Journal of Theoretical and Applied Information Technology
31st March 2024. Vol.102. No 6
© Little Lion Scientific

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SECURE AND EFFICIENT DATA SHARING SCHEME FOR MULTI-USER AND MULTI-OWNER SCENARIO IN FEDERATED CLOUD COMPUTING

Dr. IMTIYAZ KHAN[1], Dr. A.YASHWANTH REDDY[2], Dr. MANIZA HIJAB[3], Dr. KOTARI SRIDEVI[4], Dr. SYED SHABBEER AHMAD[5], Dr. D.SHRAVANI[6]

[1]Professor, Department of CSE, Shadan College of Engineering and Technology, JNTUH, Hyderabad TS India
[2]HOD Department of CSE, Sree Dattha Group of Institutions, JNTUH, Hyderabad TS India
[3]Associate Professor, Department of CSE, Muffakham Jah College of Engineering and Technology, OU, Hyderabad, TS ,India
[4]Associate Professor, Department of CSE, Muffakham Jah College of Engineering and Technology, OU, Hyderabad, TS ,India
[5]Professor, Department of CSE, Muffakham Jah College of Engineering and Technology, OU, Hyderabad, TS ,India
[6]Associate Professor Department of ADCE, Stanley College of Engineering and Technology for Women, OU, Hyderabad TS India
E-mail: [1]imtiyaz.khan.7@gmail.com , [2]yashwanth.alg@gmail.com, [3]manizahijab@mjcollege.ac.in, [4]sridevi@mjcollege.ac.in, [5]shabbeer.ahmad@mjcollege.ac.in, [6]drdasarishravani@stanley.edu.in

ABSTRACT

Cloud computing, since its inception, has undergone continuous improvements. Now federated cloud is realized with seamless integration of diversified clouds. In this context, it is essentially and multi-owner and multi-user environment where security to data of data owners is to be given paramount importance. Supporting data sharing with security in place and enabling users to perform keyword search on the encrypted content is indispensable in such environment. The existing schemes suffer from performance issues in complex multi-owner and multi-user scenarios in federated cloud setting. To address this problem, in this paper, we proposed a security scheme that enables efficient data sharing across users. Users are able to access data of multiple data owners by generating trapdoors. Different algorithms are proposed to realize the scheme. With empirical study, it is observed that our scheme is able to support secure and efficient data sharing in federated cloud environment. Our scheme performs better than existing ones in terms of storage overhead and execution time.

Keywords – *Secure Data Sharing, Cloud Computing, Federated Cloud, Aggregate Key Sharing, Cloud Data Security*

1. INTRODUCTION

In cloud based applications secure data sharing is an essential requirement. However, flexibility and sophistication in group data sharing is the main focus of this paper. Data owners who use cloud infrastructure for storage and data sharing to their users usually encrypt the data prior to sending to cloud. Such data is secure in transit and also when it is at rest. However, the data owners need to use aggregate keys and share them to users in order to help them generating trapdoors to search for required files on encrypted content in cloud. In presence of adversaries, it is important to have secure and efficient algorithms for protecting data and also prevent any kinds of attacks. Towards this end, many data security schemes came into existence. Out of them searchable encryption schemes became popular for cloud environments. However, they suffer from overhead and time complexity in presence of multi-owner and multi-user environments.

Miao et al. [1] addressed by Verifiable SE Framework, which also offers dynamic updates and multi-keyword search. Improved VSEF exhibits adaptability. The framework is expanded upon by further study. Fu et al. [4] enhanced security and cutting storage costs in cloud storage is possible with a multi-cloud searchable encryption technique built on a double-layer block chain. Xu et al. [7] mitigated the constraints of delegated searchable encryption (DSE), a technique is implemented that limits users to specific terms in order to protect user privacy. Shown by trials to be useful. Yao et al. [15]

suggested key-aggregate encryption and searchable encryption based on lattices to protect against quantum assaults and provide effective cloud storage for searchable group data sharing. Manohar et al. [20] utilized KASE to combat the problem of safely exchanging encrypted data in public cloud storage. Erande and Ranmalkar [24] observed that shared data must be selectively encrypted due to security concerns raised by cloud data breaches. It is observed from state of the art that there is need for improving security and efficiency of schemes for federated cloud environment. Our contributions in this paper are as follows.

1. Proposed a secure data sharing scheme that facilitates users to access data of many data owners by generating trapdoors.
2. The scheme supports federated cloud where many data owners can share data to multiple users.
3. Our scheme performs better than existing ones in terms of efficiency, storage overhead and execution time.

The structure of rest of the paper is as follows. Section 2 focuses on literature review of many existing schemes. Section 3 describes our scheme for federated cloud. Section 4 provides empirical observations of our research. Section 5 discusses uniqueness of our scheme and its limitations. Section 6 concludes our work and provides scope for future work.

2. RELATED WORK

This section reviews many existing security schemes useful for secure data sharing in distribute environments. Miao et al. [1] addressed by Verifiable SE Framework, which also offers dynamic updates and multi-keyword search. Improved VSEF exhibits adaptability. The framework is expanded upon by further study. Zhou et al. [2] deployed for ambient data using cloud-assisted Industrial IoT. IoT device expenses are mitigated by cloud storage. Proposed is keyword encryption as a secure and effective method for device search. Zhang et al. [3] presented for safe voice retrieval in cloud storage using multiuser searchable encryption. Makes use of LSTM, SE, and CP-ABE for privacy. Fu et al. [4] enhanced security and cutting storage costs in cloud storage is possible with a multi-cloud searchable encryption technique built on a double-layer block chain. Sun et al. [5] observed that by combining CP-ABE with auditing for data integrity and attribute revocation, a workable multi-keyword searchable encryption system improves efficiency and security. Shahien et al. [6] improved speed and security, the proposed Multi-Server Searchable Data Crypt (MS-SDC) splits encrypted data into blocks. Multithreading for speed and keyword extraction are features.

Xu et al. [7] mitigated the constraints of delegated searchable encryption (DSE), a technique is implemented that limits users to specific terms in order to protect user privacy. Shown by trials to be useful. Sangeetha et al. [8] optimized cloud-based PHR for effective storage, search, and sharing by integrating CM-SABE, DLBRE, and approved deduplication. Demonstrated to boost output. Brij et al. [9] proposed decentralized ABSE scheme for healthcare CCPS aims to improve efficiency and eliminate single points of failure by utilizing blockchain technology to spread computing burden. Xiao et al. [10] suggested multi-keyword ranked search system, or MSMR, improves the security and performance of encrypted data retrieval in cloud storage. Sharma et al. [11] demonstrated through theoretical analysis and simulations that MWMR-BKSE provides secure Boolean searches with a minimal computing cost.

Liu et al. [12] presented the innovative ICA-IBSE scheme, which emphasizes practicality, less storage, and proven security for effective encrypted data search in cloud computing. Varri et al. [13] introduced CP-ABSEL, a searchable encryption for cloud storage that uses a lattice to provide quantum security and efficient access management while protecting data privacy. Zarezadeh et al. [14] enhanced searchable encryption with access control for cloud storage was introduced, resolving problems with Pasupuleti et al.'s approach and guaranteeing efficiency and security in multi-keyword searches. Yao et al. [15] suggested key-aggregate encryption and searchable encryption based on lattices to protect against quantum assaults and provide effective cloud storage for searchable group data sharing. Martin et al. [16] addressed lattice reduction procedures, analyses algorithms, and estimates resources in order to assemble hardness results for particular cases of the learning with errors (LWE) issue.

