Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.776977

AN ADVANCED KEY BASED CLOUD SECURITY (AKBCS) USING THREE-WAY SECURITY WITH RSA

[1]Ramanjaiah Ganji, [2]Dr.Suresh Babu Yalavarthi
[1]Department of Computer Science and Engineering, Research Scholar, Acharya Nagarjuna University, Guntur, Andhra Pradesh, India.
[2]Professor, Department of Computer Science, J.K.C College, Guntur, Andhra Pradesh, India.

Abstract: Cloud computing is the domain that provides data storage, data control, and data access, irrespective of software and hardware. Many organizations are updated their storage capacity to store and process the data using several algorithms. Data protection in cloud computing is the primary factoring that protects the data from various attackers. Many solutions are introduced to provide security for the data stored in the cloud; accessing the data with the encryption key offers better protection for the cloud data. Another issue identified in cloud computing is load balancing. If the user requests are more than the threshold value, the load balancing issue occurs in this scenario. This paper describes an advanced key based cloud security (AKBCS) introduced to store and secures the data in cloud storage using three-way security systems. RSA encryption and decryption algorithm used to secure data and large files. Compared with other security algorithms, RSA is the most powerful algorithm that processes large data files in cloud storage.

Keywords: Cloud computing, Key management, RSA encryption and Decryption.

Introduction

Cloud computing mainly focuses on protecting the data stored by the users. Cloud domain extracted from the distributed environment. Over the internet, cloud technology provides online services hosted in a cloud server. In information technology, cloud services are available to new users to access various services using communities and markets [1]. Data centers have maintained cloud services in multiple places in the world. Google applications and Sharepoint developed by Microsoft.

In cloud storage, security plays a significant role in cloud services [2]. Several existing models focus on various security solutions, including technology and access policies. New attacks are introduced in the cloud platform to damage the server without recovery, considered malicious attacks. The proposed model introduced the latest solutions to detect and prevent attacks from protecting the cloud. Based on the cloud attributes, various security issues affect the cloud environment. Our research introduced a new security model, which combines all the users and is interconnected with the cloud server, and provides better cloud services to the users without data loss. In this paper, cloud security is based on three types such as integrity, confidentiality, availability, and accessibility. All these security goals are used to prevent threats and maintain data confidentiality based on encryption and decryption, which provides more security. On the other side, data leakage and load balancing is other security threat affecting cloud storage [4].

Integrated cloud security mainly focuses on securing the data from any attack and maintaining the confidentiality of the cloud environment. Connecting the data using encryption and decryption gives better protection with advanced key generation. To generate a key, an

advanced random secret key combined with the digits and words gives robust security for the cloud data. Later, the cloud operators combined all these features and provided cloud services. This paper focused on developing three-way security for the data available in cloud storage. We have developed an online file storage application that stores the data using cloud storage. Here, the data is stored by the data owner, and a secret key is generated for the data file uploaded by the data owner. The user searched for the required data and sent a request to the data owner. Now the data owner will send permission to the user by sending a secret key to decrypt the file. The proposed model is the three-way security in which every user protects the required file.

Literature Survey

Gonzales et al. [5] proposed a new cloud system that combines various security controls and increases cloud security to provide trust-based high-level security. The proposed cloud trust model combined with CSP provides confidentiality and integrity to the cloud data storage. Jamshidi et al. [6] introduced a new model that proposed cloud migration to use on-demand applications. The proposed cloud migration helps to provide security for cloud storage and presents the trust for the cloud data. Tysowski et al. [7] proposed a new encryption model that allows only authorized users to access the data from the cloud based on the attributes like high computation from encrypted methods, which assign the cloud provider with low communication cost. The proposed model reduces the load on a cloud server and provides on-demand services. Li et al. [8] proposed a new framework that provides security for health management systems. An acceptable-grained approach with encryption is used for the data files consisting of patient health reports. The proposed model in this approach uses the multiple data owner aspect, which classifies the users in the health care system and increases security by decreasing the use of key management for data owners and users. Li et al. [9] proposed model FindU is a new privacy model that provides the privacy for the profiles present in the online social networking sites (OSNS). The FindU provides the similar users that matches the profile and reduces the information to the users based on the attributes. The levels of privacy are provided to the user profiles called as secure multi-party computation (SMC) techniques proposes the privacy levels of users. H. Deng et al. [10] proposed an identity-based encryption transformation (IBET) approach by combining several multiple schemes such as IBE and IBBE to provide the better security model for the distributed systems. Among IBE and IBET, the IBET uses the bilinear groups and improve the security to prevent the powerful attacks. Lee et al. [11] introduced an IBR model which gives the rapid encryption by merging the subset difference (SD) technique with the SRE model. The IBR security model depends on the SRE model. To measure the performance these two schemes are utilized. W. Treesinthuros., [12] proposed the security model integrated in the cloud E-commerce applications. The security model focused on finding the malicious user in the cloud E-commerce application. Jung et al. [13] introduced the unsigned cloud access control that focuses on data privacy in cloud. The proposed anonymous control divide the access control and prevent the data leakage to achieve the semi anonymity. Hossein et al. [14] proposed the Lossless Compression algorithm (LCA) that provides the security for the data present in cloud storage. The LCA focused on improving the performance of cloud storage by modifying the RSA to overcome the encryption issues. W. Yang et al., [15] proposed the TRQED that supports the rapid searching of data over cloud

