RESEARCH ARTICLE OPEN ACCESS

Modified Data Sharing Mechanism In Cloud Storage

Steffy Francis , Dr. G. Kiruthiga

Student, Department of Computer Science and Engineering, IES College of Engineering

Associate Professor, Department of Computer Science and Engineering, Kerala - India

ABSTRACT

Data sharing is an important functionality in cloud storage. For example, bloggers can let their friends view a subset of their private pictures an enterprise may grant their employees access to a portion of sensitive data. The challenging problem is how to effectively share encrypted data. Of course users can download the encrypted data from the storage, decrypt them, then send them to others for sharing, but it loses the value of cloud storage. Users should be able to delegate the access rights of the sharing data to others so that they can access these data from the server directly. However, finding an efficient and secure way to share partial data in cloud storage is not trivial. In our scheme, we study how to make a decryption key more powerful in the sense that it provides decryption of multiple cipher texts, without increasing its size. In key-aggregate cryptosystem (KAC), users encrypt data not only under a public-key, butw also under an identifier of cipher text called class. That means the cipher texts are again categorized into different classes. The key owner holds a master-secret called master-secret key, which can be used to obtain secret keys for different classes. More importantly, the extracted key can be an aggregate key which is as short as a secret key for a single class, but aggregates the power of more such keys, that is the decryption power for any portion of cipher text classes.

Keywords— cloud storage, Cryptography, Data sharing.

I. INTRODUCTION

Cloud computing has been the remedy to the problem of personal data management and maintenance due to the growth of personal electronic devices. It is because users can outsource their data to the cloud with ease and low cost. The emergence of cloud computing has also influenced and dominated Information Technology industries. It is unavoidable that cloud computing also suffers from security and privacy challenges [1]-[15].

Cloud storage is gaining popularity recently. In enterprise settings, the rise in demand for data outsourcing, which assists in the strategic management of corporate data. It is also used as a core technology behind many online services for personal applications. Nowadays, it is easy to apply for free accounts for email, photo album, file sharing and/or remote access, with storage size more than 25 GB. Together with the current wireless technology, users can access almost all of their files and emails by a mobile phone in any corner of the world [16]-[20]. Considering data privacy, a traditional way to ensure it is to rely on the server to enforce the access control after authentication, which means any unexpected privilege escalation will expose all data.

In a shared-tenancy cloud computing environment, things become even worse. Data from different clients can be hosted on separate virtual machines (VMs) but reside on a single physical machine. Data in a target VM could be stolen by instantiating another VM coresident with the target one. Regarding availability of files, there are a series of cryptographic schemes which go as far as allowing a third-party auditor to check the availability of files on behalf of the data owner without leaking anything about the data, or without compromising the data owners anonymity [57]-[64]. Likewise, cloud users probably will not hold the strong belief that the cloud server is doing a good job in terms of confidentiality. A cryptographic solution, for example, with proven security relied on number-theoretic assumptions is more desirable, whenever the user is not perfectly happy with trusting the security of the VM or the honesty of the technical staff. These users are motivated to encrypt their data with their own keys before uploading them to the server.

II. MADE WORK

Nowadays, many organizations outsource data storage to the cloud such that a member of an organization (data owner) can easily share data with other members (users).Due to the existence of security concerns in the cloud, both owners and users are suggested to verify the integrity of cloud data with Provable Data Possession (PDP) before further utilization of data. However, previous methods either unnecessarily reveal the identity of a data owner to the un trusted cloud or any public verifiers, or introduce significant overheads on verification metadata for preserving anonymity. It is a simple, efficient, and publicly verifiable approach to ensure cloud data integrity without sacrificing the anonymity of data owners nor requiring significant overhead[21]-[30]. Specifically, introduce a security-mediator (SEM), which is able to generate verification metadata (i.e., signatures) on outsourced data for data owners. This approach decouples the anonymity protection mechanism from the PDP. Thus, an organization can employ its own anonymous authentication mechanism, and the cloud is oblivious to that since it only deals with typical PDP-metadata, Consequently, the identity of the data owner is not revealed to the cloud, and there is no extra storage overhead unlike existing anonymous PDP solutions [31]-[40]. The distinctive features of this scheme also include data privacy, such that the SEM does not learn anything about the data to be uploaded to the cloud at all, and thus the trust on the

SEM is minimized. In addition, extend the scheme to work with the multi-SEM model, which can avoid the potential single point of failure. Security analyses prove that scheme is secure, and experiment results demonstrate that scheme is efficient [46]-[56].

The major benefit of this approach is the decoupling of anonymity protection mechanism from the PDP itself. In other words, the protection of data owners' anonymity incurs no extra cost for cloud service providers or any public verifiers. For this introduced a security mediator maintained by the organization itself, since it is of the organization's interests to control who can use the data storage on its paid cloud service. In this way, less trust is placed by the organization on the cloud. To increase the level of trust they used multi SEM model with the technique of Shamir secret sharing [41]-[45].

