International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-12, October 2019

Secure Data Replication Management Scheme with Better Data Accessibility

M. B. Jayalekshmi, S.H. Krishnaveni

***Abstract* – Cloud computing technology has gained substantial research interest, due to its remarkable range of services. The major concerns of cloud computing are availability and security. Several security algorithms are presented in the literature for achieving better security and the data availability is increased by utilizing data replicas. However, creation of replicas for all the data is unnecessary and consumes more storage space. Considering this issue, this article presents a Secure Data Replication Management Scheme (SDRMS), which creates replicas by considering the access frequency of the data and the replicas are loaded onto the cloud server by considering the current load of it. This idea balances the load of the cloud server. All the replicas are organized in a tree like structure and the replicas with maximum hit ratio are placed on the first level of the tree to ensure better data accessibility. The performance of the work is satisfactory in terms of data accessibility, storage exploitation, replica allocation and retrieval time.**

***Keywords* – Cloud computing, cloud data availability, data replica, data organization.**

I. INTRODUCTION

Cloud computing is an inevitable technology of today's computing and storage environment, as it offers attractive service and deployment models. The Cloud Service Consumers (CSC) can choose any of the service and deployment models, based on the requirements. This flexibility results in the increased usage of cloud services.

The services being provided by the Cloud Service Provider (CSP) are differentiated into platform, infrastructure, software and storage. The cloud service models encapsulate all the essential functionalities required by a business community. The business community can utilize all or any of the service models provided by the CSP, for which a reasonable payment has to be done. The deployment model is chosen based on the work environment of the CSC.

In spite of all the attractive service and deployment models, the cloud computing paradigm still suffers from certain shortcomings. The major concerns of cloud computing are security and availability. As the private and sensitive data are shared with the CSP, it is necessary to ensure better security to the data.

Though the CSP provides Service Level Agreement (SLA) to the CSC, the security breaches are still observed. Hence, cloud data security is the active research area and several algorithms are being proposed in the literature.

Revised Manuscript Received on October 30, 2019.
* Correspondence Author
M. B. Jayalekshmi, Research Scholar, Department of Computer Science and Engineering, Noorul Islam Centre for Higher Education, Kumaracoil, India
Dr. S.H. Krishnaveni, Associate Professor, Department of Computer Science and Engineering, Baselios Mathews II College of Engineering, Kollam, India. mbjayalekshmi@gmail.com

Data availability is another challenge confronted by cloud computing technology. This term makes sense that the cloud data must be available to all the CSC, irrespective of the count of service requests.

In order to achieve data availability, the cloud data is replicated and stored. The replicas increase the data availability, however it introduces several challenges such as data modification and deletion [1-3]. When the original cloud data is undergone some modifications, the same modifications have to be reflected on the replicas also, which is highly challenging.

The modifications must reflect the replicas immediately, so as to escape from data inconsistency. These prompt modifications are possible only when the replicas are organised by some means. Considering these facts, this work attempts to organize the replicas in tree format by taking the load of the Cloud Server (CS) into account. The highlights of this article are as follows.

- The replicas are not created for all the data, but for data with high access frequency. This idea saves space and time.
- The created replicas are organized by means of a tree like structure, which is formed on considering the CS load.
- The frequently accessed cloud data are stored in the first layer of the tree, which helps in easy localization of the data.

The rest of the content is organized in the following format. Section 2 presents the review of literature of the recent replica management techniques. The proposed replica management approach is presented in section 3. The performance of the proposed approach is evaluated in section 4 and the concluding remarks are presented in section 5.

II. REVIEW OF LITERATURE

This section reviews the related review of literature with respect to replica management in cloud computing.

In [4], a technique to minimize replication based on proactive replica checking is proposed. The objective of this work is to reduce the cloud storage space by creating minimal replicas. This work claims that the reliability is maintained with minimal data replication. However, this work does not follow a concrete idea for minimizing the rate of data replication.

A data replica placement strategy based on genetic algorithm for cloud is proposed in [5]. This work utilizes a tripartite graph model to construct the data replication issue and is tackled by genetic algorithm. However, this work involves computational complexity.

A replica placement algorithm is proposed for decentralized edge computing environment in [6]. This work is based on dynamic creation, replacement and deletion of replicas by considering the data requests.

The shortcoming of this work is that the concept of dynamism introduces several computational overheads.

