DROPS DIVISION AND REPLICATION OF DATA IN CLOUD FOR OPTIMAL PERFORMANCE AND SECURITY

Dr. Abdul Khadeer[1], Fatima Khadar Basha Shaik[2]

[1] Associate Professor, Department of CSE, Deccan College of Engineering and Technology, Affiliated to Osmania University, Hyderabad, Telangana, India. Email: abdulkhadeer@deccancollege.ac.in

[2] PG Scholar, Department of CSE, Deccan College of Engineering and Technology, Affiliated to Osmania University, Hyderabad, Telangana, India. Email: fatimakhadar.in@gmail.com

Abstract: This poses security problems when data has been outsourced to a third party administrative authority like it is with cloud computing. The cause of the data breach may be attacks by other users and cloud nodes. The data in the cloud is therefore required to be secured with high security. Nevertheless, when it comes to the method of security, optimization of the time spent on retrieving data should also be taken into account. We offer Division and Replication of Data in the Cloud for Optimal Performance and Security (DROPS) in this study, to overcome the security and performance issues. In the DROPS strategy, a file is divided into fragments, and the information is replicated between the cloud nodes. To ensure that, even in case of a successful attack, no substantial information is revealed to the attacker, every node only has one fragment of a given data file. Graph T-coloring is also used to ensure that the nodes that store the pieces are separated by some distance to ensure that the attacker does not calculate the whereabouts of the pieces. The DROPS approach, also, unloads the system of computationally expensive approaches, as it does not rely on traditional cryptographic algorithms to protect data. We prove that any chance of identifying and attacking all nodes that contain pieces of a single file is very low. Also, we compare the performance of the DROPS methodology with 10 other methods. The higher level of security with slight performance overhead was observed.

*"**Keywords**: —Centrality, cloud security, fragmentation, replication, performance".*

I. INTRODUCTION

The idea of cloud computing has transformed the way individuals and companies operate and maintain IT infrastructure remarkably. Some of the basic features of cloud computing include on-demand self-service, broad-network access, resource pooling, fast flexibility, and service utilization which is measurable [1]. Due to low costs, low operation overhead, and high scaling, cloud computing is an option of choice to an individual or a business [2]. Nonetheless, there are severe security challenges associated with these advantages that make cloud computing systems not to be extensively utilized [3].

The dangers to cloud security may be categorized into three groups: risks associated with core technologies (session riding and virtual machine (VM) escape), services (SQL injection and poor authentication schemes), and cloud inherent features (IP-based attacks and vulnerabilities in data recovery) [4]. Lack of user isolation within the multi-tenant virtualized system may lead to virtual machine (VM) escapes, whereby the hacked VM can access the data of other VMs, thus giving way to data security and confidentiality [5]. Also, shared resources can be reused in elastic cloud systems, which provides attackers with the opportunity to access the data that is left over [6].

Moreover, due to the decentralization of cloud computing, a single weak system can affect the overall security of the system. The asset security of the cloud asset is now reliant on the collective strength of all the interconnected organizations and not merely on the precautions that the individual users have taken [7]. Poor media sanitization and insufficient inter-tenant isolation might additionally expose sensitive user data and cause severe privacy breaches [8].

These concerns prompt the need to have strong security measures against outsourced data on public clouds. Besides avoiding unwanted access, these should ensure that even possible breach is made to a minimum and the amount of data that can be accessed is also minimized increasing the cost and effort to make the exploitation effective.

II. RELATED WORK

Jansen [9] in his book Cloud Hooks: Security and Privacy Issues in Cloud computing has discussed the basic security problems in cloud computing. He went to the security vulnerability that are brought about by the simple construction of the cloud systems. Jansen also pointed out that cloud systems require robust hooks or mechanisms that are able to constantly monitor and implement security rules to protect sensitive user data due to their complexities and dynamism. Also, he demanded more security-conscious cloud service architecture and warned of privacy implications of virtualization and multi-tenancy.

