JES
Journal of Engineering Sciences

Vol 13, Issue 05, MAY/2022

ISSN NO:0377-9254

# SECURING DATA USING DROPS METHODOLOGY

[1]MOHAMMED SAEED, [2]MOHAMMED MOINUDDIN ANSARI, [3]Dr.MOHAMMED ABDUL BARI

[12]B.E STUDENT, [3]HOD & ASSOCIATE PROFESSOR
DEPARTMENT OF CSE
ISL Engineering College, Bandlaguda, Hyderabad

## ABSTRACT

Outsourcing data to a third-party administrative control, as is done in cloud computing, gives rise to security concerns. The data compromise may occur due to attacks by other users and nodes within the cloud. Therefore, high security measures are required to protect data within the cloud. However, the employed security strategy must also take into account the optimization of the data retrieval time. In this paper, we propose Division and Replication of Data in the Cloud for Optimal Performance and Security (DROPS) that collectively approaches the security and performance issues. In the DROPS methodology, we divide a file into fragments, and replicate the fragmented data over the cloud nodes. Each of the nodes stores only a single fragment of a particular data file that ensures that even in case of a successful attack, no meaningful information is revealed to the attacker. Moreover, the nodes storing the fragments, are separated with certain distance by means of graph T-coloring to prohibit an attacker of guessing the locations of the fragments. Furthermore, the DROPS methodology does not rely on the traditional cryptographic techniques for the data security; thereby relieving the system of computationally expensive methodologies. We show that the probability to locate and compromise all of the nodes storing the fragments of a single file is extremely low. We also compare the performance of the DROPS methodology with ten other schemes. The higher level of security with slight performance overhead was observed.

## 1.INTRODUCTION

The cloud computing paradigm has reformed the usage and management of the information technology infrastructure. Cloud computing is characterized by on-demand self-services, ubiquitous network accesses, resource pooling, elasticity, and measured services. The aforementioned characteristics of cloud computing make it a striking candidate for businesses, organizations, and individual users for adoption. However, the benefits of low-cost, negligible management (from a users perspective), and greater flexibility come with increased security concerns. Security is one of the most crucial aspects among those prohibiting the wide-spread adoption of cloud computing. Cloud security issues may stem due to the core technology′ s implementation (virtual machine (VM) escape, session riding, etc.), cloud service offerings (structured query language injection, weak authentication schemes, etc.), and arising from cloud characteristics (data recovery vulnerability, Internet protocol vulnerability, etc.). For a cloud to be secure, all of the participating entities must be secure. In any given system with multiple units, the highest level of the system′ s security is equal to the security level of the weakest entity. Therefore, in a cloud, the security of the assets does not solely depend on an individual's security measures. The neighboring entities may provide an opportunity to an attacker to bypass the users defenses. The off-site data storage cloud utility requires users to move data in cloud's virtualized and shared environment that may result in various security concerns. Pooling and elasticity of a cloud, allows the physical resources to be shared among many users. Moreover, the shared resources may be reassigned to other users at some instance of time that may result in data compromise through data recovery methodologies. Furthermore, a multi-tenant virtualized environment may result in a VM to escape the bounds of virtual machine monitor (VMM). The escaped VM can interfere with other VMs to have access to unauthorized data. Similarly, cross-tenant virtualized network

access may also compromise data privacy and integrity. Improper media sanitization can also leak customer′ s private data.

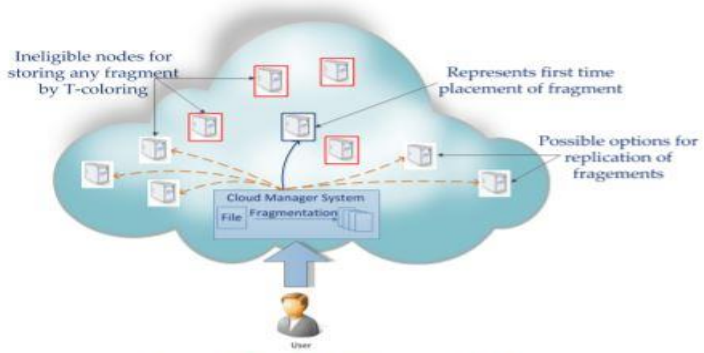



Fig. 1: The DROPS methodology

The data outsourced to a public cloud must be secured. Unauthorized data access by other users and processes (whether accidental or deliberate) must be prevented. As discussed above, any weak entity can put the whole cloud at risk. In such a scenario, the security mechanism must substantially increase an attacker's effort to retrieve a reasonable amount of data even after a successful intrusion in the cloud. Moreover, the probable amount of loss (as a result of data leakage) must also be minimized.