A QoS aware data replication technique is presented for data intensive applications for the cloud computing system in [7]. This approach presents two different algorithms for dealing with data replication related issues. Initially, the greedy algorithm is applied but the replication cost cannot be reduced. Hence, the data replication issue is converted to Minimum-Cost Maximum-Flow (MCMF) problem. However, this approach suffers from time complexity.

In [8], an optimal replica distribution scheme is proposed for edge node assisted cloud peer to peer environment. The edge nodes are utilized for their increased upload bandwidth and these nodes are employed for handling the replicas. The edge nodes deal with the user request. However, this work consumes more resources of cloud.

The identity based public multi-replica provable data possession is proposed in [9]. This work presents a third-party verification for outsourced data that has multiple replicas in the absence of Public Key Infrastructure (PKI). However, this work does not concern about the data organization and any strategy for creating replicas.

In [10], a work is proposed to reduce the joint response time for peer-assisted cloud storage systems. The varying type and size of data makes it tough to proceed with replicas being stored in various locations. This work considers a new metric namely joint response time that considers the delay in queue and service time. However, this work does not involve replica organization.

A new model to check for the consistency of the data namely Consistency as a Service (CaaS) is presented in [11]. This work is based on two-levelled auditing architecture that needs a loosely synchronized clock of the audit cloud. The algorithms to verify the metrics such as commonality and read staleness are proposed.

In [12], a replication based load balancing technique is proposed. This approach creates multiple replicas for each and every job and the replica is stored in different servers. This work considers different queue length and load conditions. However, this work focus only on the delays and not the replica based operations. A data replica placement mechanism meant for open heterogeneous storage system is proposed in [13].

This strategy takes the data availability, count of replicas and the load is balanced. However, this work does not consider the data based operations and the organization. In [14], a twin layered security scheme for cloud storage is presented to preserve the data integrity.

Motivated by these works, this work intends to present a replica management scheme that organizes the replica through which the data based operations are performed effortlessly. To our knowledge, most of the existing works does not organize the replicas and the data based operations are not carried out.

This work focuses on time and performance efficiency by creating replicas for the data with more demand. This idea saves space and time. In addition to this, the so created replicas are organised in a tree like format, which results in easy management of replicas while performing data based operations.

III. PROPOSED SECURE DATA REPLICATION MANAGEMENT SCHEME (SDRMS)

This section presents the proposed secure data replication management scheme in addition to the general flow of the work.

3.1 General Idea of the Work

Data is the pre-requisite of any task and hence, it has to be managed in a secure fashion. As all the simple process yields some beneficial data, it is difficult for the data owners to store and manage the available data. Additionally, the data is growing exponentially, which makes it tough for the data owners to provide room for it.

Cloud computing is a boon for the mid-scale industries, as it provides storage space for nominal charge. Though the cloud offers storage space to the data owners, the data owners hang-back to share the treasured data. Security is one of the major drawbacks of cloud, which makes the data owners to think thrice before outsourcing the data. However, the security aspect is presented in our previous works. The overall flow of this work is depicted in figure 1.

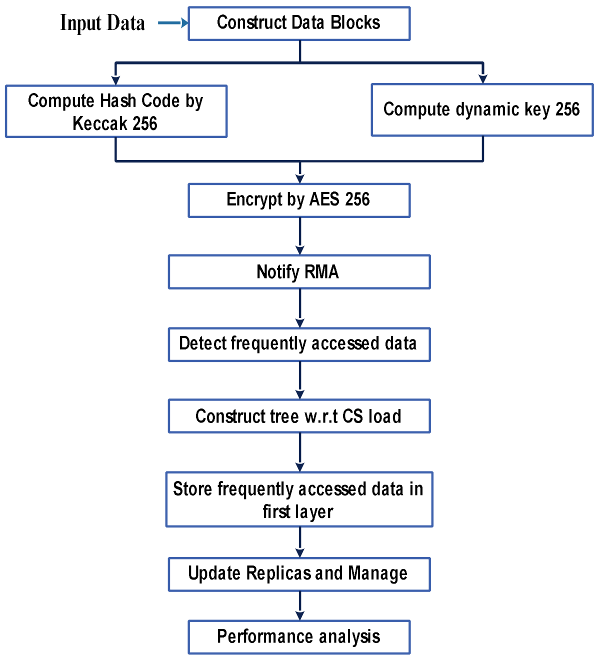

Fig.1. General flow of the work

Data availability is another crucial challenge of cloud computing, as it follows the distributed architecture. Hence in order to boost up the data availability, data replicas are created and managed. Though replicas are beneficial for increasing the data accessibility, it is not easy to manage the replicas. The major difficulty in replica management is the replica up gradation. For instance, whenever the original data is modified or deleted, the data replica must reflect the changes made in the original data. Failing to do this, data inconsistency comes into picture.