Juels and Opera [10] proposed the new methods to enhance the availability and security of data hosted in clouds. To ensure that information stored in the cloud can be retrieved in a consistent manner without being tampered, they came up with the Proofs of Retrievability (PoR) and Proofs of Data Possession (PDP). These methods are particularly useful when the user must know whether his or her data is correct without having to download the entire datasets. Their approach reduces the risks associated with third-party data host and boosts the trust in the open cloud services.

Kappes, Hatzieleftheriou and Anastasiadis [11] introduced Dike which is a virtualization-sensitive access control system that is designed to run on multitenant file systems. They indicated that existing access control mechanisms are not sufficient in the virtualized environment where sharing of resources is the default scenario. To eliminate this, Dike provides fine-grained access control which takes into account the virtualization layer. Isolation is critical to security in multitenant clouds and this is enhanced by it and prevents unwanted cross-tenant usage.

Kaufman [12] provided a practical insight into the cloud data security in his paper Data Security in the World of Cloud Computing. He studied the consequences of data transfers to a cloud services provider and the risks that are posed by such actions as insider risks, data loss, and breach. Strong authentication, encryption, and legal measures were the tactics suggested by Kaufman as the necessary means of protecting data in the cloud. Moreover, he emphasized on the shared responsibility paradigm where the user and the provider have a role to play towards the overall cloud security.

Khan and Ahmad [13] provided a comparison study of eleven methods of data replication by utilizing heuristics that are applied when data is static. Their work is relevant to cloud settings since replication of data is a vital element in ensuring availability and performance even though it was not necessarily cloud-based. Their work illuminates the process of selecting the effective replication means minimizing the latency and redundancy and maintaining the data consistency.

In a comprehensive evaluation of the security concerns with regard to mobile cloud computing, Khan et al. [14] examined the specific issues which pertain to this field. Among the risks that were mentioned by the authors are unauthorized access to the mobile-cloud interface, data leakage through the mobile applications, and insecure interfaces. They proposed a more detailed security architecture integrated with cloud security, mobile security and communication-level security in order to develop a more comprehensive defensive plan. Their study serves as a blueprint to the development of secure mobile-cloud applications.

To add towards identity protection in mobile cloud computing, Khan et al. [15] have come up with the Enhanced Dynamic Credential Generation Scheme. In order to reduce the risk of identity theft or session hijacking, their implementation generates temporary credentials on-the-fly during every session. The approach enhances secrecy and replay attacks resistance since two cryptographic operations are incorporated with session-specific data. Their methodology is quite useful in applications that frequently utilize sensitive cloud-based information by the mobile user.

III. MATERIALS AND METHODS

The proposed solution, DROPS (Division and Replication of Data in the Cloud for Optimal Performance and Security), solves the two problems of data security and access performance in the cloud setting. The system intentionally divides each user file into numerous pieces in such a way that no particular fragment is left with any meaningful information. These fragments are then spread across a number of cloud nodes which may be computational, storage, physical, or virtual computers [4]. Fragments are replicated only once and enhance fault tolerance and protect against data compromise. Pieces are placed by a graph T-coloring algorithm in order to reduce the chances of attackers co-locating. DROPS employs non-cryptographic scheme of security unlike traditional cryptographic schemes which reduces the cost of computing and enhances retrieval performance [7].

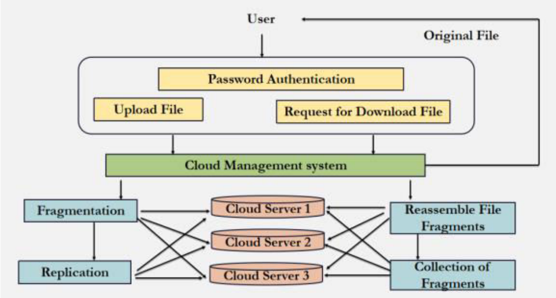


Fig.1 Proposed Architecture

MODULES:

- ❖ System Construction
- ❖ Data Fragmentation
- ❖ Centrality
- ❖ DROPS

MODULES DESCSRIPTION:

System Construction: To measure and implement Division and Replication of Data in the Cloud to the Optimal Performance and Security, we generate the System Construction module in the initial module. We would also recommend another helpful construction which is DROPS. To this end we establish User and Cloud entities. The User entity enables a user to update file blocks that have been uploaded and add new files.

