Department of Computer Science, Karpagam University, Coimbatore - 641 021

[2] Assistant Professor, Department of Information Technology, Karpagam University, Coimbatore - 641 021

accessible for general public use on pay-as-you-go model) [44], Hybrid/Multi-Cloud (cloud computing environment in which an organization provides and manages some resources in-house, and the other services are provided externally) [17]. Cloud computing services are organized as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Under the umbrella of these services a number of new services have also emerged e.g. Data Integrity as a Service(DIaaS) [1], Database as a Service [46-49], Logging as a Service [50, 51, 61], Provenance as a Service [52], Security as a Service [53, 54], Big Data as a Service [57] and Storage as a Service [55, 56]. An overview of cloud architecture is shown in Figure 1.

Cloud Storage as a service is a growing trend with features like elasticity, pay-as-you-go, business continuity with long-term retention and risk mitigation through disaster recovery [1]. All these features are not available with on-premises storage. Popular cloud-based storage services available today are Dropbox, One Drive, Amazon S3, Google Drive, Box, and Sugar Sync etc. Nowadays, to improve business strategies organizations use analysis techniques over their historical data [43]. Some business sectors for instance telecom and e-health have compliance requirements [58], which bind them to keep historical data over a specified period. Not every organization is equipped to manage large secondary storage or build their private data centers (because of the cost associated with building and maintaining such infrastructure). Cloud Storage can be of great service to such organizations because of its flexible model. However, the loss of control is an inherent issue with outsourced data storage model. Although the cloud service provider (CSP) is bounded by a service level agreement (SLA) to ensure data security, users cannot solely rely on such agreements. Furthermore, reliance on a contractual obligation may fail to detect the malicious behavior of the service provider. Cloud computing operational details are not transparent to the customers and the CSP may be untrusted. So besides the convenience provided by cloud model, data security issues such as confidentiality, privacy, and data integrity are also associated with cloud storage service model.

II. Related Works

Data can be manipulated or lost due to accidental or intentional malicious activity, which can be a nightmare for the user and an embarrassment for cloud service provider. Cloud has a provision of "multi-tenancy" i.e. cloud resources will be shared and utilized by multiple users; therefore, adversaries can take advantage of vulnerabilities in the cloud [45]. Administration errors can also damage data; for example, failures in data migration or backup/ restore process. Consequently, data integrity is at the core of outsourcing data over the cloud storage. Many data integrity schemes have been developed for ensuring correctness of outsourced data in recent years. When the integrity of outsourced data is to be checked, it is not feasible to download all data from the remote server and verify as it incurs high

communication and computational cost. To avoid this high cost, most of the data integrity schemes perform blockless verification [2]. This is done without downloading actual data from the cloud, rather it is based on some metadata, which is generated by using original data before it is outsourced. Later on, using this metadata, proofs for integrity verification are generated by cloud service provider and verified by the data owner. Research on data integrity schemes started from work on static data or append-only data and then extended to dynamic data (supporting CRUD operations on block level) [3]. Initially, private (single user/data owner) verifiability was supported and later on, public (multiple users) verifiability [23] was introduced with or without third party auditor (TPA).

Privacy issues such as leakage of data and user anonymity were identified, due to the introduction of TPA. Thus, privacy preserving data integrity schemes [23] were developed to overcome the privacy concerns of the users. Application of these techniques extended from a single copy to multiple copies of data [20], constraint-based data geolocation (CBDG) [33], multi-cloud (cloud of clouds) [29], data deduplication [26] and proof of ownership [27]. Researchers used homomorphic tags, bilinear pairing [39], algebraic signatures [42], fountain codes [34], erasure codes [30], RS codes based on Cauchy matrices [18] and other techniques as metadata for integrity verification. Efficiency being a real challenge in data integrity schemes is measured regarding computation, communication and storage costs incurred. One other desirable characteristic of data integrity is unbounded queries (which mean no restriction on how frequently user can verify data). When data size is large this desirable characteristic is hard to achieve [3]. Achieving efficiency sometimes results in a compromise on security [33]. Implementation primitives also impact computation, communication and storage efficiency. Asymmetric cryptographic primitives provide strong security, but efficiency decreases due to complex computation and large keys. This decline in efficiency becomes more prominent for large datasets.