On the contrary, it is not a good idea to create replicas for all the data unnecessarily. The reason is the unwanted memory allocation. Taking these issues into account, this work attempts to present a Secure Data Replication Management Scheme (SDRMS), which creates replicas in line with the access ratio of the data.

Security is provided to the data replicas by means of TSS and the Replica Management Agent (RMA) takes care of the replicas. The RMA requests for the current load of the servers and detects the data items with greater access rates. The RMA forms a tree like structure by taking the load into account. In each level of the tree, the cloud server is sorted in ascending order with respect to the load. This makes sense that the first level of the tree contains cloud nodes with minimal load. Hence, all the replicas are loaded to the first level of nodes. Additionally, as the cloud servers are organized by some means, it is easy to update the replica as well. The locations of all the replicas and the cloud server are maintained by the RMA.

3.2 Role of RMA

As the cloud follows distributed architecture, different RMAs are employed with respect to the geographical location. Each RMA controls multiple Cloud Servers (CS) and the CS reports the RMA then and there. The RMA is responsible for creating replicas on the basis of the access ratio of the data.

In addition to this, the current load of the CS is checked periodically by the RMA. This periodical check-up is to avoid over-burdening the CS with more data. When the CS is overloaded with data, the performance of the CS goes down. Hence, the RMA considers the CS load for loading the replicas.

On considering both the CS load and data access ratio, the replicas are created and loaded on to the CS. In addition to this, the RMA forms a tree-like structure, such that the first layer of the tree is loaded with the CS with minimal load. Based on this principle, the replicas are preferably placed in the CS of the first layer.

Hence, whenever the original data is modified, the RMA is reported. As the replicas are placed on the first level of the tree, it is easy for the RMA to upgrade the replicas.

3.3 Detection of Frequently Accessed Data

The cloud contains voluminous data and it is not fair to create replicas for all the data, as all the data are not equally important. Replicas must be created for the important and frequently accessed data. This idea saves memory and increases the overall efficiency of the replica management system.

The data accessing tendency of humans vary from time to time and hence, the RMA checks for the frequently accessed data on each CS for every period of time.

In this case, the data access frequency checking interval for an RMA is set as 300 seconds. Here, there are two possible cases and they are change in access frequency and no-change in access frequency.

- Change in access frequency: In case, when some other data is accessed other than the previous frequently accessed data, the replicas are moved on to the second layer. Based on the access demand, the replicas are determined places.
- No-change in access frequency: In this case, the previous frequently accessed data maintains the same demand and it retains the same status with no change.

The frequently accessed data is detected by the following formula.

$$Freq_{adata} = \frac{Request\ for\ data_i}{Total\ number\ of\ requests} \qquad (1)$$

When the $data_i$ is the most frequently accessed data out of all data requests, then obviously $data_i$ is the frequently accessed data. The value of $Freq_{adata}$ ranges from 0 to 1 and this $Freq_{adata}$ is arranged in descending order, such that the frequently accessed data comes first. The RMA requests for $Freq_{adata}$ and the current CS load for constructing the tree like structure and is explained in the following sub-section.

3.4 Construction of Tree-like Structure and Replica Storage

Based on the information that includes CS load and access frequency of the data, a tree-like structure is built by the RMA. This idea enables the process of loading the replicas easily and is makes the process of up-gradation easier too. The below presented figure 2 presents the replica placement strategy.