The data access mechanism for cloud tenants is based on Boneh and Franklin IBE Algorithm and Biometric recognition. The original motivation for Identity-Based Encryption was to simplify certificate management in E-mail system. when a person A sends mail to another person B at bob@company.com , the former never want to obtain latter's public key certificate, After receiving the mail latter contacts the third party(i.e. private key generator),authenticates himself and obtains his private key, then he can read his mail. The advantage of using IBE is that any string can be used as public key, which means that latter's email address itself can be used as public key(example latter@cloudmail.com).The biometric recognition automatically recognizes the identity of person depending on his/her biological traits, this includes body shape of a person, finger print, voice etc.

The biometric recognition system can operate in two modes 1.verification 2.identification. In verification mode the system accepts or rejects a person's approach to access control. Whereas in the identification determines the identity of a person whom a particular trait belongs to. It is a hypothesis testing problem involving a balance between two error types 1.false reject rate (FRR) and 2.false alarm rate (FAR)S.

III. SYSTEM ARCHITECTURE AND MODULE DESCRIPTION

The existing system consists of five steps 1. setup 2.keygen 3.encrypt 4.extract 5.decrypt. Here in the extraction phase, when the data owner wishes to share a data to his friend, he computes the aggregate key for his friend by performing EXTRACT(msk,s);where 'msk' is the master secret key and 's' is the set of data. The key thus generated is sent to his friend via E- mail. But this is not practical in cases when data owner wants to share data in a regular basis; (say research people and scientist daily shares data 24X7 basis. With the current system in practice, the process of sending the aggregate key to different friends is a time consuming thing.

So in-order to rectify this problem ,we propose a noble idea in which both encryption and sending of keygen to the concerned party are both done by the system itself rather than prompting the data owner to do the same. As a result the time consuming process of sending mails containing the aggregate key can be bypassed, thereby enhancing the user interface.

Advantages

- It is more secure
- Decryption key should send via secure channel and kept secret.
- It is an efficient public key encryption scheme which support flexible delegation.
- The extracted key have can be an aggregate key which is as compact as a secret key for a single class

♦ *MODULE DESCRIPTION*

The concerned project consists of the following modules:

A. Setup Phase
B. Encrypt Phase
C. KeyGen Phase
D. Decrypt Phase
E. User Management Phase
F. Data Management Phase

A. Setup Phase

The setup algorithm takes no input other than the implicit security parameter. It outputs the public parameters and master secret key.

B. Encrypt Phase

Encrypt(PK,M, A). The encryption algorithm takes as input the public parameters PK, a message M, and an access structure A over the universe of attributes. The algorithm will encrypt M and produce a ciphertext CT such that only a user that possesses a set of attributes that satisfies the access structure will be able to decrypt the message. It will assume that the ciphertext implicitly contains A.

C. KeyGen Phase

Phase Key Generation(MK,S). The key generation algorithm takes as input the master key MK and a set of attributes S that describe the key. It outputs a private key SK.

D. Decrypt Phase

Decrypt(PK, CT, SK). The decryption algorithm takes as input the public parameters PK, a ciphertext CT, which contains an access policy A, and a private key SK, which is a private key for a set S of attributes. If the set S of attributes satisfies the access structure A then the algorithm returns a message M.

E. User Management Phase

This part of the module mainly deals with the users who are associated with the proposed system. It is in this part of the module that the username and password are checked before they can log into the system. The user management module is also used for the management of the registered user by the administrator. The user management is the place where the administrator manages the GUI that helps the users to easily use the system without difficulty. Here we implement the side that is directly in contact with the users. A user is allowed to enter the system after authentication of that particular user. The users of the system have to provide user name and password .If a particular user is not in the login table, then he can't access the system.

F. Data Management Phase

The storage cloud is maintained by a third-party cloud provider (e.g., Amazon S3) and keeps the data on behalf of the data owner. We emphasize that we do not require any protocol and implementation changes on the storage cloud to support our system. Even a naive storage service that merely provides file upload/download operations will be suitable.

- *SYSTEM ARCHITECTURE*

The figure 1 shows the architectural diagram of the key aggregate cryptosystem. Suppose Alice wants to share her data m1, m2,...., mn on the server.