The two types of things that our system model considers include the cloud server and users. The initial user of a given file is the one that uploaded it to the cloud server and the other user is the one who confirmed the ownership of the file but failed to submit the file.

The cloud entity then authenticates the login credentials of users who are then authorised. The data of the users is then stored in blocks.

The performance of our scheme as compared to similar schemes, where m is the size of a single block, n is the number of blocks, and b is the number of challenged blocks, and is given by the asymptotic performance. Moreover, our scheme has better performance in terms of the asymptotic performance, and the only scheme that has a better performance is one with a weak security assurance.

Data Fragmentation: The Data Fragmentation is formed in this module. It will only take one node to be compromised in order to compromise one file. The number of the compromised data can be reduced by dividing segments of data file and storing them on other nodes. Successful attack of one or few nodes will only provide accessibility to a little portion of potentially irrelevant data.

Additionally, the probability of finding pieces at each node is very minimal where an attacker is not sure where the fragments are located. To avoid the attacker acquiring the given data file, we divide it into bits and post to the cloud. With cloud systems, there is a low chance of an attacker accessing a large portion of data. However, the time that is required to retrieve data will increase when every fragment is only added to the system once.

In order to accelerate retrieval of data, fragments may be copied in a manner that will reduce the time it takes to retrieve them without increasing the aforementioned probability.

Centrality: The centrality of a node within a network is the measure of its relative significance to the network. The centrality indicators are more crucial as they are made to increase the retrieval time during replication.

Some of the centrality measurements are proximity centrality, degree centrality, betweenness centrality, eccentricity centrality, and eigenvector centrality. As we use the above three centralities in this paper, we elaborate only on the proximity, betweenness, and the eccentricity centralities.

DROPS: The cloud in the DROPS approach divides and copies the file with the use of the cloud. To ensure that not even a successful attack by a node reveals any meaningful information, the pieces are evenly distributed in a way that there are more than one in a node within a cloud. The DROPS technique is used to maximize security through controlled replication, where only one time every fragment is replicated in the cloud. Controlled replication is much more secure though not as fast as full-scale replication in retrieval.

The data file is uploaded to the cloud by the user through the DROPS approach. Once the file has been received, the cloud manager system, a user-facing server within the cloud which responds to user requests, does the following: (a) fragmentation; (b) first cycle of node selection, which stores one fragment on each selected node; (c) second cycle of node selection to recreate the fragments. It is believed that the cloud manager is an organization that is secure and keeps records of the fragment location.

D) Algorithms:

Cloud computing:

Cloud computing is the use of computer resources (hardware and software) which is a service delivered to a network (typically the Internet).

The name is derived on the cloud-like symbol most commonly employed in system diagrams used as an abstraction of the complex architecture it represents.

Cloud computing provides the data, software, and processing of the users to remote services.

Cloud computing consists of hardware and software resources that are availed online as regulated third-party services. Such services tend to provide their users with access to premium server computer networks and complex computer programs.

Advanced Encryption Standard

Advanced Encryption Standard (AES) is now widely used and popular symmetric encryption algorithm. It is found to be at least 6 times faster than triple DES.

The key size of DES was too small hence it needed replacement. It was believed to be prone to exhaustive key search attack with increasing processing power. Though Triple DES was to fix this problem, it was found to be slow.

The features of AES are as follows −

Symmetric key symmetric block cipher

128-bit data, 128/192/256-bit keys

Stronger and faster than Triple-DES

Provide full specification and design details

Software implementable in C and Java

Operation of AES

This is in contrast to Feistel and AES is an iterative cipher. It is based on the sub-set of permutation network. It involves several interlinked processes some of which involve the movement of bits (permutations) and some of which substitute inputs with some outputs (substitutions).