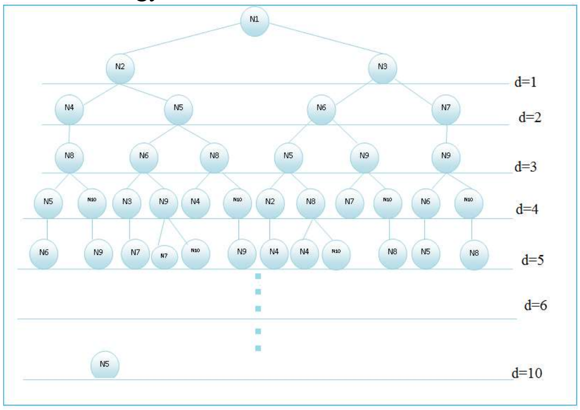

Fig.2. Sample replica placement strategy

In figure 2, N1 is the RMA. As presented in the figure, it is quite reasonable for any cloud server to be loaded with more data. On receiving current load from the CS, the RMA arranges the load in ascending order, such that the CS with minimal load comes first. It is obvious that the CSs with minimal load are minimal for sure and these CS are given place in the first layer. Similarly based on the current load of the CS, all the CSs are given place in the tree-like structure. As observed in figure 2, as the count of layers increase the CS with maximal load are present and is a normal case in cloud. As soon as the tree is constructed, the replicas are loaded on to the CSs present in the first layer. Suppose after loading so many replicas, when the load of CS in the first layer becomes greater than the load of CS in the second layer, both the CSs are swapped.

By this way, the replicas are placed on the CS. Once the original data is upgraded, the replicas can easily be located and upgraded as well. The RMA stores the location of replica as well for easy up-gradation in a separate file, which is conceived on the basis of the catalogue file [15]. The performance of the proposed work is evaluated in the following section.

IV. RESULTS AND DISCUSSION

The performance of this work is tested and the results of this work are compared with the analogous techniques [7,16]. The replica allocation system for cloud computing is scarce in the existing literature. This work provides both security and the replica management, which is an added advantage. The performance of this work is evaluated in terms of data accessibility, replica allocation time, replica retrieval time and storage exploitation analysis.

$$Data_{accessibility} = \frac{Total\ number\ of\ service\ grants}{Total\ number\ of\ service\ requests} \quad (2)$$

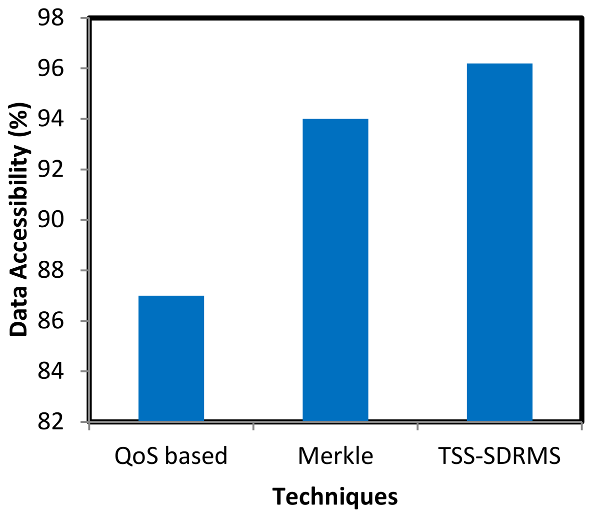

Fig.3. Data accessibility analysis

Better data accessibility is the most important objective of creating replicas. Creation of replicas helps in accessing the required data in a reasonable amount of time and this functionality has to be checked in the replica based schemes. Data accessibility is checked by considering the total count service grants to the total number of service requests and the performance of the proposed approach is analysed with the existing approaches.

On analysis, it is observed that the data accessibility of the proposed approach is better with greater data accessibility rate. The main reason for the improved data accessibility of the proposed work is the core idea of replica creation based on the demand of the data. In addition to this, the replicas are organised which helps in achieving easy access to the data.

On the other hand, QoS based approach employs MCMF pattern, which results in increased time complexity due to the poor data accessibility. The merkle based technique employs tree like format for each replica, which leads to computational complexity and reduces the data accessibility rate. The following graph presents the replica allocation time.

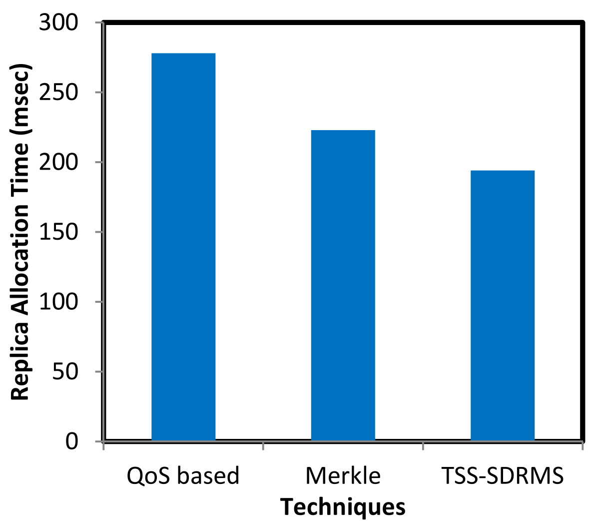

Fig.4. Replica allocation time analysis

The replica allocation time is computed by considering the creation of replicas and the allotment to the service requests is measured. On analysis, the proposed approach consumes minimal time for allotting the replicas.