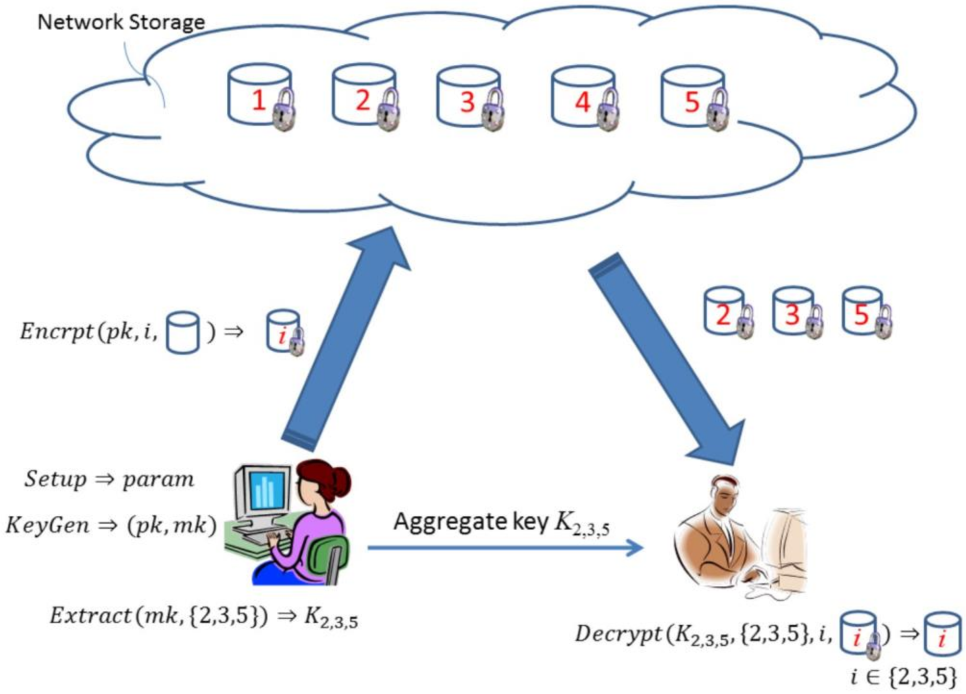

Fig 1. Architectural diagram of proposed system

She first performs Setup (1λ , n) to get param and execute KeyGen to get the public/master-secret key pair (pk, msk). The system parameter param and public-key pk can be made public and master-secret key msk should be kept secret by Alice. Anyone can then encrypt each mi by Ci = Encrypt (pk, i, mi). The encrypted data are uploaded to the server. With param and pk, people who cooperate with Alice can update Alice's data on the server. Once Alice is willing to share a set S of her data with a friend Bob, she can compute the aggregate key KS for Bob by performing Extract (msk, S). Since KS is just a constant size key, it is easy to be sent to Bob through a secure e-mail. After obtaining the aggregate key, Bob can download the data he is authorized to access. That is, for each i ε S, Bob downloads Ci from the server. With the aggregate key KS, Bob can decrypt each Ci by Decrypt (KS, S, i, Ci) for each i ε S.

IV. SYSTEM IMPLEMENTATION

A key-aggregate encryption scheme consists of five polynomial-time algorithms as follows. The data owner establishes the public system parameter via Setup and generates a public/master- secret3 key pair via KeyGen. Messages can be encrypted via Encrypt by anyone who also decides what ciphertext class is associated with the plaintext message to be encrypted. The data owner can use the master-secret to generate an aggregate decryption key for a set of ciphertext classes via Extract. The generated keys can be passed to delegatees securely (via secure e-mails or secure devices) Finally, any user with an aggregate key can decrypt any ciphertext provided that the ciphertext's class is contained in the aggregate key via Decrypt.

- Setup: executed by the data owner to setup an account on an untrusted server. On input a security level parameter 1_ and the number of ciphertext classes n (i.e., class index should be an integer bounded by 1 and n), it outputs the public system parameter param, which is omitted from the input of the other algorithms for brevity.
- KeyGen: executed by the data owner to randomly generate a public/master-secret key pair (pk; msk).
- Encrypt: executed by anyone who wants to encrypt data. On input a public-key pk, an index I denoting the ciphertext class, and a message m, it outputs a ciphertext C.
- Extract: executed by the data owner for delegating the decrypting power for a certain set of ciphertext classes to a delegatee. On input the master secret key msk and a set S of indices corresponding to different classes, it outputs the aggregate key for set S denoted by KS.
- Decrypt : executed by a delegatee who received an aggregate key KS generated by Extract. On input KS, the set S, an index i denoting the ciphertext class the ciphertext C belongs to, and C, it outputs the decrypted result m if i €S.

A canonical application of KAC is data sharing. The key aggregation property is especially useful when we expect the delegation to be efficient and flexible. The schemes enable a content provider to share her data in a confidential and selective way, with a fixed and small ciphertext expansion, by distributing to each authorized user a single and small aggregate key. Using KAC for data sharing in cloud storage .Here we describe the main idea of data sharing in cloud storage using KAC, illustrated in Architectural diagram, Figure 4.1 . Suppose Alice wants to share her data m1,m2,....my on the server. She first performs Setup to get param and execute KeyGen to get the public/master-secret key pair (pk, msk). The system parameter param and public-key pk can be made public and master-secret key msk should be kept secret by Alice. Anyone (including Alice herself) can then encrypt each mi by Ci = Encrypt(pk, i ,mi). The encrypted data are uploaded to the server. With param and pk, people who cooperate with Alice can update Alice's data on the server. Once Alice is willing to share a set S of her data with a friend Bob, she can compute the aggregate key KS for Bob by performing Extract(msk,S). Since KS is just a constant size key, it is easy to be sent to Bob via a secure e-mail. After obtaining the aggregate key, Bob can download the data he is authorized to access. That is, for each i € S, Bob downloads Ci (and some needed values in param) from the server. With the aggregate key KS, Bob can decrypt each Ci by Decrypt for each i € S.

V. SYSTEM TESTING AND RESULTS

SYSTEM TESTING :

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. For any software that is newly developed, primary importance is given to testing the system. It is the last opportunity for the developer to detect the possible error in the software before handling over to the customers.