Interestingly, AES also does not use bits but rather uses a byte in all its calculations. Consequently, AES considers the 128-bit plaintext block to be 16 bytes. These 16 bytes should be put together into four rows and four columns in order to be processed as a matrix.

Unlike DES, AES has variable number of rounds which depend on the length of the key. AES uses 14 rounds with 256-bit keys, 12 rounds with 192-bit keys, and 10 rounds with 128 -bit keys. Each of these rounds takes a distinct 128-bit round key, based on the original AES key.

The schematic of AES structure is given in the following illustration −

Encryption Process

In this case, we will restrict ourselves to a description of a typical AES encryption cycle. Each round includes four subprocesses. The first round process is as illustrated below.

Byte Substitution (SubBytes)

The 16-input bytes are substituted with a fixed table (S-box) that is supplied in the design. The final product is in the form of a 4 row by 4 column matrix.

Shiftrows

All the four rows of the matrix are shifted to the left. The fallen off entries are again added on the right side of the row. The rotation is done in the following way:

In the first row, there is no change.

One (byte) leftwards movement of the second row

The third row has been left-shifted twice.

The fourth row is shifted over to the left side three times.

The resultant product is a new matrix comprising of the same 16 bytes that were moved relative to each other.

MixColumns

Each column of four bytes is now exposed to a certain mathematical operation. That is four bytes of a single column that are inputted in this function which puts the old column into four completely new bytes. The final result is another fresh matrix of 16 more bytes. It is necessary to add that this stage is not observed in the final round.

Addroundkey

The 128 bits of the round key are XORed with the 16 bytes of the matrix which is now considered to be 128 bits. In the case of the final round, the final round is used to generate the ciphertext. Otherwise, we begin a new loop whereby the resultant 128 bits are taken to mean 16 bytes.

Decryption Process

The reverse encrypted ciphertext using the AES is equivalent to decryption. Each round has the four procedures done in the reverse order.

Add round key

Mix columns

Shift rows

Byte substitution

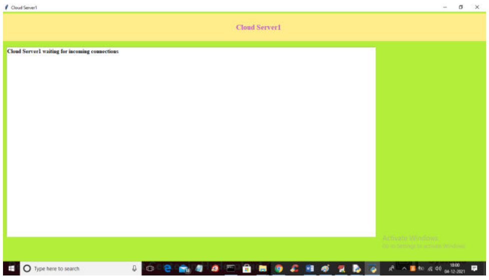
Although closely related, the encryption and decryption algorithms cannot be executed in parallel as the sub-processes of each round run opposite to each other as in the case of Feistel Cipher.

AES Analysis

In modern encryption, AES is typically employed and combined with software and hardware. The cryptanalytic attacking of AES is currently unknown to be effective. Moreover, AES key length flexibility means that it can do certain future-proofing to the innovations in terms of the ability to carry out extensive key searches.

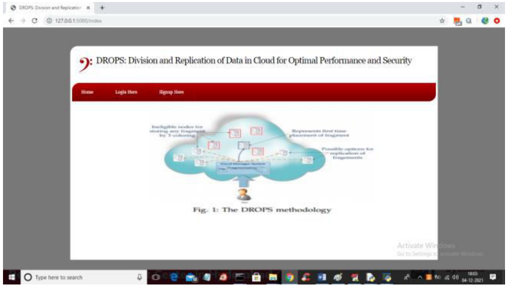
There again, just as it is in the case of DES, the AES security may only be assured when it is implemented adequately and an efficient key management is employed.