Disk I/O is another crucial factor affecting the efficiency of the verification process in data integrity schemes. For allowing data sharing over cloud storage, strong data access control policies are needed to ensure data security. The requirements of a data integrity scheme vary according to the deployment environment. Schemes designed for a single copy of data are not applicable within server and cross server replication [4]. The requirements of data integrity scheme change with cloud deployment model e.g. hybrid or multi-cloud [17] whereby the scope of data integrity is extended. Moreover, Service Level Agreement (SLA) checking is becoming a fundamental part of data integrity. Existing surveys [24, 59,60] on data integrity schemes have performed only a comparative analysis based on their characteristics and discussed their internal implementation.

III. Taxonomy of Data Integrity Schemes

The following are the attributes taken into consideration for defining the taxonomy of data integrity schemes. These attributes not only describe the capabilities of any scheme but also give an idea about their limitations in practical scenarios.

Approach: The approach refers to the nature of guarantee provided by a data integrity scheme, which can be deterministic or probabilistic. Deterministic schemes [40,41] need to access the complete file to determine the integrity and give 100% possession guarantee. Whereas, probabilistic schemes [2,3] use randomly chosen blocks of data to verify the integrity and gives less than 100% guarantee. Deterministic schemes are not suitable for large size files e.g. archives having data size in GBs or TBs as integrity verification of such a file takes too much time. This can lead to a limitation on how frequently the user can perform integrity verification. Therefore, probabilistic schemes are more appropriate for large files. On the other hand, deterministic verification is suitable for smaller sized files having the size in few megabytes. Digital libraries (e-books, articles, research papers) and media contents (songs, pictures, etc.,) normally have small size files but are large in number. For such type of data, deterministic schemes are ideal to provide 100% possession guarantees. Ateniese et al. [2] proposed the first probabilistic integrity scheme in 2007 [2], which used random sampling of blocks instead of reading the whole file. By inspecting 4.6% of total blocks of a file, it provides 99% possession guarantee. This scheme provided the base for probabilistic integrity checking schemes, but its performance decreases as the size of data increase.

Nature of Data: This attribute describes the nature of data, on which the scheme is applicable.Data can be of static nature [37] (archival data, backups, or data that is never modified but is appended only) or dynamic nature [38] (that frequently changes due to operations like create, read, update and delete). The Integrity of volatile data is not considered in this survey. The nature of data also plays an import role in scheme evaluation. The scheme proposed in [2], is only suitable for static data, because it requires re-computation of complete file tags when new data is inserted in between existing data. Similarly, the scheme [3] provides support for modification, deletion and appends operations; however insertion operation is not supported. The scheme in [6] provides support for dynamic data operations such as insertion, updation, deletion, and modification. Furthermore, the scheme[6] also kept the fundamental properties of verification (unforgeability, unbounded queries) intact. The use of authenticated skip list based FlexList to support dynamic operations was proposed in [32]. Bowers et al. [8] proposed a POR scheme for static data having low storage overhead and providing higher tolerance rate for errors. The highlighted feature of their proposed design was an incremental encoding of data blocks enabling the support for handling large size datasets. Barsoum et al. [19] extended the integrity checking of static data for multiple copies. The strength of their proposed scheme is that verification time is

independent of the number of copies. Stefanov et al. [22] introduced a framework named "Iris" giving cloud users high assurance of data integrity, data freshness and data recovery in case of any corruption.