provides the high security for cloud data. The TRQED model integrated with the safest range query model called as TRQED+ which provides high security compare with TRQED.
N. H. Sultan et al. [16] proposed the Role-Based Encryption (RBE) used to encrypt the cloud data by combining the role-based access control (RBAC) model. The proposed RBAC mainly focuses on the revocation of users for cloud service providers. Authorized users of the specific organization use the data stored in the cloud. To reduce the overhead in the cloud, outsourced decryption is integrated with the cryptographic functionalities in the cloud. C. Esposito et al. [17] proposed a new cloud computing model that develops the domains more capable of solving the issues in the cloud. Y. Xie et al. [18] proposed a unique model which provides safe and secure functionality to access the data by authorized users. The proposed model works on mobile cloud computing platforms which can provide advanced security based on the prevention methods used by the author. H. Li et al. [19] proposed an effective multi-keyword searching model that adequately protects cloud data storage. The proposed approach focused on analyzing the security levels based on the functionality of the users. Z. Xiao et al. [20] provide a unique security model that provides efficient security for the data stored in the cloud.

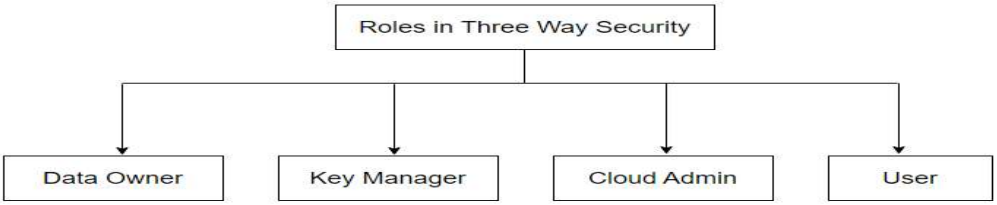

Figure 1: Functionalities of Three Way Security

Keyword Searching

In this step, the user wants to search for the encrypted file uploaded by the data owner. The permission to the user should be given by the data owner, key manager, and cloud admin. In this paper, keyword searching follows the linear searching process to increase search performance. In this process, the files are identified sequentially in the list. Based on the sorted list, this method is performed. The search starts from '0' and continues till the required file is found. The computation time is also significantly less than 1 Sec, and the file list is placed in the array.

Algorithms Steps:

Step 1: Keyword Entered based on the required file.
Step 2: matchedfile(file[], pattern[]){
//The file size is n and m is the pattern size
Pattern for (x=0; x<n; x++){
For (y=0; y<m&&x+y<n; y++)
If(file[x + y]!=pattern[y] break;
//mismatch found, break the inner loop if(y==m)//match found
}}
}

Calculation of Load Imbalance Factor

In this section, the load balancing is calculated by using the following equations. The load balancing plays the major role in maintaining the quality of service (QoS) provided by the

cloud service providers. The virtual machines (VM) used to maintain the load balancing at the server level, because the overall requests are received and managed by the VM's only.

In given virtual machines the sums of loads are as follows:

$$L = \sum_{i=1}^{k} l_i \quad --(1)$$

The total number of VM's in the data centre represented as i. The capacity of load per unit is defined as:

$$LPC = \frac{L}{\sum_{i=1}^{m} c_i} \quad --(2)$$

Threshold $T_i = LPC * c_i$-- (3)

c_i- represented as capacity of the node.