Bindel et al. [17] analysed several attacks and techniques for the Learning with Errors (LWE) issue within a limited amount of samples. Pol and Priyadarshi [18] explored and observed that by using global secret keys, asymmetric key management, and key aggregation, the suggested solution protects integrity and privacy in cloud computing. Feng and Si et al. [19] combined public key authentication with searchable encryption to provide a certificate less searchable encryption solution for many users that improves security and efficiency against keyword guessing attacks. Manohar et al. [20] utilized key aggregate searchable encryption (KASE) to combat the problem of safely exchanging encrypted data in public cloud storage. Anusha et al. [21] suggested key aggregate searchable encryption (KASE) as a solution to the problem of safe data sharing in public cloud storage.

Goutham et al. [22] studied selective encrypted data exchange which is essential in public cloud storage. Practical privacy is provided by key aggregate searchable encryption (KASE), which effectively handles keys. Kamimura et al. [23] introduced two provably secure techniques and explore KASE security. Although the primary construction assures privacy in a two-server setup, the first construction ensures security without adding to computational expenses. Practicality and verifiable security are achieved by both systems. Further efforts will encompass optimizing efficiency and developing a universal framework for all broadcast encryption and aggregate signatures. Erande and Ranmalkar [24] observed that shared data must be selectively encrypted due to security concerns raised by cloud data breaches. The proposal for Key-Aggregate Searchable Encryption (KASE) tackles real-world problems related to safe sharing, with a focus on trapdoor minimization and federated cloud support. Rane et al. [25] introduced constant-size ciphertexts for delegation and focuses on safe, effective data exchange in cloud storage. Sharing of practical data is facilitated by the proposed key-aggregate searchable encryption scheme (KASE). Future considerations, however, will include lowering trapdoors in multi-owner scenarios and modifying KASE for federated clouds. Lee et al. [26] provided a KASE strategy for data sharing without a Trusted Third Party (TTP) in order to overcome privacy issues in cloud servers. The new approach provides protection against several types of attacks, mutual authentication, multi-delegation, and keyword verification.

Kavatagi and Rachh [27] implemented a searchable encryption with key aggregation to address the security of cloud computing. It provides quick document retrieval using a trapdoor and allows secure sharing with a single key. The goal of future research is to minimize trapdoors in situations with many owners. Sonkar and Wakchaure [28] discussed safe cloud data sharing, emphasizing effective search functions, key management, and encryption. The Key-Aggregate Searchable Encryption (KASE) technique adds synonym search for increased performance, provides flexible authorization, and improves security. Experimental findings show enhanced security, efficiency, and performance of the system. Gadekar and Pradip [29] addressed security issues with cloud data sharing because to data breaches. The study promotes cloud-based effective key storage and data exchange, emphasizing the advantages of key aggregation for maximum throughput and best space use. Rekesh and Anoop [30] provided efficient cryptographic data sharing in cloud storage. A single aggregate key is used for big document sets in this manner to enable secure sharing. Users provide cloud querying with a single aggregate gateway. An ideal option for realistic data sharing in public cloud storage is offered by the suggested method, which improves security and efficiency. Reviewing approaches and highlighting KASE's efficacy, it takes into account the difficulties in exchanging data without leaking. Upcoming efforts will focus on reducing trapdoor creation and addressing multi-owner data sharing difficulties. Bankar and Sidramappa [31] addressed by encrypting all data before uploading. It is difficult to manage and distribute keys for encryption and search in a safe manner, though. Key-Aggregate Search Encryption (KASE), a unique technique, suggests an effective key distribution scheme for cloud data access. The solution enables safe, useful, and private data sharing using Role-Based Access Control (RBAC) and secure revocation for untrusted users in a cloud context, addressing overlooked practical difficulties.

Thakre et al. [32] found that although file deployment and sharing are available with cloud computing, security is an issue. Secure, scalable data exchange is made easier with key-based encryption. Bagga [33] existed data security techniques are vulnerable to legal threats. Access control and encryption are combined in a suggested solution to solve these problems and offer strong security for sensitive data. Brindha et al. [34] enhanced key-aggregate searchable encryption, is a suggested solution that effectively and safely handles these problems. Subsequent research endeavours to optimize keys for increased effectiveness. Samalla et al. [35] explained cloud computing privacy concerns and introduces Key Aggregate Searchable Encryption (KASE), a safe data sharing method. Sumeen et al. [36] presented KASE, addressing privacy concerns and offering effective solutions, enabling safe data sharing in cloud storage.

Gayathri and Srinaganya [37] enhanced security for remote access services, and the proposed Key Tree (KTR) system effectively maintains keys for safe transmission in distributed storage. Wang et al. [38] presented the EVKAKSE system to solve security and efficiency issues in cloud data sharing. In the event that an assist server is not there, the structure takes efficiency and security into account without sacrificing data integrity. The goal of future research is to improve the algorithm such that it can remove the help server without sacrificing security. Wang et al. [39] addressed the issues of key generation and trapdoors while concentrating on effective keyword search on encrypted data stored in cloud storage. Prioritizing efficiency and security, the study provides information on how to thwart hostile user and server collaboration. Future research seeks to find more effective treatments. Guo et al. [40] suggested the use of a key-aggregate authentication mechanism to allow safe data exchange in dynamic cloud storage. Addressing issues with dynamic cloud storage, the method is both leakage-resistant and cost-stable. Notwithstanding some restrictions, the plan has potential uses in a number of contexts, such as searchable encryption in cloud storage and patient-controlled encryption. Su et al. [41] introduced Verifiable Multi-Key Searchable Encryption (VMKSE) that ensures efficiency and verifiability against hostile entities while enabling safe data exchange in multi-user scenarios through the use of Garbled Bloom Filter. From the literature, it was observed that the existing data sharing schemes suffer from mediocre performance in multi-owner and multi-user setting and not designed for federated cloud environments.

3. PROPOSED SECURITY SCHEME

This section presents our proposed scheme and also algorithms required for realization of the scheme.

3.1 Problem Definition

Secure data sharing in cloud based applications among groups of users is challenging. In multi-user and multi-owner federated cloud environment, development of a data sharing scheme with security is the challenging problem considered.

3.2 Federated Cloud

Federated cloud, as shown in Figure 1, is the integration of various clouds for ultimate scalability and elasticity. Imagine a world where you could seamlessly move workloads and data between different cloud providers, leveraging the best of each while maintaining centralized control and governance. That's the essence of cloud federation - Deploying and managing multiple cloud services (public, private, community) from different providers to create a unified computing platform. Intermediary between cloud coordinator and broker, evaluating expenses, requirements, and suppliers Resources are assigned by Cloud Coordinator according to user credits in the cloud bank and their requests. By working with cloud coordinators and examining the resources provided by various cloud providers in cloud exchange, Cloud Broker finds the best bargains for clients. Centralized or decentralized user interaction. Applications, both for profit and non-profit, MaaS and global visibility monitoring an offer/demand procedure that is centralized. Federation-based consumption of infrastructure, software, and platform. Lower use of energy, heightened dependability, Cost-effectiveness and scalability, worldwide communication and service sharing.

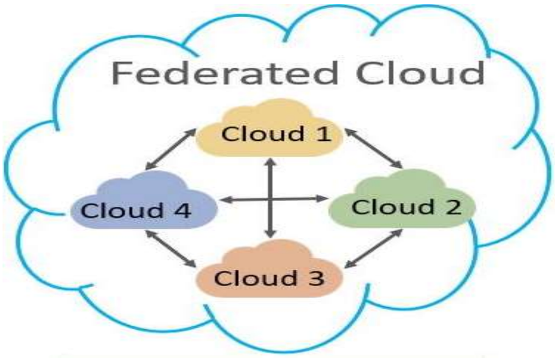

Figure 1: Illustration Of Federated Cloud

Demand distribution across providers, Interoperability in diverse environments and Building a seamless user experience are the challenges. There are three different kinds of technology that support cloud federation and cloud services. They're Eucalyptus is an open-source framework for accessing cloud resources, whereas Aneka Coordinator facilitates cloud service interaction (proposal of the Aneka services and Aneka peer components). Open Nebula is a cloud computing platform that manages remote data centres and resources. In summary, cloud federation provides an adaptable and affordable way to combine resources from many cloud providers, but complexity, security, and standards must be carefully considered.