Setup: The setup refers to the nature of environment in which data integrity scheme will be deployed. Since data is placed over the cloud, which can be simple (self-sufficient, providing all types of capabilities itself) or "hybrid/cloud of cloud/multi-cloud" (managing some resources internally and other are provided externally) [17], nature of setup leads to some additional requirements in data integrity schemes. For example, a cloud setup can support replication within a server and across servers, extending the integrity checking from single to multiple replicas of files. Curtmola et al. [4] proposed scheme is called "multiple replicas provable data possession scheme (MR-PDP)", which is capable of generating new replicas on demand in case of any corruption or deletion. Application of MR-PDP is limited to a single cloud having multiple storage servers but is not extendable to a multi-cloud environment. Bowers et al. [7] focused on high availability by exploiting the redundancy within a server and across servers. Peterson et al. [15] extended data integrity for verification of SLA to ensure that data is in some specified geographical boundary when outsourced to the cloud. Zhu et al. [28] introduced characteristics of the high security, high performance, and transparent verification for PDP model in the context of multi-cloud.

Tendency: The tendency of a data integrity scheme can be verification only (identification of corruption/ deletion of data) or verification with data recovery (if any corruption is identified). From the perspective of tendency, data integrity schemes are categorized into provable data possession (PDP) or proof of retrievability (POR) [25].

a. Provable Data Possession (PDP): PDP schemes are probabilistic as they use the sampling of random blocks instead of reading the whole file for verification. In PDP, the original data is preprocessed to generate some metadata. This metadata is placed with original data and is used later to verify the integrity of user's data. These schemes can only identify corruption in data but do not support-corrupted data recovery.

b. Proof of Retrievability (POR): POR is very much similar to PDP, but it also provides data recovery. POR schemes use the redundant encoding of data and hence provide recovery in case of failure.

In other words, a PDP scheme can be transformed into POR by using error correcting or erasure codes. The Auditing protocol of POR provides the guarantee that CSP is holding all the data of client and is still retrievable. However, PDP only ensures that the server is holding most of the data of a particular client, due to the probabilistic nature of the algorithm used in verification.

Metadata: All data integrity schemes use some additional metadata along with the original data,

which is utilized in integrity verification process. This metadata can be "tags" [27, 37, 38], having homomorphic verifiable property, which help in generating an aggregated value in the verification process. The metadata may be "signatures" [14, 35], which are used as an alternative to tags, e.g. algebraic signatures to improve techniques like Reed-Solomon codes. Finally, metadata may consist of network codes that also support data recovery instead of tags and signatures, which may be used for identifying small (bit or byte level) data corruptions. For example, RSA-based homomorphic tags are used as verification metadata in [2], "HAIL" [7] scheme provides data recovery capability using integrity-protected error-correcting code (IP-ECC). Chen et al. [10] utilized network codes to provide reduced storage overhead in distributed storage system used for backup and archival data. Okamoto protocol based signatures have been used in the scheme [21] as metadata. Ateniese et al. [11] improve the capability of PDP scheme by introducing robustness using Forward Error-Correcting Codes (FEC).

Encryption: Over the year's data integrity schemes have utilized both symmetric [3, 13] and asymmetric encryption [12,14, 16, 19, 27, 31, 35-37] for security. Ateniese et al. [3] achieved scalability and efficiency by using symmetric key encryption and cryptographic hash functions. Ateniese et al. [2], used RSA based homomorphic verifiable tags (HVT) as the metadata. HVTs are unforgeable and are used in blockless verification by which a server can construct a proof of possession and the client can verify it to ensure data integrity. Curtmola et al. [4] utilized homomorphic tags of Ateniese et al. [2] for multiple replicas and used a Pseudo Random Function (PRF) for masking. Wang et al. [9] proposed a scheme for proof of retrievability using BLS signatures supporting public verifiability along with dynamic data operations. Zhu et al. [28] proposed scheme is based on homomorphic verifiable response (HVR) and hash index hierarchy (HIH) and presents a cooperative PDP (CPDP) scheme from bilinear pairings to provide knowledge soundness. Lee et al. [13] proposed a hybrid scheme using both symmetric and asymmetric encryption for efficient data integrity verification protocol. Wang et al. [23] scheme is based on group signatures and Homomorphic MACs, providing integrity verification of data shared among large user groups.