Each virtual machine load imbalance is given by

$$\text{If VM} \begin{cases} < \left| T_i - \sum_{v=1}^{k} L_v \right|, \text{Underloaded} \quad --\text{Case}(1) \\ < \left| T_i - \sum_{v=2}^{k} L_v \right|, \text{Overloaded} \quad --\text{Case}(2) \\ < \left| T_i - \sum_{v==1}^{k} l_i \right|, \text{Balanced} ----\text{Case}(3) \end{cases}$$

The task are load balancing should be migrated from the overloaded VM to under loaded VM should be allowed until the load on the overloaded VM drops below the threshold and the difference is represented as μi. The load balancing of VM's are calculated by using case-1, 2, 3. In case-1 the under loaded is initialized, case-2 represents the overloaded and case-3 initialized the balanced and load in VM improved by showing the threshold difference is λj as shown below. The transfer of overloaded VM to under loaded VM is carried out until its load is less than the threshold. The under loaded VM can accept load only up to its threshold, thus avoiding it being overloaded. This implies that the amount of load that can be transferred from the under loaded VM should be in the range of μ and λ.

Working of RSA

Public Key PR: n: two prime numbers a and b initialized as; e: the mutually prime numbers represent as (a - 1) and (b - 1).

Private Key PU: n: A and B represent A and B.

$$\mathrm{d} = \mathrm{e}^{-1}(\mathrm{mod}(\mathrm{a}-1)(\mathrm{b}-1)$$

Encrypt $\mathrm{c} = \mathrm{m}^{\mathrm{e}}\mathrm{mod}n$

Decrypt: $\mathrm{c}^{\mathrm{d}} = \mathrm{mod}n$

The process of RSA algorithm is as follows:

(1) Initialize two integers with long digits;

(2) By using the equation n = a*b; the 'n' value is calculated;

(3) Measure (a - 1) * (b - 1), whose result is calculated as f (n) (a, b are impenetrable);

(4) Initialize an integer e > 1, which should be less than (a-1) * (q-1) and correlative integer;

(5) By using this equation, d*e≡1mod f(n),'e' and f(n) are known; 'd' is identified. mod is a remainder operation, the remainder value of 'B' to 'C' is equal to 1;
(6) 'n' and 'e' are initialized as public keys, and 'n' and 'd' are initialized as private keys;
(7) The encryption function is given as C≡M^e modn, segmentation function is required, if the length is long.
(8) The decryption function is given as M≡C^d modn, the relationship between plaintext length and key length in the RSA algorithm is explained in this section.
Here, 'e' and 'n' are public keys and these are utilized for encryption. However, d is not extracted from 'e' and 'n'.

Experimental Results

The experiments are conducted by using python programming language. Totally 100 files are used for encryption and decryption and all these files ate text files with the different sizes. These files are synthetic files collected from online sources.

Performance Metrics

The performance metrics shows the stability and reliability of the model. Here the performance is measured by using the following metrics such as data searching time, data encryption time, data decryption time. The encryption time measures at the data owner's level. Decryption time measures at user's level. The following formulas used to measure the performance of proposed model.

Data searching time (DST): DST measures the overall computation time for retrieving the required files from the cloud server.

$$\mathrm{DST} = \mathrm{Ending}_{\mathrm{time}} - \mathrm{initial}_{\mathrm{time}}$$

Data Encryption time (DET): DET measures the encryption time for every file uploaded by the data owner.

$$\mathrm{DET} = \frac{\text{Data Size (KB)}}{\text{Speed (Sp)}}$$

Data Description time (DDT): DDT measures the decryption time for every file downloaded by the user.

$$\mathrm{DDT} = \frac{\text{Data Size (KB)}}{\text{Speed (Sp)}}$$

The comparison between three traditional security algorithms with the proposed RSA algorithm is implemented with different sizes of data files.

Table 1: Measuring the time for searching the file, encryption and decryption using AES (256-bit)

File Size (KB)	Searching Time (MS)	DET (MS)	DDT (MS)
Textfile-1 (12 KB)	123	145	144
Textfile-2 (24 MB)	1345	1578	1698
Textfile-3 (33 MB)	2189	2467	2687
Textfile-4 (56 MB)	3256	3576	3645

Textfile-5 (123 MB)	8234	7234	7156

Table 1 shows the result analysis for the files uploaded at the data owner level and shows the file searching time (ms) and encryption and decryption using AES (256-bit). Experiments are conducted on 100 files; here we have shown the result analysis of 5 files. Based on the size of the file the searching time is increased gradually, and also decryption time and encryption time.