3.3 System Model

Our system model has provision for federated cloud and an adversary might launch attacks to break security. In presence of an attacker, the system model is designed and based on that our entire security scheme is built. The system model enables multiple data owners to save their data in cloud in encrypted format and share the required keys to designated users. The system model also has provision for an attacker who tries to break the system. Our system model is shown in Figure 2 on top of which the proposed security scheme is built.

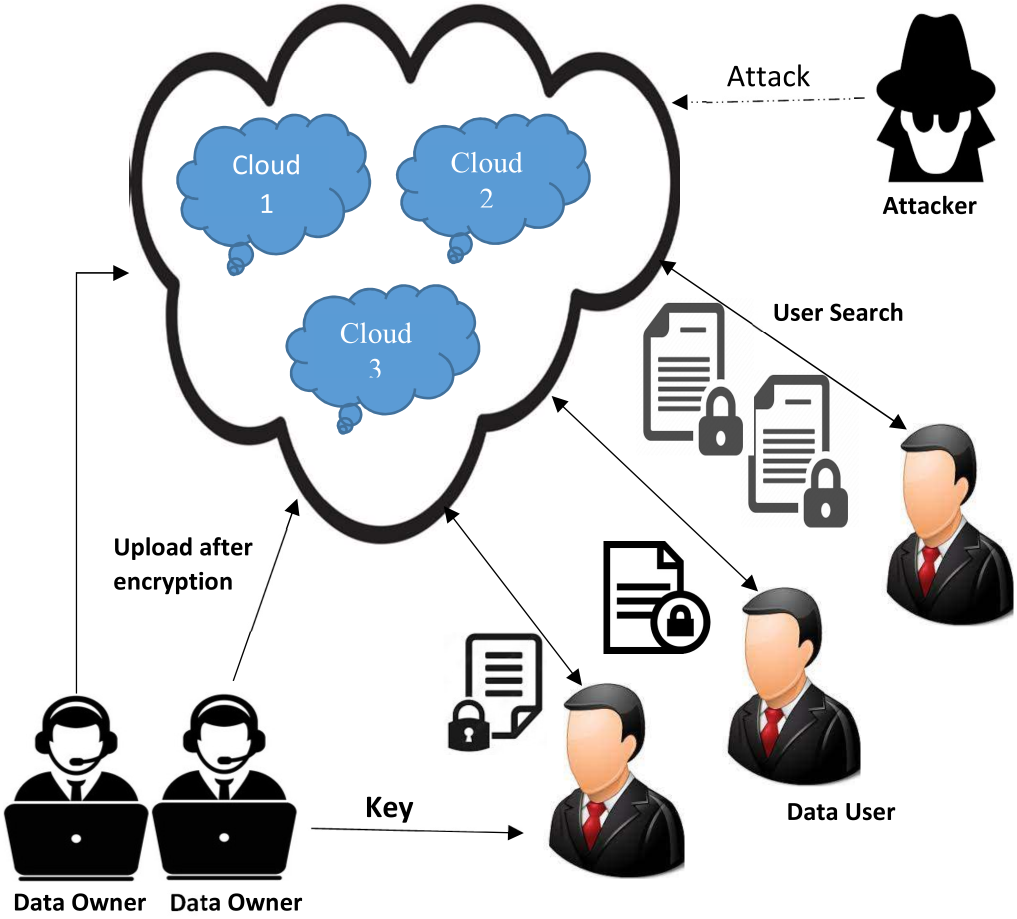

Figure 2: System Model For The Proposed Secure Data Sharing Scheme

In federated cloud environment with multi-owner and multi-user setting, the proposed scheme is implemented. Multiple data owners can send their data to cloud after performing encryption. They also save keywords for search in order to help users to access their data with given trapdoor. With trapdoors, designated users can perform keyword search on the shared documents. Thus controlled access is exercised in the proposed system. The scheme facilitates search facility using trapdoors. In other words, users can access shared documents with the help of search keywords. The proposed scheme exhibits compactness where the trapdoor size does not depend on number of data owners or documents. The scheme also exhibits keyword privacy as attackers cannot gain sensitive information from keywords. Data owners provide aggregate keys to users for generating trapdoors. Aggregate keys are generated in such a way that they cannot be forged by adversaries.

3.2 Our Scheme

We proposed complete security scheme that is based on the concepts of aggregate signatures and broadcast encryption. Thus it serves secure data sharing in multi-owner and multi-user setting. Set of users can get access to shared content and they can perform keyword search. The cipher text also has indices of users embedded. Thus the scheme is suitable for searchable encryption. The scheme is also designed to protect keywords from privacy attacks. The construction of aggregate keys is based on the notion of aggregate signatures that helps in fixing the size that is impacted by number of users leading to compactness property. The following are different algorithms proposed to be part of the scheme.

3.2.1 Setup Algorithm

This algorithm is used to generate system parameters that are used in different operations involved in secure data sharing. This algorithm generates bilinear group and bilinear map such as B =(p,G,GT,e(·,·)) where order is denoted by p and it is designed such that G and $2^{\lambda} < p < 2^{\lambda+1}$. It involves specifying number of documents denoted as n. Then a random number generated is employed. It is denoted as g ∈ G and consider α ∈ Zp before actually performing computation of $g_i = g^{(\alpha i)}$ ∈ G. H : $\{0,1\}^* \rightarrow G$ is used to have a hash function. Eventually, public parameters denoted as params = (B,PubK,H) are generated. In the public parameters PubK $=(g,g_1,...,g_n,g_{n+2},...,g_{2n}) \in G^{2n}$.

3.2.2 KeyGen Algorithm

This algorithm uses the parameters obtained through the setup algorithm and generates secret key required for secure data sharing. In the process, it uses β ∈Zp as a random value. Then a secret key is generated which is associated with β.

3.2.3 Encrypt Algorithm

This algorithm is used to perform encryption on given data. It is executed by data owner to protect data. It takes different inputs such as params (generated by setup algorithm), sk (secret key generated by KeyGen algorithm), i and wl for generating encrypted keyword. Towards this end, a random number, denoted as $t_{i,l}$ ∈ Zp, picked. Then, before encrypting a keyword, the algorithm computes desired variables as expressed in Eq. 1.

$$\mathcal{C}_{1,i,l} = g^{t_{i,l}}, \qquad \mathcal{C}_{2,i,l} = \left(g^{\beta} . g_i\right)^{t_{i,l}}, \qquad \mathcal{C}_{3,i,l} = \frac{e\,(H(w_l),\ g)^{t_{i,l}}}{e\,(g1,\ gn)^{t_{i,l}}} \qquad (1)$$

These variables are used by the algorithm to complete encryption process.

3.2.4 Extract Algorithm

Extract is the algorithm used to generate aggregate key. It takes a set of documents S of data owner, secret key sk and params as input and generates k_{agg}. As subset of documents from S (S ⊆ [1,n]) is considered for generating aggregate key as in Eq. 2.

$$k_{agg = \pi_{j \epsilon s g^{\beta}_{n+1-j}}} \qquad (2)$$

3.2.5 Trapdoor Algorithm

This algorithm is meant for generating trapdoor efficiently. It takes system parameters, set of documents, wl and k_{agg} as input and compute trapdoor. For each document in S, the algorithm computes trapdoor as in Eq. 3.

$$Tr = k_{agg} \cdot H(w_l) \qquad (3)$$

3.2.6 Adjust Algorithm

This algorithm is used to compute trapdoor outcome for set of given documents. For each document in S, it produces trapdoor output by using the expression in Eq. 4.