IV. RESULTS AND DISCUSSION



Cloud Server1 waiting for incoming connections

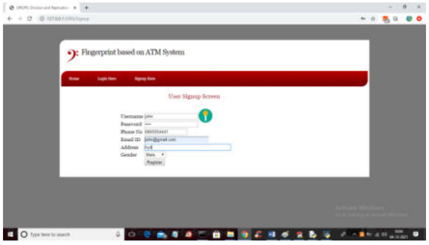
Fig.3: Cloud server1



DROPS: Division and Replication of Data in Cloud for Optimal Performance and Security

Fig. 1: The DROPS methodology

Fig.4: Webpage

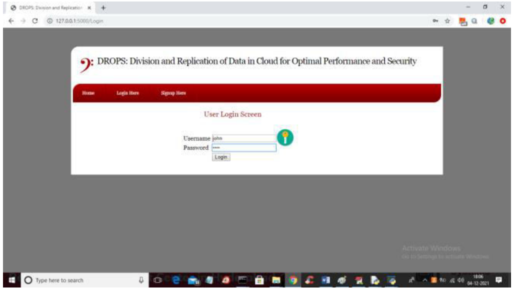


Fingerprint based on ATM System

User Signup Screen

Username john
Password ***
Phone No [illegible]
Email ID john@gmail.com
Address hyd
Gender Male

Fig.5: User signup

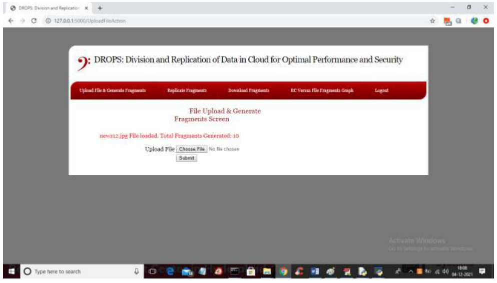


DROPS: Division and Replication of Data in Cloud for Optimal Performance and Security

User Login Screen

Username john
Password ****

Fig.6: User login


DROPS: Division and Replicatio
127.0.0.1:5000/UploadFileAction
Upload File & Generate Fragments
Replicate Fragments
Download Fragments
RC Versus File Fragments Graph
Logout
Choose File
Submit
Type here to search
18:06
04-12-2021

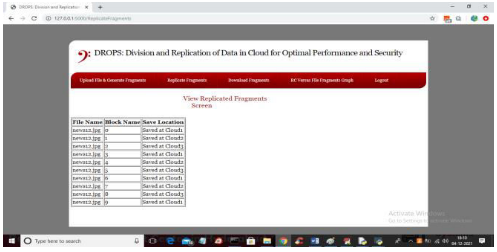
DROPS: Division and Replication of Data in Cloud for Optimal Performance and Security

File Upload & Generate Fragments Screen

news12.jpg File loaded. Total Fragments Generated: 10

Upload File No file chosen

Fig.7: Upload file & generate fragments

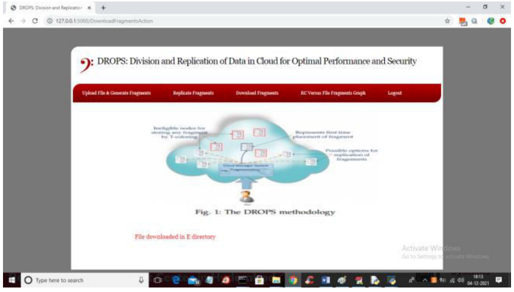

DROPS: Division and Replicatio
127.0.0.1:5000/ReplicateFragments
Upload File & Generate Fragments
Replicate Fragments
Download Fragments
RC Versus File Fragments Graph
Logout
Type here to search
18:10
04-12-2021

DROPS: Division and Replication of Data in Cloud for Optimal Performance and Security

View Replicated Fragments Screen

File Name	Block Name	Save Location
news12.jpg	0	Saved at Cloud1
news12.jpg	1	Saved at Cloud2
news12.jpg	2	Saved at Cloud3
news12.jpg	3	Saved at Cloud1
news12.jpg	4	Saved at Cloud2
news12.jpg	5	Saved at Cloud3
news12.jpg	6	Saved at Cloud1
news12.jpg	7	Saved at Cloud2
news12.jpg	8	Saved at Cloud3
news12.jpg	9	Saved at Cloud1