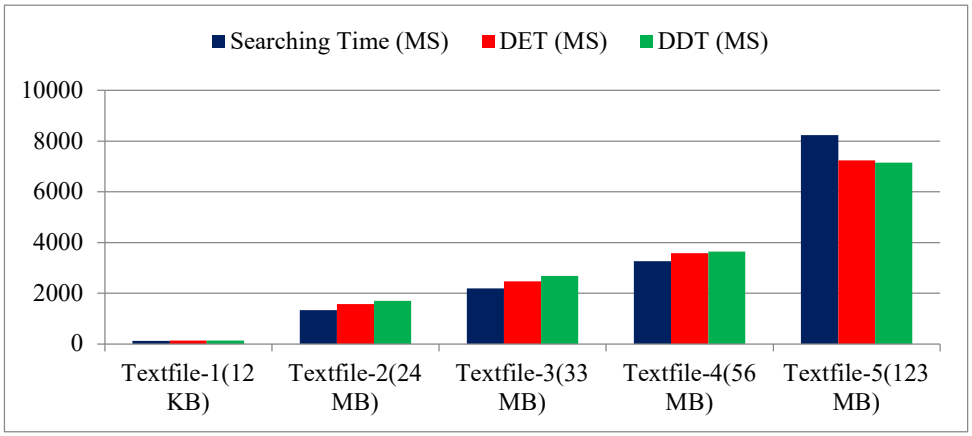

Figure 3: Performance of AES (256-bit) in terms of searching the file, encryption and decryption

Table 2: Measuring the time for searching the file, encryption and decryption using 3DES (256-bit)

File Size (KB)	Searching Time (MS)	DET (MS)	DDT (MS)
Textfile-1 (12 KB)	112	130	130
Textfile-2 (24 MB)	1298	1478	1589
Textfile-3 (33 MB)	2078	2378	2578
Textfile-4 (56 MB)	3176	3432	3489
Textfile-5 (123 MB)	8197	7023	6987

Table 2 shows the result analysis for the files uploaded at the data owner level and shows the file searching time (ms) and encryption and decryption using 3DES (256-bit). Experiments are conducted on 100 files; here we have shown the result analysis of 5 files. Based on the size of the file the searching time is increased gradually, and also decryption time and encryption time. Compare with AES the 3DES shows the high performance.

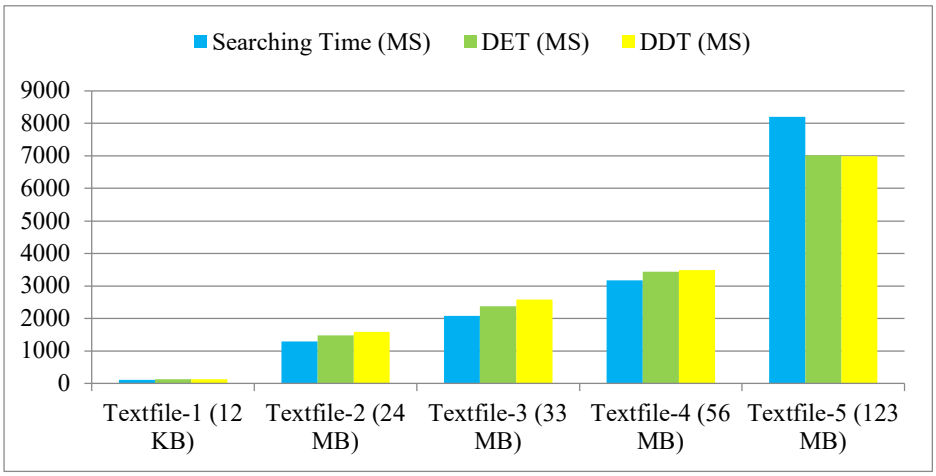

Figure 4: Performance of 3DES (256-bit) in terms of searching the file, encryption and decryption

Table 3: Measuring the time for searching the file, encryption and decryption using AKBCS

File Size (KB)	Searching Time (MS)	DET (MS)	DDT (MS)
Textfile-1 (12 KB)	98.23	112	121
Textfile-2 (24 MB)	1123	1398	1521
Textfile-3 (33 MB)	1991	2212	2434
Textfile-4 (56 MB)	2989	3389	3379
Textfile-5 (123 MB)	7987	6987	6867