Tri = Tr · ΠjЄS,j6=ign+1−j+i. (4)

3.2.7 Test Algorithm

This algorithm plays important role in the proposed scheme. It helps in matching given document and keyword in the process of secure data access. It takes set of documents S, encrypted data $c_{i,l,}$ and trapdoor outcome as inputs. From the trapdoor outcome, it performs the computation expressed in Eq. 5 for each keyword.

$$\frac{e(Tr_i\,,\ \mathcal{C}_{1,i,l})}{e(\mathcal{C}_{2,i,l}\ ,\ \ pub)} =^{?}\ \mathcal{C}_{3,i,l} \qquad (5)$$

Where $pub = \ \pi_{j \epsilon s\ \ n+1-j}$

The result of the Eq. 5 is either true or false, based on validation test, which determines an action while accessing data from cloud. When compared with existing schemes found in [41] and [42], our scheme has specific advantages in trapdoor generation and improving performance of secure data sharing. The existing schemes used various random numbers per document. In other words,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

they used same random number for each encrypted keyword in the document. The idea of using same random number enables attacker to gain access to original keyword. To address this problem, in the proposed scheme, unique random number is used for each document. Thus, the proposed scheme satisfies correctness and searchability. In the proposed scheme the trapdoor size does not rely on S. The reason for this is expressed as in Eq. 6.

$k_{agg} = \pi_{j\epsilon s g^{\beta}_{n+1-j}}$ ∈ $\mathbb{G}$ and Tr = $k_{agg} \cdot H(w_l)$ ∈ G (6)

As mentioned earlier in this section, the proposed scheme exhibits compactness and privacy of keywords. Besides forging aggregate key by adversaries is not possible.

4. EXPERIMENTAL RESULTS

We built a Java based standalone application to evaluate proposed scheme and compare it with two existing schemes such as Verifiable Searchable Encryption (VSE) [41] and KASE for Group Data Sharing [42]. Each operation is executed for 50 times and average observations are presented in this section.

Table 1: Storage Overhead In Presence Of Many Data Owners

# Data Owners	Storage Overhead (bytes)		
	KASE-GDS	VSE	Proposed
250	25000	25000	500
500	45000	45000	500
750	100000	100000	500
1000	175000	175000	500
1250	225000	225000	500
1500	250000	250000	500

As presented in Table 1, the storage overhead against number of data owners, due to trapdoor generation, is observed for VSE, KASE-GDS and the proposed schemes.

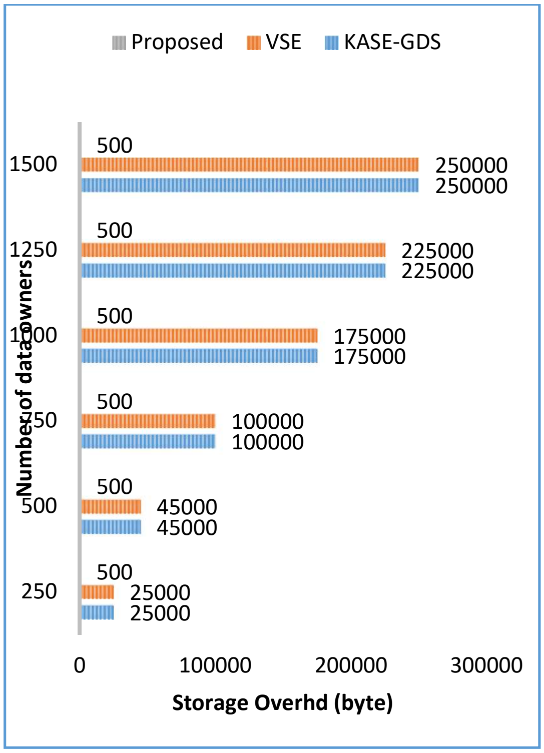

Figure 3: Storage Overhead In Presence Of Many Data Owners

As presented in Figure 3, performance of our scheme is compared against VSE and KSE-GDS in terms of storage overhead. The observations are made in presence of many data owners. As the number of data owners are increased, storage overhead is increased gradually for existing schemes. Due to trapdoor efficiency in the proposed scheme, it does not incur storage overhead. It is observed in the results that the proposed scheme required 500 bytes storage overhead for different number of data owners.

Table 2: Impact Of Number Of Shared Documents On System Parameters' Size

# Shared Documents	Size of System Parameters		
	KASE-GDS	VSE	Proposed
10	30	28	10
50	100	98	75
100	350	330	140
200	700	980	205
300	1050	1630	270
400	1400	2280	335

500	1750	2930	400
600	2100	3580	465
700	2450	4230	530

As presented in Table 2, the impact of number of shared documents on system parameters' size in presence of number of shared documents is observed for VSE, KASE-GDS and the proposed schemes.

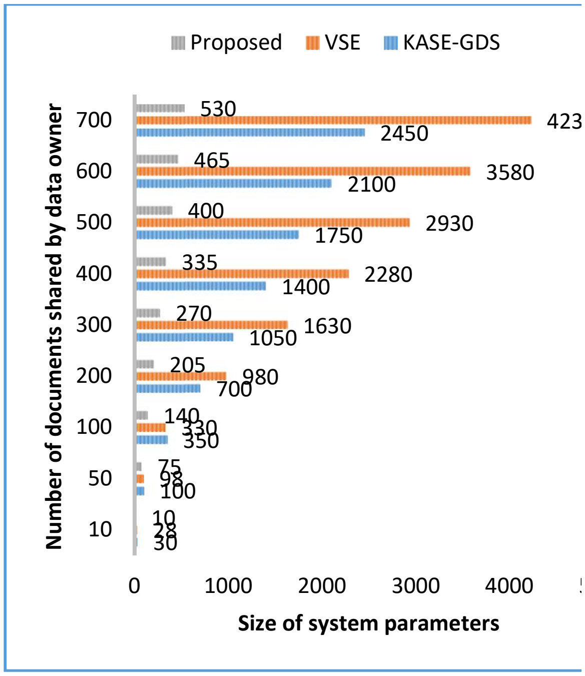

Figure 4: System Parameters' Size In Presence Of Many Shared Documents

As presented in Figure 4, performance of our scheme is compared against VSE and KSE-GDS in terms of parameters' size. The observations are made in presence of many documents shared by data owners. As the number of documents is increased, size of system parameters is increased gradually for existing and proposed schemes. However, the parameters' size of the proposed system is least for each set of documents shared by data owners. When the number of documents is 700, KASE-GDS showed 2450, VSE showed 4230 and the proposed scheme needed 530. Therefore, it is observed that our scheme is better in comparison with existing ones.

Table 3: Computation time analysis in presence of many keywords

# Keywords	**Computation Time (ms)**		
	KASE-GDS	**VSE**	**Proposed**
1	500	100	10
5	500	350	60
10	970	900	100
100	1500	1000	300
200	2990	2100	500
300	4480	3200	700
400	5970	4300	900
500	7460	5400	1100
600	8950	6500	1300
700	10440	7600	1500

As presented in Table 3, computation time is provided in presence of many keywords for VSE, KASE-GDS and the proposed schemes.