Auditing: Data integrity schemes either allow public (conducted by a third party auditor or multiple users) auditing or private auditing (performed by data owner). In the case of public auditing, privacy issues such as data leakage, user anonymity etc. are associated. Public auditing schemes are further categorized into privacy preserving and non-privacy preserving schemes. Ateniese et al. [6] proposed a public verifiable scheme, which is called Proof of Storage (POS). However, this scheme was vulnerable to privacy attacks. The same author proposed another POS scheme [13], which provided privacy protection. Krzywiecki et al. [20], scheme utilized secure pseudorandom numbers (SPRN) with Lagrangian interpolation and provides broadcast encryption with public auditability for multiple users. Shacham et al.

[5] have used BLS signatures to provide public auditing. Public auditing with privacy protection is achieved along with traceability, which means that the original user can trace a signature on a block and reveal the identity of the signer.

Another significant contribution of this work is that it supports a large number of users without affecting the performance of the verification process [23]. Shen et al. [12] scheme supports delegation of auditing authority, however, there is a restriction of re-delegation. Therefore, a user authorized for auditing cannot re-delegate the authority to others. Wang et al. [36] have discussed the issues of public auditing and proposed provable secure and efficient proxy provable data possession (PPDP) scheme for cloud storage.

IV. Conclusion

In this research work several research articles are reviewed pertaining to data possession, homomorphic identification, geolocating data, integrity verification, integrity checks, multi cloud storage, retrievability, remote data and much more. From this we came to a conclusion that in the cloud computing arena lot of research scope falls under the security aspect. More importance are given to data possession and security under private cloud scenario, multi cloud storage and remote data storage. We further aim to propose a model for authentication over the cloud data storage.

References

1. Nepal, Surya, et al. *"DIaaS : Data integrity as a service in the cloud."* Cloud Computing (CLOUD), 2011 IEEE International Conference on. IEEE, 2011.

2. Ateniese, G.; Burns, R.; Curtmola, R.; Herring, J.; Kissner, L.; Peterson, Z. & Song, D. (2007), Provable Data Possession at Untrusted Stores, in *'Proceedings of the 14th ACM Conference on Computer and Communications Security'*, ACM, New York, NY, USA, pp. 598-609.

3. Ateniese, G.; Di Pietro, R.; Mancini, L. V. &Tsudik, G. (2008), Scalable and Efficient Provable Data Possession, in *'Proceedings of the 4th International Conference on Security and Privacy in Communication Networks'*, ACM, New York, NY, USA, pp. 9:1—9:10.

4. Curtmola, R.; Khan, O.; Burns, R. &Ateniese, G. (2008), MR-PDP: Multiple-Replica Provable Data Possession, in *'Distributed Computing Systems, 2008. ICDCS '08. The 28th International Conference on'*, pp.411-420.

5. Shacham, H. & Waters, B. (2008), Compact Proofs of Retrievability, in Josef Pieprzyk, ed., *'Advances in Cryptology - ASIACRYPT 2008'*, Springer Berlin Heidelberg, , pp. 90-107.

6. Ateniese, G.; Kamara, S. & Katz, J. (2009), Proofs of Storage from Homomorphic Identification Protocols, in *'Proceedings of the 15th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology'*, Springer-Verlag, Berlin, Heidelberg, pp. 319-333.

7. Bowers, K. D.; Juels, A. &Oprea, A. (2009), HAIL: A High-availability and Integrity Layer for Cloud Storage, in *'Proceedings of the 16th ACM Conference on Computer and Communications Security'*, ACM, New 30 York, NY, USA, pp. 187—198.