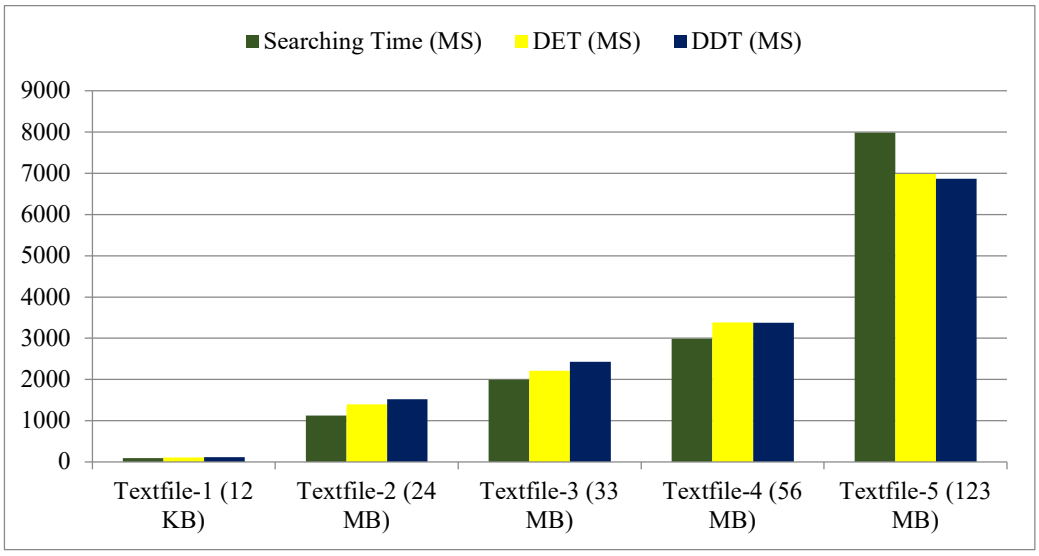

Figure 4: Performance of AKBCS in terms of searching the file, encryption and decryption

Table3 shows the result analysis for the files uploaded at the data owner level and shows the file searching time (ms) and encryption and decryption using AKBCS (RSA). Experiments are conducted on 100 files; here we have shown the result analysis of 5 files. Based on the size of the file the searching time is increased gradually, and also decryption time and encryption time. Compare with AES, 3DES the proposed shows the high performance in terms of searching time, encryption and decryption time.

Conclusion

This paper describes the new security for data storage in cloud computing. Proposed model AKBCS process the cloud data by adding protection to the data. RSA (2048-bit) is used to encrypt data files uploaded by the data owner and user. The proposed approach provides an advanced search based on the encryption of data. An efficient search is applied to data available in a cloud server. The public and private keys are generated for every file, and only authorized users with unique identification can access the data. The proposed approach can efficiently prevent malicious users and attackers from accessing the data. The integrated system in this paper can deploy the application in the public and private clouds. From the experimental evaluation RSA shows the better performance in terms of all the parameters. The small size file is Textfile-1 (12 KB) consumed the 98.23 for searching file , DET (MS) takes 112 and DDT (MS) takes 121.

References

[1] P. T. Jaeger, J. Lin, and J. M. Grimes, "Cloud computing and information policy: Computing in a policy cloud?" J. Inf. Technol. Politics, vol. 5, no. 3, pp. 269–283, Oct. 2008.

[2] C. Vidal and K.-K. R. Choo, "Situational crime prevention and the mitigation of cloud computing threats," in Proc. Int. Conf. Secur. Privacy Commun. Syst. Springer, 2017, pp. 218–233.

[3] N. Khan and A. Al-Yasiri, "Cloud security threats and techniques to strengthen cloud computing adoption framework," in Cyber Security and Threats: Concepts, Methodologies, Tools, and Applications. Hershey, PA, USA: IGI Global, 2018, pp. 268–285.

[4] S. M. Habib, S. Ries, and M. Muhlhauser, " Cloud computing landscape and research challenges regarding trust and reputation," in Proc. 7th Int. Conf. Ubiquitous Intell. Comput. 7th Int. Conf. Autonomic Trusted Comput., Oct. 2010, pp. 410–415.

[5] D. Gonzales, J. M. Kaplan, E. Saltzman, Z. Winkelman and D. Woods, "Cloud-Trust—a Security Assessment Model for Infrastructure as a Service (IaaS) Clouds," in IEEE Transactions on Cloud Computing, vol. 5, no. 3, pp. 523-536, 1 July-Sept. 2017, doi: 10.1109/TCC.2015.2415794.

[6] P. Jamshidi, A. Ahmad, and C. Pahl, "Cloud migration research: A systematic review," IEEE Trans. Cloud Comput., vol. 1, no. 2, pp. 142–157, Jul.–Dec. 2013.