RESULTS :

Fig 2. HomePage

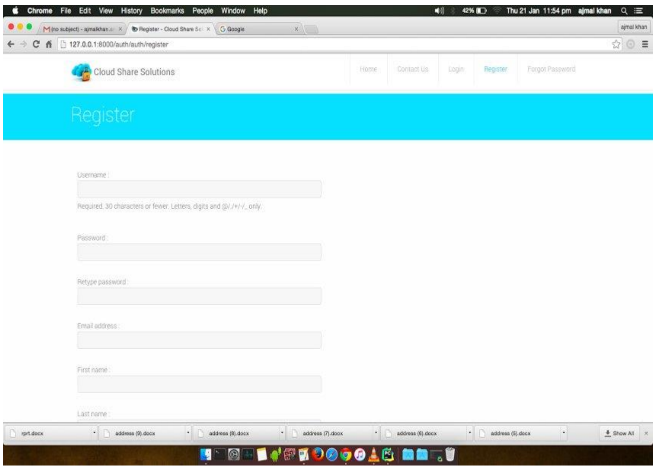

Fig 3. Registration Page

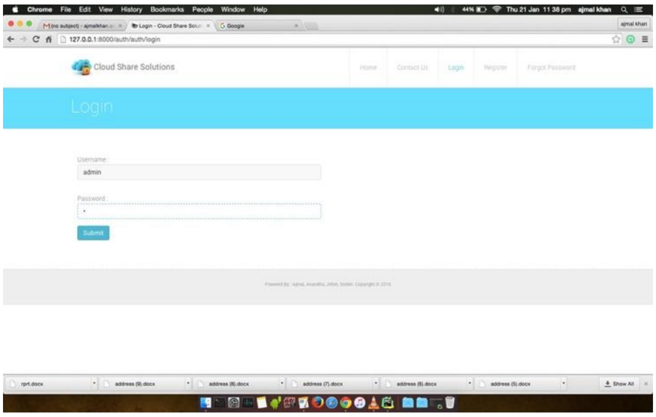

Fig 4. Login Page

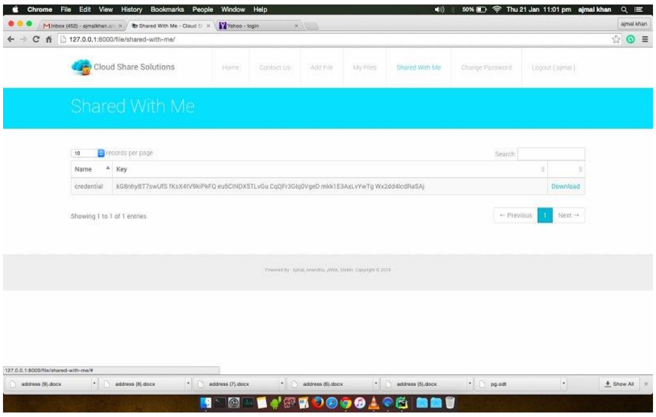

Fig 7. Shared With Me

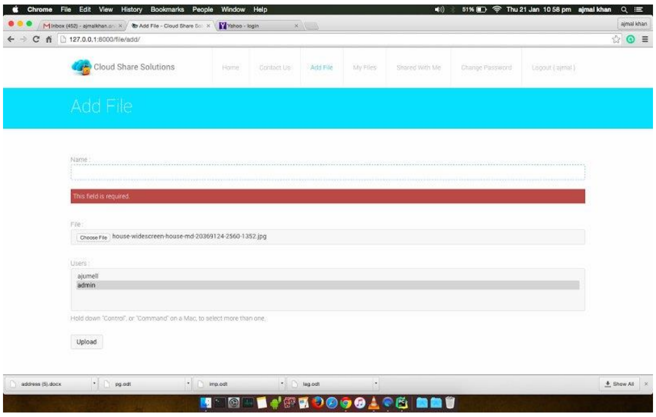

Fig 5. User File Upload

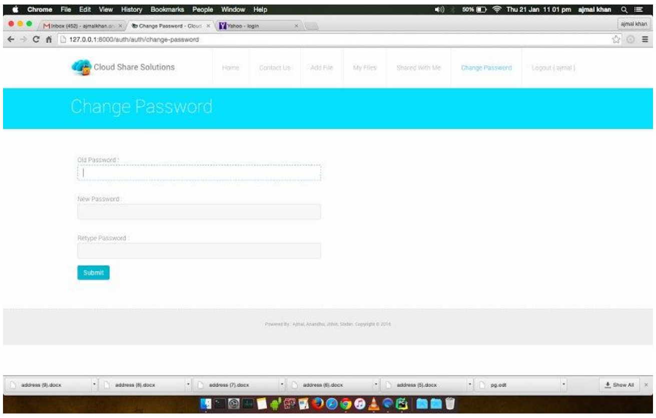

Fig 8. Change Password

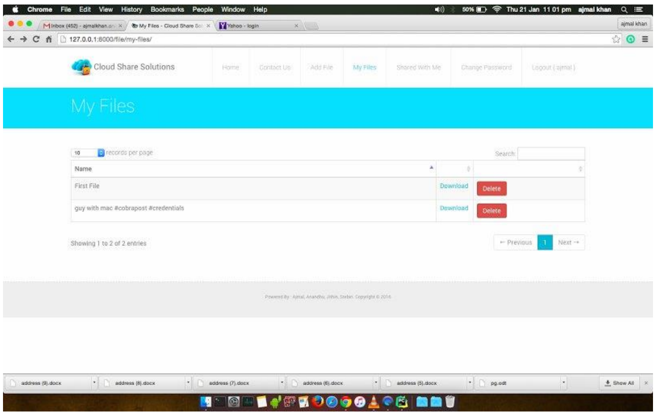

Fig 6. My File

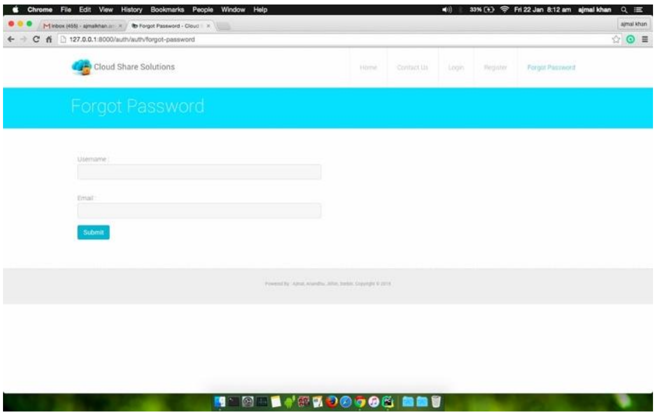

Fig 9. Forgot Password

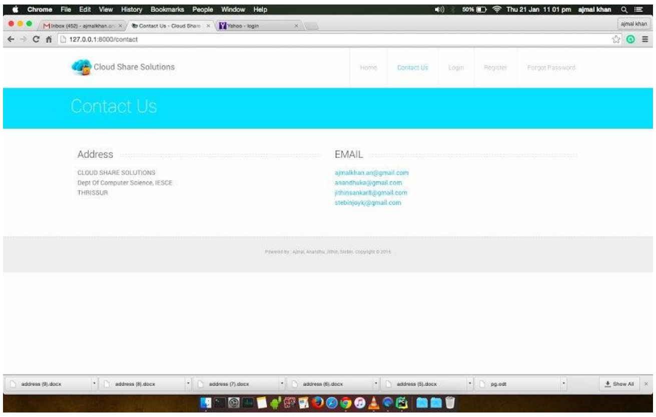

Fig 10. Contact Us

VI. CONCLUSION

Users information privacy is a central question of cloud storage. With extra mathematical tools, cryptographic schemes are getting more flexible and involve multiple keys for a single application. The scheme introduce how to "reduce" secret keys in public-key cryptosystems which support delegation of secret keys for various encrypted classes in cloud storage. These approach is more flexible than hierarchical key assignment which simply save spaces if whole key owner distribute a similar set of privileges. A restriction is the predefined bound of number of most cipher text classes. In cloud storage, number of encrypted text generally grows fastly. That's why we have to reserve more cipher text classes for future work otherwise extend public key.

The parameter can be downloaded with encrypted text, it would be better if its size is not dependent of more number of cipher text classes. On the other side when one carries delegated keys around mobile device without particular accurate hardware, the key is prompt to leakage, designing a leakage resilient cryptosystem allows competent and flexible key delegation is interesting way.