8. Bowers, K. D.; Juels, A. &Oprea, A. (2009), Proofs of Retrievability: Theory and Implementation, in *'Proceedings of the 2009 ACM Workshop on Cloud Computing Security'*, ACM, New York, NY, USA, pp. 43— 54.

9. Wang, Q.; Wang, C.; Li, J.; Ren, K. & Lou, W. (2009), Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing, in *'Proceedings of the 14th European Conference on Research in Computer Security'*, Springer-Verlag, Berlin, Heidelberg, pp. 355—370.

10. Chen, B.; Curtmola, R.; Ateniese, G. & Burns, R. (2010), Remote Data Checking for Network Coding-based Distributed Storage Systems, in *'Proceedings of the 2010 ACM Workshop on Cloud Computing Security Workshop'*, ACM, New York, NY, USA, pp. 31-42.

11. Ateniese, G.; Burns, R.; Curtmola, R.; Herring, J.; Khan, O.; Kissner, L.; Peterson, Z. & Song, D. (2011), Remote Data Checking Using Provable Data Possession, ACM Trans. Inf. Syst. Secur. 14(1), 12:1-12:34.

12. Shen, S.-T. &Tzeng, W.-G. (2011), Delegable Provable Data Possession for Remote Data in the Clouds, in *'Proceedings of the 13th International Conference on Information and Communications Security'*, Springer Verlag, Berlin, Heidelberg, pp. 93-111.

13. Lee, N.-Y. & Chang, Y.-K. (2011), Hybrid Provable Data Possession at Untrusted Stores in Cloud Computing, in *'Proceedings of the 2011 IEEE 17th International Conference on Parallel and Distributed Systems'*, IEEE Computer Society, Washington, DC, USA, pp. 638—645.

14. Luo, W. &Bai, G. (2011), Ensuring the data integrity in cloud data storage, in *'Cloud Computing and Intelligence Systems (CCIS), 2011 IEEE International Conference on'*, pp. 240-243.

15. Peterson, Z. N. J.; Gondree, M. & Beverly, R. (2011), A Position Paper on Data Sovereignty: The Importance of Geolocating Data in the Cloud, in *'Proceedings of the 3rd USENIX Conference on Hot Topics in Cloud Computing'*, USENIX Association, Berkeley, CA, USA, pp. 9—9.

16. Zhu, Y.; Wang, H.; Hu, Z.; Ahn, G.-J.; Hu, H. &Yau, S. S. (2011), Dynamic Audit Services for Integrity Verification of Outsourced Storages in Clouds, in *'Proceedings of the 2011 ACM Symposium on Applied Computing'*, ACM, New York, NY, USA, pp. 1550-1557.

17. Zhu, Y.; Hu, H.; Ahn, G.-J.; Han, Y. & Chen, S. (2011), Collaborative integrity verification in hybrid clouds, in *'Collaborative Computing: Networking, Applications and Work sharing (CollaborateCom), 2011 7th International Conference on'*, pp. 191-200.

18. Chen, B. &Curtmola, R. (2012), Robust Dynamic Remote Data Checking for Public Clouds, in *'Proceedings of the 2012 ACM Conference on Computer and Communications Security'*, ACM, New York, NY, USA, pp. 1043-1045.

19. Barsoum, A. F. &Hasan, M. A. (2012), Integrity Verification of Multiple Data Copies over Untrusted Cloud Servers, in *'Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (Ccgrid 2012)'*, IEEE Computer Society, Washington, DC, USA, pp. 829-834.

20. Krzywiecki&Kuty, M. (2012), Proof of Possession for Cloud Storage via Lagrangian Interpolation Techniques, in *'Proceedings of the 6th International Conference on Network and System Security'*, Springer-Verlag, Berlin, Heidelberg, pp. 305-319.

21. Mohan, A. &Katti, R. (2012), Provable Data Possession Using Sigma-protocols, in *'Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on'*, pp. 565-572.