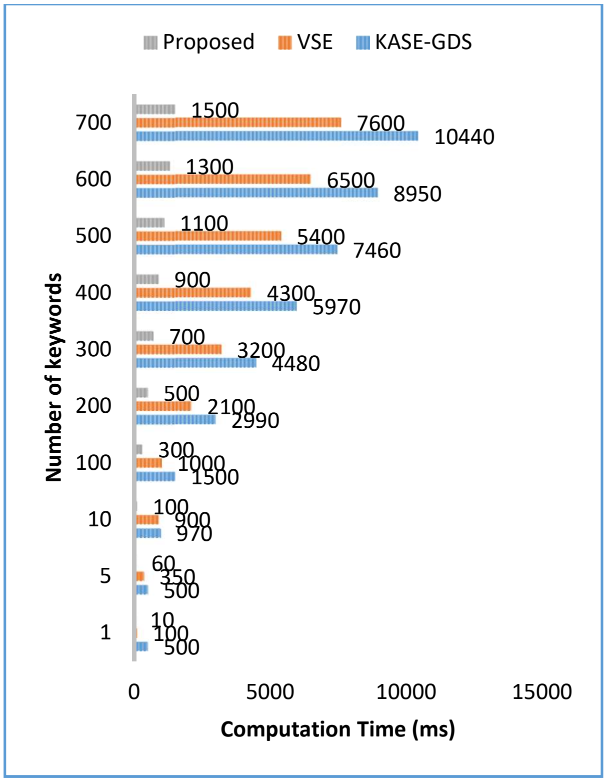

Figure 5: Time Required For Encryption In Presence Of Many Keywords

As presented in Figure 5, performance of our scheme is compared against VSE and KSE-GDS in terms of computation time. The observations are made in presence of many keywords. As the keywords are increased, computation time is increased gradually for existing and proposed schemes. However, the computation time of our system is least for each set of keywords. In presence of 700 keywords, KASE-GDS showed 10440 ms, VSE showed 7600 ms and our scheme needed 1500 ms. Therefore, it is observed that our scheme is better in comparison with existing ones.

Table 4: Computation time of test algorithm in presence of many data owners

# Data Owners	Computation Time of Test Algorithm (ms)		
	KASE-GDS	VSE	Proposed
250	200000	200000	5000
500	300000	300000	5000
750	400000	400000	5000
1000	500000	500000	5000
1250	600000	600000	5000
1500	700000	700000	5000

Table 4 shows test algorithm's computation time in presence of many data owners for VSE, KASE-GDS and the proposed schemes.

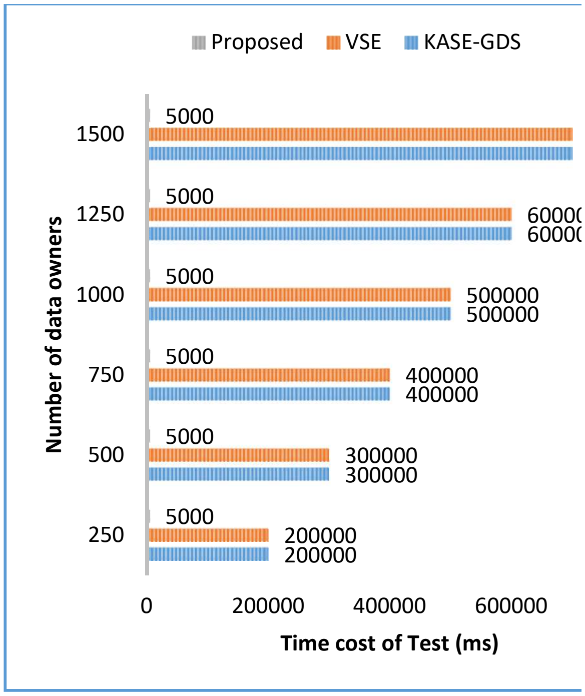

Figure 6: Time cost of test algorithm in presence of many data owners

As presented in Figure 6, performance of our scheme is compared against VSE and KSE-GDS in terms of computation time of test algorithm. The observations are made in presence of many data owners. As presence of data owners increase, computation time is increased gradually for existing schemes. However, our system's computation time requirement is constant irrespective of data owners. When the number of data owners is 1500, KASE-GDS showed 700000 ms, VSE showed 700000 ms and the proposed scheme needed 5000 ms. Therefore, it is observed that our scheme is better in comparison with existing ones.

Table 5: Test algorithm's computation time in presence of many keyword ciphertexts

# Keyword Ciphertexts	Computation Time of Test Algorithm (ms)		
	KASE-GDS	VSE	Proposed
10	10000	10000	10000
50	65000	63000	10000
100	100000	97000	10000
200	200000	580000	10000
300	300000	1063000	10000
400	400000	1546000	10000
500	500000	2029000	10000
600	600000	2512000	10000
700	700000	2995000	10000

As presented in Table 5, test algorithm's computation time is observed in presence of many keyword ciphertexts for VSE, KASE-GDS and the proposed schemes.

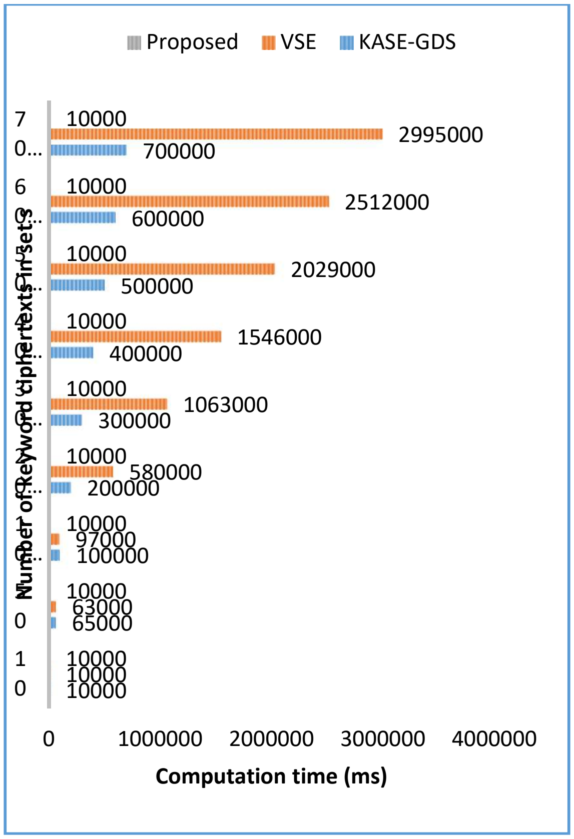

Figure 7: Test algorithm's computation time in presence of many keyword cipher texts

As presented in Figure 7, performance of our scheme is compared against VSE and KSE-GDS in terms of computation time of test algorithm. The observations are made in presence of many keyword ciphertexts. As the number of data keyword ciphertexts is increased, computation time is increased gradually for existing schemes. However, our system's computation time is constant for each number of keyword ciphertexts. When the number of keyword ciphertexts is 700, KASE-GDS showed 700000 ms, VSE showed 2995000 ms and the proposed scheme needed 10000 ms. Therefore, it is observed that our scheme performs better than existing ones.

5. DISCUSSION

The proposed scheme presented in this paper has two significant advantages over prior works. The first contribution is the proposal of a security scheme that is flexible and help in secure group data sharing in cloud. The second contribution is that the scheme is designed for working in federated cloud environment where services of multiple clouds are seamlessly integrated. With the two significant contributions, the proposed scheme is more useful in secure group data sharing in cloud.

5.1 Limitations

The proposed scheme is evaluated in a simulated federated cloud environment. Therefore, it can be evaluated in future with more meaningful testbed to generalize our conclusions. Our scheme can also be improved to have single trapdoor to support users accessing data of many owners.