REFERENCES

1. B. Wang, S. S. M. Chow, M. Li, and H. Li, "Storing Shared Data on the Cloud via Security-Mediator," in International Conference on Distributed Computing Systems - ICDCS 2013. IEEE, 2013.
2. M. J. Atallah, M. Blanton, N. Fazio, and K. B. Frikken, "Dynamic and Efficient Key Management for Access Hierarchies," ACM Transactions on Information and System Security (TISSEC), vol. 12, no. 3, 2009
3. Chunming Rong and Hongbing Cheng, "A Secure Data Access Mechanism for Cloud Tenants," Cloud computing 2012: The Third International Conference on Cloud Computing, GRIDs, and Virtualization, ISBN: 978-1-61208-216-5, IARIA, 2012.
4. Baalamurugan, K. M., & Bhanu, S. V. (2019). An efficient clustering scheme for cloud computing problems using metaheuristic algorithms. Cluster Computing, 22(5), 12917-12927.
5. Baalamurugan, K. M., & Bhanu, S. V. (2020). A multi-objective krill herd algorithm for virtual machine placement in cloud computing. The Journal of Supercomputing, 76(6), 4525-4542.
6. Umar, M., Babu, D., Baalamurugan, K. M., & Singh, P. (2020). Automation of Energy Conservation for Nodes in Wireless Sensor Networks. International Journal of Future Generation Communication and Networking, 13(3).
7. Baalamurugan, K. M., & Bhanu, D. S. V. (2018). Analysis of Cloud Storage Issues in Distributed Cloud Data Centres by Parameter Improved Particle Swarm Optimization (PIPSO) Algorithm. Int. J. Future Revolut. Comput. Sci. Commun. Eng, 4, 303-307.
8. Saravanabhavan, C., Saravanan, T., Mariappan, D. B., Nagaraj, S., Vinotha, D., & Baalamurugan, K. M. (2021, March). Data Mining Model for Chronic Kidney Risks Prediction Based on Using NB-CbH. In 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) (pp. 1023-1026). IEEE.
9. Ramalingam, V., Mariappan, D. B., Gopal, R., & Baalamurugan, K. M. (2020). An Effective Social Internet of Things (SIoT) Model for Malicious Node Detection in Wireless Sensor Networks. Artificial Intelligence Techniques in IoT Sensor Networks, 181.
10. Baalamurugan, K. M., Gopal, R., Vinotha, D., Daniel, A., & Ramalingam, V. (2020). An Energy-Efficient Quasi-Oppositional Krill Herd Algorithm-Based Clustering Protocol for Internet of Things Sensor Networks. Artificial Intelligence Techniques in IoT Sensor Networks, 167.
11. Bala, B. K. (2019). Enhanced Palm Vein Recognition Algorithm with Equalizer Technique. International Journal of Engineering and Advanced Technology, 8(5), 888-890.
12. Bala, B. K., & Nithya, T. M. Remedy For Disease Affected Iris In Iris Recognition. International Journal of Research in Engineering and Technology, (2012), 332-334.
13. Bala, B. K., & Kumar, R. (2017). A novel method of cultivation of different varieties of tomato without using soil. BIOSCIENCE BIOTECHNOLOGY RESEARCH COMMUNICATIONS, 10(4), 802-804.
14. Kumar, R. S., & Bala, B. K. (2017). Identification Of Cancer From The Mammogram Images By Using Frequency Domain Approaches. International Journal of ChemTech Research, 10(5).
15. Balakumar, A., & Bala, B. K. (2019). Comparison Of Various Transformations In Fingerprint Recognition. RESEARCH JOURNAL OF PHARMACEUTICAL BIOLOGICAL AND CHEMICAL SCIENCES, 10(1), 888-890.
16. Deborah, L. J., Karthika, R., Audithan, S., & Bala, B. K. (2015). Enhanced Expressivity using Deontic Logic and Reuse Measure of Ontologies. Procedia Computer Science, 54, 318-326.
17. Muthuperumal, S., Venkatachalam, A., & Bala, B. K. (2021). Analysis of Various Pet Animals by Using Deep Learning Algorithm. Annals of the Romanian Society for Cell Biology, 7362-7365.
18. Chellamuthu, K., Vasanthanathan, A., & Bala, B. K. (2021). Experimental Analysis of Fiber Reinforced Plastic Structure with Bio Filler under Impact Load. Annals of the Romanian Society for Cell Biology, 6652-6660.
19. Raj, I. I., & Bala, B. K. (2021). A Novel approach for Infected Lungs by using Different transformations. Bull. Env. Pharmacol. Life Sci, 10, 180-183.