22. Stefanov, E.; van Dijk, M.; Juels, A. &Oprea, A. (2012), Iris: A Scalable Cloud File System with Efficient Integrity Checks, in *'Proceedings of the 28th Annual Computer Security Applications Conference'*, ACM, New York, NY, USA, pp. 229-238.

23. Wang, B.; Li, B. & Li, H. (2012), Knox: Privacy-preserving Auditing for Shared Data with Large Groups in the Cloud, in *'Proceedings of the 10th International Conference on Applied Cryptography and Network Security'*, Springer-Verlag, Berlin, Heidelberg, pp. 507—525.

24. Worku, S.; Ting, Z. &Zhi-Guang, Q. (2012), Survey on Cloud Data Integrity Proof Techniques, in *'Information Security (Asia JCIS), 2012 Seventh Asia Joint Conference on'*, pp. 85-91.

25. Xu, J. & Chang, E.-C. (2012), Towards Efficient Proofs of Retrievability, in *'Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security'*, ACM, New York, NY, USA, pp. 79-80.

26. Zheng, Q. &Xu, S. (2012), Secure and Efficient Proof of Storage with Deduplication, in *'Proceedings of the Second ACM Conference on Data and Application Security and Privacy'*, ACM, New York, NY, USA, pp. 1- 12.

27. Zhu, Y.; Hu, H.; Ahn, G.-J. &Yau, S. S. (2012), Efficient Audit Service Outsourcing for Data Integrity in Clouds, J. Syst. Softw. 85(5), 1083-1095.

28. Zhu, Y.; Hu, H.; Ahn, G.-J. & Yu, M. (2012), Cooperative Provable Data Possession for Integrity Verification in Multicloud Storage, Parallel and Distributed Systems, IEEE Transactions on 23(12), 2231-2244.

29. Cash, D.; Küpçü, A. &Wichs, D. (2013), Dynamic Proofs of Retrievability via Oblivious RAM, in Thomas Johansson &PhongQ. Nguyen, ed., *'Advances in Cryptology – EUROCRYPT 2013'*, Springer Berlin Heidelberg, pp. 279-295.

30. Chen, L. (2013), Using Algebraic Signatures to Check Data Possession in Cloud Storage, Future Gener. Comput. Syst. 29(7), 1709-1715.

31. Chen, L.; Zhou, S.; Huang, X. &Xu, L. (2013), Data Dynamics for Remote Data Possession Checking in Cloud Storage, Comput. Electr. Eng. 39(7), 2413-2424.

32. Esiner, E.; Kachkeev, A.; Braunfeld, S.; Küpçü, A. &Özkasap, Ö. (2013), FlexDPDP: FlexList-based Optimized Dynamic Provable Data Possession, Cryptology ePrint Archive, Report 2013/645, http://eprint.iacr.org/.

33. Rashid, F.; Miri, A. &Woungang, I. (2013), Proof of Retrieval and Ownership Protocols for Enterprise-level Data Deduplication, in *'Proceedings of the 2013 Conference of the Center for Advanced Studies on Collaborative Research'*, IBM Corp., Riverton, NJ, USA, pp. 81—90.

34. Shi, E.; Stefanov, E. &Papamanthou, C. (2013), Practical Dynamic Proofs of Retrievability, in *'Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security'*, ACM, New York, NY, USA, pp. 325—336.

35. Tate, S. R.; Vishwanathan, R. & Everhart, L. (2013), Multi-user Dynamic Proofs of Data Possession Using Trusted Hardware, in *'Proceedings of the Third ACM Conference on Data and Application Security and Privacy'*, ACM, New York, NY, USA, pp. 353-364.

36. Wang, H. (2013), Proxy Provable Data Possession in Public Clouds, Services Computing, IEEE Transactions on 6(4), 551-559.

37. Yuan, J. & Yu, S. (2013), Proofs of Retrievability with Public Verifiability and Constant Communication Cost in Cloud, in *'Proceedings of the 2013 International Workshop on Security in Cloud Computing'*, ACM, 16 New York, NY, USA, pp. 19-26.