The main reason for the minimal time consumption is that the proposed approach does not create replicas for all the data, but the data with greater access frequency. This idea supports in allotting the replica faster than the existing approaches.

The QoS based approach does not include any replica organization technique and the merkle approach forms a tree for all the data, which leads to complexity in replica allocation. As the proposed approach employs RMA based on location, the tree can easily be accessed and allotted.

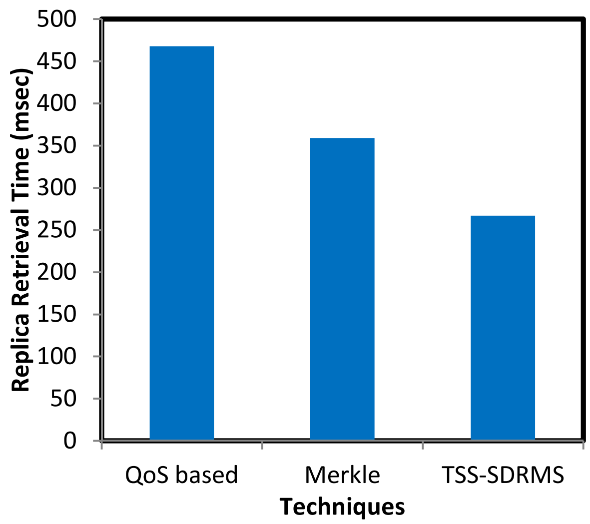

Fig.5. Replica retrieval time analysis

The replica retrieval time denotes the time it takes to retrieve the replica from the cloud server. Obviously, the QoS based approach consumes more time, as it does not follow any replica organization technique for storing replicas.

On the other hand, merkle approach follows tree like structure and the sub-tree is meant for every single replica. This makes the replica retrieval complex and the proposed approach maintains the frequently accessed replicas on the first level of the tree. This idea helps in easy localization of the replicas and hence can be retrieved in no time. The following graph presents the storage exploitation analysis of the proposed approach.

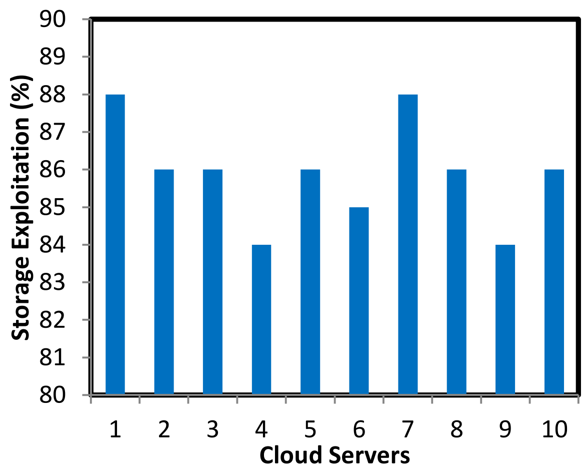

Fig.6. Storage exploitation analysis

Analysis on storage exploitation is one of the most important factors of any replica management system.
The reason behind the fact is that no single cloud server should be overburdened by placing more number of replicas. The proposed approach is much concerned about achieving better balance of load and hence, the cloud servers are sorted on the basis of load after which the replicas are placed. On analysis, it is noted that all the cloud servers are loaded in a balanced fashion and not overburdened. Hence, the proposed approach proves better results in terms of storage exploitation, data accessibility, replica retrieval and allocation time.

V. CONCLUSIONS

This paper proposes a Secure Data Replication Management Scheme (SDRMS) for cloud data, which relies on Twin layered Security Scheme (TSS) and Replica Management Agent (RMA). The TSS is meant for encrypting the data and the replica management agent creates replicas based on the access frequency of the data. The replicas are stored in the cloud server by considering the current load of the server. This idea conserves space, as the replicas are not created for all the data. In addition to this, the proposed approach achieves better load balance, such that all the cloud servers are utilized effectively. The replicas are organised in a tree like structure and the frequently accessed replicas are placed on the first level of the tree, which increases the ease of data retrieval. In future, this work is planned to consider a real-time scenario for attaining better data availability and security as well.