20. Daniel, A., & Baalamurugan, K. M. (2020). A novel approach to minimize classifier computational overheads in Big Data using neural networks. Physical Communication, 42, 101130.
21. Kousik, N., Natarajan, Y., Raja, R. A., Kallam, S., Patan, R., & Gandomi, A. H. (2021). Improved salient object detection using hybrid Convolution Recurrent Neural Network. Expert Systems with Applications, 166, 114064.
22. Yuvaraj, N., Srihari, K., Dhiman, G., Somasundaram, K., Sharma, A., Rajeskannan, S., ... & Masud, M. (2021). Nature-Inspired-Based Approach for Automated Cyberbullying Classification on Multimedia Social Networking. Mathematical Problems in Engineering, 2021.
23. Natarajan, Y., Kannan, S., & Mohanty, S. N. (2021). Survey of Various Statistical Numerical and Machine Learning Ontological Models on Infectious Disease Ontology. Data Analytics in Bioinformatics: A Machine Learning Perspective, 431-442.
24. Raja, R. A., Yuvaraj, N., & Kousik, N. V. (2021). Analyses on Artificial Intelligence Framework to Detect Crime Pattern. Intelligent Data Analytics for Terror Threat Prediction: Architectures, Methodologies, Techniques and Applications, 119-132.
25. Kannan, S., Dhiman, G., Natarajan, Y., Sharma, A., Mohanty, S. N., Soni, M., ... & Gheisari, M. (2021). Ubiquitous Vehicular Ad-Hoc Network Computing Using Deep Neural Network with IoT-Based Bat Agents for Traffic Management. Electronics, 10(7), 785.
26. Yuvaraj, N., Raja, R. A., Karthikeyan, T., & Kousik, N. V. (2020). 11 Improved Privacy Preservation Framework for Cloud-Based Internet of Things. Internet of Things: Integration and Security Challenges, 165.
27. Yuvaraj, N., Karthikeyan, T., & Praghash, K. (2021). An improved task allocation scheme in serverless computing using gray wolf Optimization (GWO) based reinforcement learning (RIL) approach. Wireless Personal Communications, 117(3), 2403-2421.
28. Mariappan, L. T., & Yuvaraj, N. (2020). Analysis On Cardiovascular Disease Classification Using Machine Learning Framework. Solid State Technology, 63(6), 10374-10383.
29. Gowrishankar, J., Narmadha, T., Ramkumar, M., & Yuvaraj, N. (2020). Convolutional Neural Network Classification On 2d Craniofacial Images. International Journal of Grid and Distributed Computing, 13(1), 1026-1032.
30. Karthick, S., Yuvaraj, N., Rajakumari, P. A., & Raja, R. A. (2021). Ensemble Similarity Clustering Frame work for Categorical Dataset Clustering Using Swarm Intelligence. In Intelligent Computing and Applications (pp. 549-557). Springer, Singapore.
31. Yuvaraj, N., Raja, R. A., & Kousik, N. V. (2021). Privacy Preservation Between Privacy and Utility Using ECC-based PSO Algorithm. In Intelligent Computing and Applications (pp. 567-573). Springer, Singapore.
32. Yuvaraj, N., Raja, R. A., Palanivel, P., & Kousik, N. V. (2020, April). EDM Process by Using Copper Electrode with INCONEL 625 Material. In IOP Conference Series: Materials Science and Engineering (Vol. 811, No. 1, p. 012011). IOP Publishing.
33. Veerappan Kousik, N. G., Natarajan, Y., Suresh, K., Patan, R., & Gandomi, A. H. (2020). Improving Power and Resource Management in Heterogeneous Downlink OFDMA Networks. Information, 11(4), 203.
34. Natarajan, Y., Raja, R. A., Kousik, D. N., & Johri, P. (2020). Improved Energy Efficient Wireless Sensor Networks Using Multicast Particle Swarm Optimization. Available at SSRN 3555764.
35. Khadidos, A., Khadidos, A. O., Kannan, S., Natarajan, Y., Mohanty, S. N., & Tsaramirsis, G. (2020). Analysis of COVID-19 Infections on a CT Image Using DeepSense Model. Frontiers in Public Health, 8.
36. Yuvaraj, N., Srihari, K., Chandragandhi, S., Raja, R. A., Dhiman, G., & Kaur, A. (2021). Analysis of protein-ligand interactions of SARS-Cov-2 against selective drug using deep neural networks. Big Data Mining and Analytics, 4(2), 76-83.
37. Karthick, S., Yuvaraj, N., Rajakumari, P. A., & Raja, R. A. (2021). Ensemble Similarity Clustering Frame work for Categorical Dataset Clustering Using Swarm Intelligence. In Intelligent Computing and Applications (pp. 549-557). Springer, Singapore.
38. Yuvaraj, N., Raja, R. A., & Kousik, N. V. (2021). Privacy Preservation Between Privacy and Utility Using ECC-based PSO Algorithm. In Intelligent Computing and Applications (pp. 567-573). Springer, Singapore.
39. Daniel, A., Kannan, B. B., Yuvaraj, N., & Kousik, N. V. (2021). Predicting Energy Demands Constructed on Ensemble of Classifiers. In Intelligent Computing and Applications (pp. 575-583). Springer, Singapore.
40. Yuvaraj, N., Raja, R. A., Kousik, N. V., Johri, P., & Diván, M. J. (2020). Analysis on the prediction of central line-associated bloodstream infections (CLABSI) using deep neural network classification. In Computational Intelligence and Its Applications in Healthcare (pp. 229-244). Academic Press.
41. Sangeetha, S. B., Blessing, N. W., Yuvaraj, N., & Sneha, J. A. (2020). Improving the training pattern in back-propagation neural networks using holt-winters' seasonal method and gradient boosting model. In Applications of Machine Learning (pp. 189-198). Springer, Singapore.
42. Natarajan, Y., Raja, R. A., Kousik, D. N., & Johri, P. (2020). Improved Energy Efficient Wireless Sensor Networks Using Multicast Particle Swarm Optimization. Available at SSRN 3555764.
43. Yuvaraj, N., Kousik, N. V., Jayasri, S., Daniel, A., & Rajakumar, P. (2019). A survey on various load balancing algorithm to improve the task scheduling in cloud computing environment. J Adv Res Dyn Control Syst, 11(08), 2397-2406.
44. Kiruthiga, G., & Mohanapriya, M. (2019). An adaptive signal strength based localization approach for wireless sensor networks. Cluster Computing, 22(5), 10439-10448.
45. Swaraj, P. K., & Kiruthiga, G. DESIGN AND ANALYSIS ON MEDICAL IMAGE CLASSIFICATION FOR DENGUE DETECTION USING ARTIFICIAL NEURAL NETWORK CLASSIFIER, ICTACT JOURNAL ON IMAGE AND VIDEO PROCESSING, FEBRUARY 2021, VOLUME: 11, ISSUE: 03, pp. 2407-2412.
46. G Kiruthiga, M Mohanapriya, "NEIGHBOR BASED CLUSTER LOCATION AWARE ROUTING FOR MAXIMUM PACKET DELIVERY IN WSN", Jour of Adv Research in Dynamical & Control Systems 9 (special issue), 1046-1064, 2017.
47. G Kiruthiga, M Mohanapriya, " Survey on the Localization with Secured Routing in WSN", Jour of Adv Research in Dynamical & Control Systems 9 (special issue), 649-655, 2017.