38. Zhang, Y. & Blanton, M. (2013), Efficient Dynamic Provable Possession of Remote Data via Balanced Update Trees, in *'Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and 20 Communications Security'*, ACM, New York, NY, USA, pp. 183-194.

39. Wang, H. & Zhang, Y. (2014), On the Knowledge Soundness of a Cooperative Provable Data Possession Scheme in Multicloud Storage, Parallel and Distributed Systems, IEEE Transactions on 25(1), 264-267.

40. Deswarte, Y.; Quisquater, J.-J. &Saïdane, A. (2004), Remote Integrity Checking, in SushilJajodia& Leon Strous, ed., *'Integrity and Internal Control in Information Systems VI', Springer US, pp. 1-11.Gazzoni, E. L.; Luiz, D.; Filho, G.; Sérgio, P.; Barreto, L. M. &Politécnica, E. (2006), 'Demonstrating Data Possession and Uncheatable Data Transfer'.*

41. Han, S.; Liu, S.; Chen, K. &Gu, D. (2014), Proofs of Retrievability Based on MRD Codes, in Xinyi Huang &Jianying Zhou, ed., *'Information Security Practice and Experience'*, Springer International Publishing, , pp. 330-345.

42. Yu, Y.; Ni, J.; Ren, J.; Wu, W.; Chen, L. & Xia, Q. (2014), Improvement of a Remote Data Possession Checking Protocol from Algebraic Signatures, in Xinyi Huang &Jianying Zhou, ed., *'Information Security Practice and Experience'*, Springer International Publishing, pp. 359-372.

43. Yu, Y.; Ni, J.; Ren, J.; Wu, W.; Chen, L. & Xia, Q. (2014), Improvement of a Remote Data Possession Checking Protocol from Algebraic Signatures, in Xinyi Huang &Jianying Zhou, ed., *'Information Security Practice and Experience'*, Springer International Publishing, pp. 359-372.

44. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A. D.; Katz, R. H.; Konwinski, A.; Lee, G.; Patterson, D. A.; Rabkin, A.; Stoica, I. &Zaharia, M. (2009), *'Above the Clouds: A Berkeley View of Cloud Computing'*(UCB/ EECS-2009-28), Technical report, EECS Department, University of California, Berkeley.

45. Ristenpart, T.; Tromer, E.; Shacham, H. & Savage, S. (2009), Hey, You, Get off of My Cloud: Exploring Information Leakage in Third-party Compute Clouds, in *'Proceedings of the 16th ACM Conference on Computer and Communications Security'*, ACM, New York, NY, USA, pp. 199-212.

46. Curino, Carlo et al. *"Relational Cloud: A Database-as-a-Service for the Cloud."* 5th BiennialConference on Innovative Data Systems Research, CIDR 2011, January 1 9-12, 2011 Asilomar, California.

47. ALzain, M. &Pardede, E. (2011), Using Multi Shares for Ensuring Privacy in Database-as-a-Service, in *'System Sciences (HICSS),*

2011 44th Hawaii International Conference on', pp. 1-9.

48. Arasu, A.; Blanas, S.; Eguro, K.; Joglekar, M.; Kaushik, R.; Kossmann, D.; Ramamurthy, R.; Upadhyaya, P. &Venkatesan, R. (2013), Secure Database-as-a-service with Cipherbase, in *'Proceedings of the 2013 ACM 8 SIGMOD International Conference on Management of Data'*, ACM, New York, NY, USA, pp. 1033-1036.

49. Seibold, M. & Kemper, A. (2012), *'Database as a Service'*, Datenbank-Spektrum 12(1), 59-62.

50. Frenot, S. &Ponge, J. (2012), LogOS: An Automatic Logging Framework for Service-Oriented Architectures, in *'Software Engineering and Advanced Applications (SEAA), 2012 38th EUROMICRO Conference on'*, pp. 224-227.