6. CONCLUSION AND FUTURE WORK

We proposed a security scheme suitable for federated cloud. In such environment data of multiple owners can be shared and trapdoors can be generated by users to perform keyword search to gain access to the desired data. The existing schemes suffer from performance issues in complex scenarios in federated cloud setting. Different algorithms are proposed to realize secure and efficient data sharing in cloud. The proposed scheme is designed to address the problems of existing schemes that involve in key aggregation, searchable encryption and trapdoor generation. Our scheme is evaluated and found that it performs better than existing schemes such as VSE and KASE-GDS. Our scheme has important limitation that could be addressed in our future work. As of now, our scheme needs multiple trapdoors for users to facilitate accessing data of many owners. Though our scheme is performing efficiently over existing ones, the aforementioned drawback is yet to be overcome in future for reducing overhead and improving efficiency further.

REFERENCES

[1] Miao, Yinbin; Tong, Qiuyun; Deng, Robert; Choo, Kim-Kwang Raymond; Liu, Ximeng; Li, Hongwei (2020). Verifiable Searchable Encryption Framework against Insider Keyword-Guessing Attack in Cloud Storage. IEEE Transactions on Cloud Computing, pp.1–1. doi:10.1109/TCC.2020.2989296

[2] Zhou, R., Zhang, X., Wang, X., Yang, G., Dai, H.N. and Liu, M., (2021). Device-oriented keyword searchable encryption scheme for cloud-assisted industrial IoT. IEEE Internet of Things Journal.

[3] Zhang, Q., Fu, M., Huang, Y. and Zhao, Z., (2022). Encrypted Speech Retrieval Scheme Based on Multiuser Searchable Encryption

in Cloud Storage. Security and Communication Networks, pp.1-14.

[4] Fu, Shaojing; Zhang, Chao; Ao, Weijun (2020). Searchable encryption scheme for multiple cloud storage using doubleâ layer blockchain. Concurrency and Computation: Practice and Experience. doi:10.1002/cpe.5860

[5] Sun, Jin; Ren, Lili; Wang, Shangping; Yao, Xiaomin (2019). Multi-Keyword Searchable and Data Verifiable Attribute-Based Encryption Scheme for Cloud Storage. IEEE Access, pp.1–13. doi:10.1109/ACCESS.2019.2917772

[6] Shahien, Toka; Sarhan, Amany M.; Alshewimy, Mahmoud A. M. (2020). Multi-server searchable data crypt: searchable data encryption scheme for secure distributed cloud storage. Journal of Ambient Intelligence and Humanized Computing, pp.1-19. doi:10.1007/s12652-020-02621-8

[7] Xu, Lingling; Sun, Zhiwei; Li, Wanhua; Yan, Hongyang (2020). Delegatable searchable encryption with specified keywords for EHR systems. Wireless Networks, pp.1–13. doi:10.1007/s11276-020-02410-3

[8] D. Sangeetha, S. Sibi Chakkaravarthy, Suresh Chandra Satapathy, V. Vaidehi, Meenalosini Vimal Cruz. (2021). Multi keyword searchable attribute based encryption for efficient retrieval of health Records in Cloud . Multimedia Tools and Applications, pp.1–21. doi:10.1007/s11042-021-10817-z

[9] Mamta, Brij B. Gupta, Kuan-Ching Li, Victor C. M. Leung, Kostas E. Psannis, and Shingo Yamaguchi. (2021). Blockchain-Assisted Secure Fine-Grained Searchable Encryption for a Cloud-Based Healthcare Cyber-Physical System. IEEE/CAA Journal of Automatica Sinica, pp.1–14. doi:10.1109/jas.2021.1004003

[10] Xiao, Tingting; Han, Dezhi; He, Junhui; Li, Kuan-Ching; de Mello, Rodrigo Fernandes (2020). Multi-Keyword ranked search based on mapping set matching in cloud ciphertext storage system. Connection Science, pp.1–18. doi:10.1080/09540091.2020.1753175

[11] Sharma, Dhruti; Jinwala, Devesh (2020). Multi-writer Multi-reader Boolean Keyword Searchable Encryption. Arabian Journal for Science and Engineering, pp.1–21. doi:10.1007/s13369-020-04829-4

[12] Liu, Z.Y., Tseng, Y.F., Tso, R., Chen, Y.C. and Mambo, M., (2021). Identity-Certifying Authority-Aided Identity-Based Searchable Encryption Framework in Cloud Systems. IEEE Systems Journal. Pp.1-12.

[13] Uma Sankararao Varri;Syam Kumar Pasupuleti;K. V. Kadambari; (2021). CP-ABSEL: Ciphertext-policy attribute-based searchable encryption from lattice in cloud storage. Peer-to-Peer Networking and Applications, pp.1290-1302. doi:10.1007/s12083-020-01057-3

[14] Zarezadeh, Maryam; Mala, Hamid; Ashouri-Talouki, Maede (2019). Multi-keyword ranked searchable encryption scheme with access control for cloud storage. Peer-to-Peer Networking and Applications, pp.1–12. doi:10.1007/s12083-019-00736-0

[15] Yao, Yanqing; Zhai, Zhengde; Liu, Jianwei; Li, Zhoujun (2019). Lattice-based Key-Aggregate (Searchable) Encryption in Cloud Storage. IEEE Access, pp.1–13. doi:10.1109/access.2019.2952163

[16] Albrecht, Martin R.; Player, Rachel; Scott, Sam (2015). On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology, 9(3), pp.1–35. doi:10.1515/jmc-2015-0016

[17] Bindel, Nina; Buchmann, Johannes; Göpfert, Florian; Schmidt, Markus (2019). Estimation of the hardness of the learning with errors problem with a restricted number of samples. Journal of Mathematical Cryptology, 13(1), pp.47–67. doi:10.1515/jmc-2017-0040

[18] Pol, Pooja; Priyadarshi, Amrit (2016). Secured Cloud data sharing using auditable Aggregate key. , IEEE, pp.267–272. doi:10.1109/ICAECCT.2016.7942596

[19] Tao Feng and Jiewen Si. (2022). Certificateless Searchable Encryption Scheme in Multi-User Environment. MDPI, pp.1-10.

[20] K.Manohar, R. Anil Kumar, N.Parashuram. (2015). Key Aggregate Searchable Encryption for Group Data Sharing Via Cloud Data Storage. International Journal of Computer Engineering In Research Trends. 2(12), pp.1132-1136.

[21] K.ANUSHA1 , V.LALITHA2 , P.SIVA KUMAR3 , S.S.V.R KUMAR.A. (2016). Key- Aggregate Searchable Encryption (KASE) For Group Data Sharing Via Cloud Storage. International Journal of Computer

Science and Mobile Computing. 5(4), p. 370 – 374.

[22] Dr. V. Goutham, Lalbahadur Kethavath and K.Swetha. (2016). KEY AGGREGATE SEARCHABLE ENCRYPTION WITH SECURE AND EFFICIENT DATA SHARING IN CLOUD. International Journal of Computer Engineering & Technology (IJCET). 7(4), pp.41–47.

[23] MASAHIRO KAMIMURA, NAOTO YANAI, SHINGO OKAMURA and JASON PAUL CRUZ. (2019). Key-Aggregate Searchable Encryption, Revisited: Formal Foundations for Cloud Applications, and Their Implementation. IEEE TRANSACTIONS and JOURNALS. 4, pp.1-17.

[24] Smital Erande1 and V. S. Ranmalkar. (2015). Key-Aggregate Searchable Encryption (KASE) for Group Data Sharing via Cloud Storage. International Journal of Science and Research (IJSR). 5(12), pp.1793-1797.