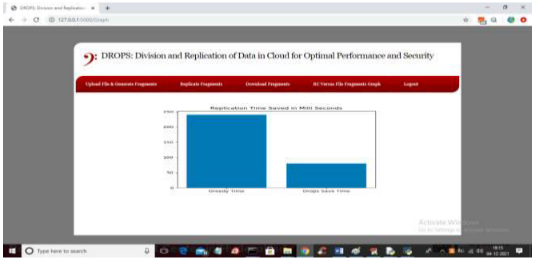
Fig.8: Replicate fragments


DROPS: Division and Replicatio
127.0.0.1:5000/DownloadFragmentsAction
Upload File & Generate Fragments
Replicate Fragments
Download Fragments
RC Versus File Fragments Graph
Logout
Ineligible nodes for storing any fragment by T-coloring
Represents first time placement of fragment
Possible options for replication of fragments
Cloud Manager System
Fragmentation
Fig. 1: The DROPS methodology
Type here to search
18:13
04-12-2021

DROPS: Division and Replication of Data in Cloud for Optimal Performance and Security

File downloaded in E directory

Fig.9: Download fragments



DROPS: Division and Replication of Data in Cloud for Optimal Performance and Security

Fig.10: RC versus File Fragments Graph

V. CONCLUSION

Altogether, the proposed way of detecting hate speech related to cyberbullying on social media platforms is a comprehensive and efficient solution to a major issue. The approach utilizes two various datasets of Cyberbullying and a range of data sources in order to ensure the dependability of hate speech patterns. To extract features, more advanced tools as TFIDF and Countvectorizer are utilized and SMOTE addresses the issue of class imbalance so that the model can withstand skewed distributions of data. The ensemble Voting Classifier is a major area of achieving high accuracy on both datasets since it incorporates Boosted Decision Trees and Bagging Random Forest. This approach enhances better performance and reliability because of the free qualities of every classifier. The algorithm has managed to overcome the ambiguity and other context-related peculiarities of the social media language convincingly, and it displays great potential in detecting various forms of hate speech related to cyberbullying. The proposed method contributes to the establishment of safer online environment and supports positive communication by providing a reliable approach of identifying hate speech. Its successful accomplishment underscores how in the digital age in existence, solutions to complex issues in society can be achieved through the application of machine learning.

The scope of this system in the future will involve exploring the latest advances in deep learning, including neural networks, e.g. LSTM or BERT to learn more and further about context understanding and accuracy in detecting hate speech associated with cyberbullying. More differentiated languages and social media sites should also be included in the data to enhance the generalizability of the model. The environment in the online realms could be brought to safer standards by actively controlling the dangerous material using real-time detection and feedback options.

REFERENCES

1. K. Bilal, S. U. Khan, L. Zhang, H. Li, K. Hayat, S. A. Madani,N. Min-Allah, L. Wang, D. Chen, M. Iqbal, C. Z. Xu, and A. Y.Zomaya, “Quantitative comparisons of the state of the art datacenter architectures,” Concurrency and

Computation: Practice andExperience, Vol. 25, No. 12, 2013, pp. 1771-1783.