51. Zawoad, S.; Dutta, A. K. &Hasan, R. (2013), SecLaaS: Secure Logging-as-a-service for Cloud Forensics, in *'Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security'*, ACM, New York, NY, USA, pp. 219-230.

52. Hammad, R. & Wu, C.-S. (2014), Provenance as a Service: A Data-centric Approach for Real-Time Monitoring, in *'Big Data (BigData Congress), 2014 IEEE International Congress on'*, pp. 258-265.

53. Al-Aqrabi, H.; Liu, L.; Xu, J.; Hill, R.; Antonopoulos, N. & Zhan, Y. (2012), Investigation of IT Security and Compliance Challenges in Security-as-a-Service for Cloud Computing, in *'Object/Component/ Service-Oriented Real-Time Distributed Computing Workshops (ISORCW), 2012 15th IEEE International Symposium on'*, pp. 124-129.

54. Hussain, M. &Abdulsalam, H. (2011), SECaaS: Security As a Service for Cloud-based Applications, in *'Proceedings of the Second Kuwait Conference on e-Services* and e-Systems', ACM, New York, NY, USA, pp. 8:1—8:4.

55. Calder, B.; Wang, J.; Ogus, A.; Nilakantan, N.; Skjolsvold, A.; McKelvie, S.; Xu, Y.; Srivastav, S.; Wu, J.; Simitci, H.; Haridas, J.; Uddaraju, C.; Khatri, H.; Edwards, A.; Bedekar, V.; Mainali, S.; Abbasi, R.; Agarwal, A.; Haq, M. F. u.; Haq, M. I. u.; Bhardwaj, D.; Dayanand, S.; Adusumilli, A.; McNett, M.; Sankaran, S.; Manivannan, K. &Rigas, L. (2011), Windows Azure Storage: A Highly Available Cloud Storage Service with Strong Consistency, in *'Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles'*, ACM, New York, NY, USA, pp. 143-157.

56. Nabeel, M. &Bertino, E. (2012), Privacy preserving delegated access control in the storage as a service model, in *'Information Reuse and Integration (IRI), 2012 IEEE*

13th International Conference on', pp. 645-652.

57. Zheng, Z.; Zhu, J. &Lyu, M. (2013), Service-Generated Big Data and Big Data-as-a-Service: An Overview, in *'Big Data (BigData Congress), 2013 IEEE International Congress on'*, pp. 403-410.

58. David D. Luxton, Robert A. Kayl, and Matthew C. Mishkind. Telemedicine and e-Health. May 2012, 18(4): 284-288.

59. Oualha, N.; Leneutre, J. &Roudier, Y. (2012), *'Verifying remote data integrity in peer-to-peer data storage:A comprehensive survey of protocols'*, Peer-to-Peer Networking and Applications 5(3), 231-243.

60. Sookhak, M.; Talebian, H.; Ahmed, E.; Gani, A. & Khan, M. K. (2014), *'A review on remote data auditing in single cloud server: Taxonomy and open issues'* Journal of Network and Computer Applications 43(0), 121 - 141.

61. Zawoad, S.; Dutta, A.; Hasan, R., *"Towards Building Forensics Enabled Cloud Through Secure Logging as-a-Service,"* in Dependable and Secure Computing, IEEE Transactions on , vol.PP, no.99, pp.1-1.

Author's Biography

She is working as Assistant Professor, Department of Computer Science in Muthayammal College of Arts & Science. She has completed Mphil degree in computer science in Periyar University. She has the experience 10 years 8 Months. She published 3 papers in various national and international journals. Currently Persuing Ph.D in Karpagam University.

Dr. R. Gunasundari received the Ph.D. Degree in Computer Science from Karpagam University, Coimbatore in 2014. She is working as an Associate Prof & Head in the Department of Information Technology, Karpagam University, Coimbatore. She has published 20 National and 21 International papers in various journals. Her broad field of research is in Data mining.