[25] Mugdha Rane, Shruti Kokate, Jyoti Sonawane, Pranita Panchal and Meenkshi. (2017). Key-Aggregate Searchable Encryption (KASE) Through trapdoor for Group Data Sharing via Cloud Storage. IJARIIE. 3(2), pp.3491-3494.

[26] JoonYoung Lee, MyeongHyun Kim, JiHyeon Oh, YoungHo Park and KiSung. (2021). A Secure Key Aggregate Searchable Encryption with Multi Delegation in Cloud Data Sharing Service. MDPI, pp.1-20.

[27] Sanjana M. Kavatagi and Dr. Rashmi Rachh. (2017). Implementation of Searchable Encryption using Key Aggregation for Group Data Sharing in Cloud. SSRG International Journal of Computer Science and Engineering (SSRG-IJCSE). 4(8), pp.1-4.

[28] Sonkar, S. K. and Wakchaure, S. P. (2016). Group data searching and sharing using key aggregate cryptosystem, IEEE, pp.169–174. doi:10.1109/ICGTSPICC.2016.7955291

[29] Pratiksha Gadekar and Prof. Ugale Pradip. (2017). Secure Key Aggregate Searchable Encryption (KASE) and Efficient Data Sharing in Cloud. IJEDR. 5(2), pp.1-5.

[30] R. Rakesh and Anoop S. (2016). Review on Key-Aggregate Searchable Encryption (KASE) for Group Data Sharing. International Journal of Computer Applications. 139(2), pp.1-7.

[31] Niraj Jaywant Bankar, Prof. Kore Kunal Sidramappa. (2019). Efficient Key-Aggregate Searchable Encryption (KASE) for Group Data Sharing via Cloud Storage. International Journal of Innovative Research in Science, Engineering and Technology. 8(6), pp.1-6.

[32] Miss. Swati V.Thakre1, Prof. K.K.Chhajed2, Prof.V.B.Bhagat. (2017). Review on Key Based Encryption Scheme for Secure Data Sharing on Cloud. International Research Journal of Engineering and Technology (IRJET). 4(1), pp.1-4.

[33] Sunaina Bagga. (2021). Research on Decryption Methodologies and Key Aggregate Searchable Encryption for Data Security Storage in Cloud. International Journal of Innovative Research in Computer Science & Technology (IJIRCST). 9(6), pp.314-319.

[34] S. Brindha, M. Raghini, R. Birundha and V. R. Hemalatha. (2017). EKASE: Enhanced KeyAggregate Searchable Encryption for Multi-owner Data Sharing via Cloud. International Journal of Engineering Research & Technology (IJERT). 5(9), pp.1-5.

[35] Krishna Samalla. (2017). A Novel Algorithm for Multiple Data Sharing Via Cloud Storage. International Journal of Engineering and Advanced Technology (IJEAT). 6(3), pp.1-5.

[36] NABEELA SUMEEN, S. SHALINI and M. SWAPNA. (2016). Scalable Data Sharing in Cloud Storage with Key Aggregate Cryptosystem. IJATIR. 8(14), pp.2741-2745.

[37] M.Gayathri and G.Srinaganya. (2023). SECURE AND EFFICIENT CLOUD STORAGE BASED ON KEY AGGREGATE SEARCHABLE ENCRYPTION FOR GROUP DATA SHARING. International Journal of Novel Research and Development. 8(2), pp.a510-a513.

[38] Wang, Xuqi; Cheng, Xiangguo (2019). Efficient Verifiable Key-Aggregate Keyword Searchable Encryption for Data Sharing in Outsourcing Storage. IEEE Access, pp.1–12. doi:10.1109/ACCESS.2019.2961169

[39] Wang, Xuqi; Xie, Yu; Cheng, Xiangguo; Jiang, Zhengtao (2019). An Efficient Key-Aggregate Keyword Searchable Encryption for Data Sharing in Cloud Storage, IEEE,

pp.1–6. doi:10.1109/gcwkshps45667.2019.9024540

[40] Guo, Cheng; Luo, Ningqi; Bhuiyan, Md Zakirul Alam; Jie, Yingmo; Chen, Yuanfang; Feng, Bin; Alam, Muhammad (2017). Key-aggregate authentication cryptosystem for data sharing in dynamic cloud storage. Future Generation Computer Systems, pp.1–30. doi:10.1016/j.future.2017.07.038

[41] Su, Yaping; Wang, Jianfeng; Wang, Yunling; Miao, Meixia (2019). Efficient Verifiable Multi-Key Searchable Encryption in Cloud Computing. IEEE Access, 7, pp.141352–141362. doi:10.1109/access.2019.2943971

[41] Tong Li, Zheli Liu, Ping Li, Chunfu Jia, Zoe L Jiang, and Jin Li. Verifiable searchable encryption with aggregate keys for data sharing in outsourcing storage. In Australasian Conference on Information Security and Privacy, pages 153–169. Springer, 2016.

[42] Cui BJ, Liu ZL, Wang LY, 2016. Key-aggregate searchable encryption (KASE) for group data sharing via cloud storage. IEEE Trans Comput, 65(8):2374-2385. https://doi.org/10.1109/TC.2015.2389959

[43] Chander, Nenavath, and Mummadi Upendra Kumar. "Metaheuristic feature selection with deep learning enabled cascaded recurrent neural network for anomaly detection in Industrial Internet of Things environment." Cluster Computing 26.3 (2023): 1801-1819.

[44] Chander, Nenavath, and M. Upendra Kumar. "MACHINE LEARNING BASED OUTLIER DETECTION TECHNIQUES FOR IoT DATA ANALYSIS: A COMPREHENSIVE SURVEY."

[45] CHANDER, NENAVATH, and MUMMADI UPENDRA KUMAR. "METAHEURISTICS WITH DEEP CONVOLUTIONAL NEURAL NETWORK FOR CLASS IMBALANCE HANDLING WITH ANOMALY DETECTION IN INDUSTRIAL IOT ENVIRONMENT." Journal of Theoretical and Applied Information Technology 101.10 (2023).

[46] Chander, Nenavath, and M. Upendra Kumar. "Comparative Analysis on Deep Learning Models for Detection of Anomalies and Leaf Disease Prediction in Cotton Plant Data." Congress on Intelligent Systems. Singapore: Springer Nature Singapore, 2022.

[47] Chander, N., Upendra Kumar, M. “Metaheuristic feature selection with deep learning enabled cascaded recurrent neural network for anomaly detection in Industrial Internet of Things environment”, Cluster Computing, 2022.

[48] RAVI, ESLAVATH, and MUMMADI UPENDRA KUMAR. "A COMPARATIVE STUDY ON MACHINE LEARNING AND DEEP LEARNING METHODS FOR MALWARE DETECTION." Journal of Theoretical and Applied Information Technology 100.20 (2022). https://doi.org/10.5281/zenodo.10475511

[49] Ravi, Eslavath, and Mummadi Upendra Kumar. "Android malware detection with classification based on hybrid analysis and N-gram feature extraction." International Conference on Advancements in Smart Computing and Information Security. Cham: Springer Nature Switzerland, 2022. https://doi.org/10.1007/978-3-031-23095-0_13

[50] Eslavath, Ravi, and Upendra Kumar Mummadi. "ENSIC: Feature Selection on Android Malware Detection Attributes Using an Enhanced Non-Linear SVM Integrated with Cross Validator." International Journal of Intelligent Systems and Applications in Engineering 12.2 (2024): 495-504. https://doi.org/10.5281/zenodo.10491837

[51] Ravi, Eslavath, and Mummadi Upendra Kumar. " A Novel Mechanism for Tuning Neural Network for Malware Detection in Android Device." International Conference on Advancements in Smart Computing and Information Security. Cham: Springer Nature Switzerland, 2023.