REFERENCES

1. N. R. Herbst, S. Kounev, and R. Reussner, “Elasticity in cloud computing: What it is, and what it is not,” in Proceedings of the 10th International Conference on Autonomic Computing (ICAC 2013), San Jose, CA, 2013.
2. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica et al., “A view of cloud computing,” Communications of the ACM, vol. 53, no. 4, pp. 50–58, 2010.
3. M. Al-Ayyoub, Y. Jararweh, M. Daraghmeh, and Q. Althebyan, “Multi-agent based dynamic resource provisioning and monitoring for cloud computing systems infrastructure,” Cluster Computing, vol. 18, no. 2, pp. 919–932, 2015.
4. Wenhao Li ; Yun Yang ; Dong Yuan, "Ensuring Cloud Data Reliability with Minimum Replication by Proactive Replica Checking", IEEE Transactions on Computers, Vol.65, No.5, pp.1494-1506, 2016.
5. Lizhen Cui ; Junhua Zhang ; Lingxi Yue ; Yuliang Shi ; Hui Li ; Dong Yuan, "A Genetic Algorithm Based Data Replica Placement Strategy for Scientific Applications in Clouds", IEEE Transactions on Services Computing, Vol.11, No.4, pp.727-739, 2018.
6. Atakan Aral ; Tolga Ovatman, "A Decentralized Replica Placement Algorithm for Edge Computing", IEEE Transactions on Network and Service Management, Vol.15, No.2, pp.516-529, 2018.
7. Jenn-Wei Lin ; Chien-Hung Chen ; J. Morris Chang, "QoS-Aware Data Replication for Data-Intensive Applications in Cloud Computing Systems", IEEE Transactions on Cloud Computing, Vol.1, No.1, pp.101-115, 2013.
8. Wei Zhao, Jiajia Liu, Takahiro Hara, "Optimal Replica Distribution in Edge-Node-Assisted Cloud-P2P Platforms for Real-Time Streaming", IEEE Transactions on Vehicular Technology, Vol.67, No.9, pp.8637-8646, 2018.
9. Su Peng ; Fucai Zhou ; Qiang Wang ; Zifeng Xu ; Jian Xu, "Identity-Based Public Multi-Replica Provable Data Possession", IEEE Access, Vol.5, pp.26990-27001, 2017.
10. Ming-Hung Chen ; Yu-Chih Tung ; Shih-Hao Hung ; Kate Ching-Ju Lin ; Cheng-Fu Chou, "Availability Is Not Enough: Minimizing Joint Response Time in Peer-Assisted Cloud Storage Systems", IEEE Systems Journal, Vol.10, No.4, pp. 1424-1434, 2016.
11. Qin Liu ; Guojun Wang ; Jie Wu, "Consistency as a Service: Auditing Cloud Consistency", IEEE Transactions on Network and Service Management, Vol.11, No.1, pp.25-35, 2014.
12. Amir Nahir ; Ariel Orda ; Danny Raz, "Replication-Based Load Balancing", IEEE Transactions on Parallel and Distributed Systems, Vol. 27, No.2, pp. 494-507, 2016.
13. X.Xu, C. Yang, J.Shao, "Data Replica Placement Mechanism for Open Heterogeneous Storage Systems", Procedia Computer Science, Vol.109, pp. 18-25, 2017.
14. M. B. Jayalekshmi, Dr. S. H. Krishnaveni, "TSS – Twin Layered Security Scheme for Cloud Storage to Preserve Data Integrity", International Journal of Intelligent Engineering and Systems, Vol.10, No.3, pp. 94-101, 2017.
15. M. B. Jayalekshmi, Dr. S. H. Krishnaveni, "A Secure Storage Model for Cloud Computing Environment by Employing Distribution Agent", International Arab Journal of Information Technology, Article in Review, 2018.
16. Chang Liu ; Rajiv Ranjan ; Chi Yang ; Xuyun Zhang ; Lizhe Wang ; Jinjun Chen, "MuR-DPA: Top-Down Levelled Multi-Replica Merkle Hash Tree Based Secure Public Auditing for Dynamic Big Data Storage on Cloud", IEEE Transactions on Computers, Vol.64, No.9, pp.2609-2622, 2015.