2. K. Bilal, M. Manzano, S. U. Khan, E. Calle, K. Li, and A.Zomaya, "On the characterization of the structural robustnessof data center networks," IEEE Transactions on Cloud Computing,Vol. 1, No. 1, 2013, pp. 64-77.
3. D. Boru, D. Kliazovich, F. Granelli, P. Bouvry, and A. Y. Zomaya,"Energy-efficient data replication in cloud computing datacenters,"In IEEE Globecom Workshops, 2013, pp. 446-451.
4. Ganji, M. (2025). Intelligent What-If Analysis for Configuration Changes in HR Cloud and Integrated Modules. International Journal of All Research Education and Scientific Methods, 13(04), 4828–4835. https://doi.org/10.56025/ijaresm.2025.1304254828.
5. Y. Deswarte, L. Blain, and J-C. Fabre, "Intrusion tolerance in distributedcomputing systems," In Proceedings of IEEE ComputerSociety Symposium on Research in Security and Privacy, OaklandCA, pp. 110-121, 1991.
6. Rongali, L. P. (2025). Leveraging Opentelemetry for Enhanced Application Security through Telemetry Data. https://doi.org/10.36227/techrxiv.175790707.71761473/v1
7. B. Grobauer, T.Walloschek, and E. Stocker, "Understandingcloud computing vulnerabilities," IEEE Security and Privacy, Vol.9, No. 2, 2011, pp. 50-57.
8. Todupunuri, A. (2025). Improving Customer Experience With Modern Banking Solutions. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5120615
9. W. K. Hale, "Frequency assignment: Theory and applications,"Proceedings of the IEEE, Vol. 68, No. 12, 1980, pp. 1497-1514.
10. K. Hashizume, D. G. Rosado, E. Fernndez-Medina, and E. B.Fernandez, "An analysis of security issues for cloud computing,"Journal of Internet Services and Applications, Vol. 4, No. 1,2013, pp. 1-13.
11. M. Hogan, F. Liu, A.Sokol, and J. Tong, "NIST cloud computingstandards roadmap," NIST Special Publication, July 2011.
12. Naga Charan Nandigama, "Data-Driven Cyber-Physical Customer Experience Management In Iort-Enabled Banking Infrastructures," International Journal of Data Science and IoT Management System, vol. 2, no. 3, pp. 22–27, Aug. 2023, doi: 10.64751/ijdim.2023.v2.n3.pp22-27.
13. W. A. Jansen, "Cloud hooks: Security and privacy issues incloud computing," In 44th Hawaii IEEE International ConferenceonSystem Sciences (HICSS), 2011, pp. 1-10.
14. A. Juels and A. Opera, "New approaches to security andavailability for cloud data," Communications of the ACM, Vol.56, No. 2, 2013, pp. 64-73.
15. Mallick, P. (2022). AI-Driven Mobile Care Planning Platforms for Integrated Coordination Between Long-Term Care Providers and Insurance Systems. Available at SSRN 6066586.
16. G. Kappes, A. Hatzieleftheriou, and S. V. Anastasiadis, "Dike:Virtualization-aware Access Control for Multitenant Filesystems,"University of Ioannina, Greece, Technical Report No.DCS2013-1, 2013.
17. Bajarang Bhagwat, V. (2023). Optimizing Payroll to General Ledger Reconciliation: Identifying Discrepancies and Enhancing Financial Accuracy. JOURNAL OF ADVANCE AND FUTURE RESEARCH, 1(4). https://doi.org/10.56975/jaafr.v1i4.501636
18. L. M. Kaufman, "Data security in the world of cloud computing,"IEEE Security and Privacy, Vol. 7, No. 4, 2009, pp. 61-64.

19. Vikram, S. (2025). Edge-Aware Federated AI: Scalable LLM Integration for Privacy-Preserving Big Data Networks. 2025 5th International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), 1–7. https://doi.org/10.1109/iceccme64568.2025.11277672
20. S. U. Khan, and I. Ahmad, "Comparison and analysis often static heuristics-based Internet data replication techniques,"Journal of Parallel and Distributed Computing, Vol. 68, No. 2, 2008,pp. 113-136.
21. A. N. Khan, M. L. M. Kiah, S. U. Khan, and S. A. Madani,"Towards Secure Mobile Cloud Computing: A Survey," FutureGeneration Computer Systems, Vol. 29, No. 5, 2013, pp. 1278-1299.
22. A. N. Khan, M.L. M. Kiah, S. A. Madani, and M. Ali, "Enhanceddynamic credential generation scheme for protectionof user identity in mobile-cloud computing, The Journal ofSupercomputing, Vol. 66, No. 3, 2013, pp. 1687-1706 .