[52] Kumar, M.U., Kumar, D.S., Rani, B.P., Rao, K.V., Prasad, A.V.K., Shravani, D. “Dependable Solutions Design by Agile Modeled Layered Security Architectures”, Advances in Computer Science and Information Technology. Networks and Communications. CCSIT 2012. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, 2012 vol 84. Springer, Berlin, Heidelberg.

[53] Shravani, D., Suresh Varma, P., Padmaja Rani, B., Upendra Kumar, M., Krishna

Prasad, A.V.: Designing dependable web services security architecture solutions. In: Wyld, D.C., Wozniak, M., Chaki, N., Meghanathan, N., Nagamalai, D. (eds.) CNSA 2011. CCIS, vol. 196, pp. 140–149. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22540-6_14

[54] Krishna Prasad, A. V., Ramakrishna, S., Padmaja Rani, B., Upendra Kumar, M., Shravani, D.: Designing dependable business intelligence solutions using agile web services mining architectures. In: Das, V.V., Thomas, G., Lumban Gaol, F. (eds.) AIM 2011. CCIS, vol. 147,pp. 301–304. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20573-6_51

[55] Mahalakshmi, C.V.S.S., Mridula, B., Shravani, D. “Automatic water level detection using IoT” Satapathy, S., Raju, K., Shyamala, K., Krishna, D., Favorskaya, M. (eds.) Advances in Decision Sciences, Image Processing, Security and Computer Vision. LAIS, vol. 4. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-24318-0_76

[56] Ahmad, S.S., Khan, A., Kawadkar, P., Khan, I., Kumar, M.U., Shravani, D. (2022). A Machine Learning Framework for Automatic Detection of Malware. In: Rajagopal, S., Faruki, P., Popat, K. (eds) Advancements in Smart Computing and Information Security. ASCIS 2022. Communications in Computer and Information Science, vol 1760. Springer, Cham. https://doi.org/10.1007/978-3-031-23095-0_6.

[57] Ali, M.M., Qaseem, M.S., Ahmad, S.S. (2023). Rumour Detection Model for Political Tweets Using ANN. In: Kumar, A., Ghinea, G., Merugu, S. (eds) Proceedings of the 2nd International Conference on Cognitive and Intelligent Computing. ICCIC 2022. Cognitive Science and Technology. Springer, Singapore. https://doi.org/10.1007/978-981-99-2742-5_15

[58] Iffath, N., Mummadi, U. K., Taranum, F., Ahmad, S. S., Khan, I., & Shravani, D. (2024, February). Phishing website detection using ensemble learning models. In AIP Conference Proceedings (Vol. 3007, No. 1). AIP Publishing. https://doi.org/10.1063/5.0192754

[59] Khanum, S. N. A., Mummadi, U. K., Taranum, F., Ahmad, S. S., Khan, I., & Shravani, D. (2024, February). Emotion recognition using multi-modal features and CNN classification. In AIP Conference Proceedings (Vol. 3007, No. 1). AIP Publishing. https://doi.org/10.1063/5.0192751

[60] Ahmad, S. S., Tejaswi, S., Latha, S. B., Kumari, D. S., Prasad, S. D. V., & Bethu, S. (2023, December). Deep learning based mitosis detection for breast cancer prognosis. In AIP Conference Proceedings (Vol. 2938, No. 1). AIP Publishing. https://doi.org/10.1063/5.0181568

[61] Ahmad, S. S., Shailaja, M., Latha, S. B., Bethu, S., & Varaprasad, S. D. (2023, December). Network security and digital forensics using deep learning. In AIP Conference Proceedings (Vol. 2938, No. 1). AIP Publishing. https://doi.org/10.1063/5.0181569

[62] AHMAD, S. S., DESHMUKH, A., SABA, M., KHAN, I., SHRAVANI, D., & KUMAR, M. U. (2023). ENHANCING AGILE DEVELOPMENT WITH SECURITY INTEGRATION: INTRODUCING THE HSSCRUM FRAMEWORK FOR OPTIMIZED AND SECURE SOFTWARE DEVELOPMENT. Journal of Theoretical and Applied Information Technology, 101(21).

[63] B. Raj et al. (Eds.): ICETE 2023, AER 223, pp. 484–493, 2023. https://doi.org/10.2991/978-94-6463-252-1_51

[64] D. Shravani, Imtiyaz Khan, Amogh Deshmukh, Veeramalla Anitha, Masrath Saba, & Syed Shabbeer Ahmad. (2023). LISF: A Security Framework for Internet of Things (IoT) Integrated Distributed Applications. Journal of Advanced Zoology, 44(4), 537–547. https://doi.org/10.17762/jaz.v44i4.1985

[65] Imtiyaz Khan, Syed Shabbeer Ahmad, Shaik Neeha, Asad Hussain Syed, Sayyada Mubeen, “A Deep Reinforcement Learning Framework for Task Scheduling for Leveraging Energy Efficiency in Cloud Computing”, B. Raj et al. (Eds.): ICETE 2023, AER 223, pp. 484–493, 2023. https://doi.org/10.2991/978-94-6463-252-1_51

[66] Imtiyaz Khan et al,"A Deep Reinforcement Learning Based Framework for Task Scheduling for Enhancing Efficiency in Cloud Computing", Indian Institute of Industrial Engineering Journal ISSN NO 0970-2555 UGC Care Gr I Sr No 155 (Sciences) Impact Factor 6.82, Volume 52, Issue 2, No 1, March 23 pp 7-20

[67] Imtiyaz Khan et al,"An AI Enabled Framework for Optimal Load Balancing in Cloud Towards Best Resource Utilization and Efficiency",GIS Science Journal ISSN No 1869-9391,UGC Care Approved Journal Group II journal Volume 10 Issue 3 March 2023 pp 989-1001

[68] Boddupally Janaiah, et al. (2023). Artificial Intelligence Enabled Methods for Human Action Recognition using Surveillance Videos. International Journal on Recent and Innovation Trends in Computing and Communication, 11(9), 3937–3945. https://doi.org/10.17762/ijritcc.v11i9.9734

[69] Surender Mogilicharla,Upendra Kumar Mummadi; Grain quality analysis from the image through the approaches of segmentation. *AIP Conf. Proc.*20 February 2024; 3007 (1): 070001.https://doi.org/10.

[70] Surender Mogilicharla, Upendra Kumar Mummadi; The literature survey: Precision agriculture for crop yield optimization. AIP Conf. Proc. 20 February 2024; 3007 (1): 090005. https://doi.org/10.

[71] Mariyam, Ayesha et al. "A literature survey on recurrent attention learning for text classification." IOP Conference Series: Materials Science and Engineering 1042 (2021): page no. 1-4.

[72] Mariyam, Ayesha, Sk Altaf Hussain Basha, And S. Viswanadha Raju. "A Learning Based Optimized Hybrid Model For Efficient And Scalable Long Document Classification." Journal Of Theoretical And Applied Information Technology 101.19 (2023).

[73] Mariyam, Ayesha, Sk Althaf Hussain Basha, and S. Viswanadha Raju. "Industry 4.0: augmented reality in smart manufacturing industry environment to facilitate faster and easier work procedures." *Cloud Analytics for Industry 4.0* 6 (2022): 141.

[74] Mariyam, Ms Ayesha, SK Althaf Hussain Basha, and S. Viswanadha Raju. "On Optimality of Long Document Classification using Deep Learning."

[75] Mariyam, Ayesha, SK Althaf Hussain Basha, and S. Viswanadha Raju. "Long Document Classification using Hierarchical Attention Networks." International Journal of Intelligent Systems and Applications in Engineering 11.2s (2023): 343-353.