[7] P. K. Tysowski and M. A. Hasan, "Hybrid attribute- and reencryption-based key management for secure and scalable mobile applications in clouds," IEEE Trans. Cloud Comput., vol. 1, no. 2, pp. 172–186, Jul.–Dec. 2013.

[8] Li, M., Yu, S., Zheng, Y., Ren, K., & Lou, W. (2013). "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption." IEEE transactions on parallel and distributed systems, 24(1), 131-143.

[9] Li, M., Yu, S., Cao, N., & Lou, W. (2013). "Privacy-preserving distributed profile matching in proximity-based mobile social networks." IEEE Transactions on Wireless Communications, 12(5), 2024-2033.
[10] H. Deng et al., "Identity-Based Encryption Transformation for Flexible Sharing of Encrypted Data in Public Cloud," in IEEE Transactions on Information Forensics and Security, vol. 15, pp. 3168-3180, 2020, doi: 10.1109/TIFS.2020.2985532.
[11] K. Lee and J. Park, "Identity-based revocation from subset difference methods under simple assumptions," IEEE Access., vol. 7, pp. 60333–60347, 2019.
[12] W. Treesinthuros, "E-commerce transaction security model based on cloud computing," 2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems, Hangzhou, China, 2012, pp. 344-347, doi: 10.1109/CCIS.2012.6664425.
[13] T. Jung, X. Y. Li, Z. Wan, and M. Wan, "Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption," IEEE Transactions on Information Forensics and Security, vol. 10, no. 1, pp. 190–199, 2015.
[14] S. M. Hossein, D. De, P. Mohapatra, S. P. Mondal, and N. Senu, "DNA sequences compression by GP2R and selective encryption using modified RSA technique," IEEE Access, vol. 8, no. 99, p. 1, 2020.
[15] W. Yang, Y. Geng, L. Li, X. Xie and L. Huang, "Achieving Secure and Dynamic Range Queries Over Encrypted Cloud Data," in IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 1, pp. 107-121, 1 Jan. 2022, doi: 10.1109/TKDE.2020.2983030.
[16] N. H. Sultan, V. Varadharajan, L. Zhou and F. A. Barbhuiya, "A Role-Based Encryption (RBE) Scheme for Securing Outsourced Cloud Data in a Multi-Organization Context," in IEEE Transactions on Services Computing, 2022, doi: 10.1109/TSC.2022.3194252.
[17] C. Esposito, A. Castiglione, B. Martini, and K.-K. R. Choo, "Cloud manufacturing: Security, privacy, and forensic concerns," IEEE Cloud Comput., vol. 3, no. 4, pp. 16–22, Jul./Aug. 2016.
[18] Y. Xie, H. Wen, B. Wu, Y. Jiang and J. Meng, "A Modified Hierarchical Attribute-Based Encryption Access Control Method for Mobile Cloud Computing," in IEEE Transactions on Cloud Computing, vol. 7, no. 2, pp. 383-391, 1 April-June 2019, doi: 10.1109/TCC.2015.2513388.
[19] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. S. Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," IEEE Trans. Dependable Secure Comput., vol. 13, no. 3, pp. 312–325, May/Jun. 2016.
[20] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," IEEE Commun. Surveys Tuts., vol. 15, no. 2, pp. 843–859, 2nd Quart, 2013.
[21] Jayaram, B., et al. "A Survey On Social Media Data Analytics And Cloud Computing Tools." International Journal of Mechanical and Production Engineering Research and Development 8.3 (2018): 243.
[22] Khanna, Nikhil, Nishant Behar, and Nishi Yadav. "Improved Algorithm for Dynamic Memory Allocation in Cloud Computing." (2016).
[23] Kaur, Jaspreet, Manpreet Kaur, and Sahil Vashist. "Efficient Virtual Machines Migration in Cloud Computing." International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR) 4.5 (2014): 31-36.

[24] Neelakanteswara, P., and P. SURYANARAYANA Babu. "Efficient trust management technique using neural network in cloud computing." J Comput Netw Wirel Mobile Commun 9.1 (2019): 29-40.
[25] Kushwah, Virendra Singh, Sandip Kumar Goyal, and Priusha Narwariya. "A survey on various fault tolerant approaches for cloud environment during load balancing." Int J Comput Netw Wirel Mobile Commun 4.6 (2014): 25-34.
[26] Khan, Mudassir, and Mohd Ayyoob. "The scope of E-learning in the computer science & technologies." International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR) 6.6 (2016): 93-98.