48. G.Kiruthiga, P.MayilVel Kumar, K.M.Murugesan, Improved Fault Diagnosis in Wireless Sensor Networks using Deep Learning Technique, International Journal of Recent Technology and Engineering (IJRTE), Vol. 8, Issue. 1S4, pp. 757-760, 2019.
49. G Kiruthiga, K Kalaiselvi, RS Shudapreyaa, V Dineshbabu, Detection of Faults in Flying Wireless Sensor Networks Using Adaptive Reinforcement Learning, International Journal of Recent Technology and Engineering (IJRTE), Vol. 8, Issue. 1S4, pp. 761-763, 2019.
50. G Kiruthiga, MULTI CRITERIA ARCHITECTURE FOR WIRELESS SENSOR NETWORKS, International Journal of Advanced Information Science and Technology (IJAIST), Vol. 9, Issue 9, pp. 54-58, 2013.
51. G Kiruthiga, Reducing Energy Consumption in Wireless Sensor Networks, International Journal of Advanced Information Science and Technology (IJAIST), Vol. 8, Issue 8, pp. 90-95, 2012.
52. Shakkeera, L., & Saranya, A. (2019). Efficient Collaborative Key Management Protocol for Secure Mobile Cloud Data Storage. In International Conference on Intelligent Computing and Applications (pp. 41-51). Springer, Singapore.
53. SHAKKEERA, L., & TAMILSELVAN, L. (2017). SATISFYING SLA OBJECTIVES OF SEAMLESS EXECUTION OF MOBILE APPLICATIONS IN CLOUD WITH NET PROFIT. Journal of Theoretical & Applied Information Technology, 95(11).
54. Latha, T., & Shakkera, L. (2017). Towards Maximum Resource Utilization and Optimal Task Execution for Gaming IoT Workflow in Mobile Cloud. International Journal of Intelligent Engineering and Systems, 10(1), 134-143.
55. Shakkeera, L., & Tamilselvan, L. (2016). QoS and load balancing aware task scheduling framework for mobile cloud computing environment. International Journal of Wireless and Mobile Computing, 10(4), 309-316.
56. Tamilselvan, L. (2014, April). QoS based dynamic task scheduling in IaaS cloud. In 2014 International Conference on Recent Trends in Information Technology (pp. 1-8). IEEE.
57. Lodh, A., Saxena, U., Motwani, A., Shakkeera, L., & Sharmasth, V. Y. (2020, November). Prototype for Integration of Face Mask Detection and Person Identification Model–COVID-19. In 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 1361-1367). IEEE.
58. Lakshmanan, S. K., Shakkeera, L., & Pandimurugan, V. (2020, December). Efficient Auto key based Encryption and Decryption using GICK and GDCK methods. In 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS) (pp. 1102-1106). IEEE.
59. Azath, H., & Wahidabanu, R. S. D. (2008). Function point: a quality loom for the effort assessment of software systems. International Journal of Computer Science and Network Security, 8(12), 321-328.
60. Maheswari, R., Azath, H., Sharmila, P., & Gnanamalar, S. S. R. (2019, May). Smart village: Solar based smart agriculture with IoT enabled for climatic change and fertilization of soil. In 2019 IEEE 5th International Conference on Mechatronics System and Robots (ICMSR) (pp. 102-105). IEEE.
61. Amudhavalli, D., Rajalakshmi, D. S., & Marikannan, D. M. (2018). An efficient software effort estimation by combining neural network and optimization technique. International Journal of Applied Engineering Research, 13(6), 3890-3897.
62. Azath, H., Mohanapriya, M., & Rajalakshmi, S. (2018). Software Effort Estimation Using Modified Fuzzy C Means Clustering and Hybrid ABC-MCS Optimization in Neural Network. Journal of Intelligent Systems, 29(1), 251-263.
63. Shukla, A., Kalnoor, G., Kumar, A., Yuvaraj, N., Manikandan, R., & Ramkumar, M. (2021). Improved recognition rate of different material category using convolutional neural networks. Materials Today: Proceedings.
64. Yuvaraj, N., Chang, V., Gobinathan, B., Pinagapani, A., Kannan, S., Dhiman, G., & Rajan, A. R. (2021). Automatic detection of cyberbullying using multi-feature based artificial intelligence with deep decision tree classification. Computers & Electrical Engineering, 92, 107186.
65. B Prasanalakshmi, A Kannammal "Secure credential federation for hybrid cloud environment with SAML enabled multifactor authentication using biometrics" International Journal of Computer Applications, (2012),Vol.53, Issue.18.
66. Satish, Karuturi S R V, and M Swamy Das. "Multi-Tier Authentication Scheme to Enhance Security in Cloud Computing." IJRAR (International Journal of Research and Analytical Reviews) 6, no. 2 (2019): 1-8, 2019.