## 2.LITERATURE SURVEY

### 1) On the characterization of the structural robustness of data center networks

**AUTHORS:** K. Bilal, M. Manzano, S. U. Khan, E. Calle, K. Li, and A. Zomaya

Data centers being an architectural and functional block of cloud computing are integral to the Information and Communication Technology (ICT) sector. Cloud computing is rigorously utilized by various domains, such as agriculture, nuclear science, smart grids, healthcare, and search engines for research, data storage, and analysis. A Data Center Network (DCN) constitutes the communicational backbone of a data center, ascertaining the performance boundaries for cloud infrastructure. The DCN needs to be robust to failures and uncertainties to deliver the required Quality of Service (QoS) level and satisfy Service Level Agreement (SLA). In this paper, we analyze robustness of the state-of-the-art DCNs. Our major contributions are: (a) we present multi-layered graph modeling of various DCNs; (b) we study the classical robustness metrics considering various failure scenarios to perform a comparative analysis; (c) we present the inadequacy of the classical network robustness metrics to appropriately evaluate the DCN robustness; and (d) we propose new procedures to quantify the DCN robustness. Currently, there is no detailed study available centering the DCN robustness. Therefore, we believe that this study will lay a firm foundation for the future DCN robustness research.

### 2) Energy-efficient data replication in cloud computing datacenters

**AUTHORS:** D. Boru, D. Kliazovich, F. Granelli, P. Bouvry, and A. Y. Zomaya

Cloud computing is an emerging paradigm that provides computing resources as a service over a network. Communication resources often become a bottleneck in service provisioning for many cloud applications. Therefore, data replication, which brings data (e.g., databases) closer to data consumers (e.g., cloud applications), is seen as a promising solution. It allows minimizing network delays and bandwidth usage. In this paper we study data replication in cloud computing data centers. Unlike other approaches available in the literature, we consider both energy efficiency and

bandwidth consumption of the system, in addition to the improved Quality of Service (QoS) as a result of the reduced communication delays. The evaluation results obtained during extensive simulations help to unveil performance and energy efficiency tradeoffs and guide the design of future data replication solutions.

**3) Intrusion tolerance in distributed computing systems**

**AUTHORS:** Y. Deswarte, L. Blain, and J-C. Fabre

An intrusion-tolerant distributed system is a system which is designed so that any intrusion into a part of the system will not endanger confidentiality, integrity and availability. This approach is suitable for distributed systems, because distribution enables isolation of elements so that an intrusion gives physical access to only a part of the system. In particular, the intrusion-tolerant authentication and authorization servers enable a consistent security policy to be implemented on a set of heterogeneous, untrusted sites, administered by untrusted (but nonconspiring) people. The authors describe how some functions of distributed systems can be designed to tolerate intrusions. A prototype of the persistent file server presented has been successfully developed and implemented as part of the Delta-4 project of the European ESPRIT program.

## III. SYSTEM ANALYSIS

### 3.1 EXISTING SYSTEM:

- Juels et al. presented a technique to ensure the integrity, freshness, and availability of data in a cloud. The data migration to the cloud is performed by the Iris file system. A gateway application is designed and employed in the organization that ensures the integrity and freshness of the data using a Merkle tree. The file blocks, MAC codes, and version numbers are stored at various levels of the tree.
- G. Kappeset. al. approached the virtualized and multi-tenancy related issues in the cloud storage by utilizing the consolidated storage and native access control. The Dike authorization architecture is proposed that combines the native access control and the tenant name space isolation.

#### 3.1.1 DISADVANTAGES OF EXISTING SYSTEM:

- The leakage of critical information in case of improper sanitization and malicious VM is not handled.
- Such schemes do not protect the data files against tempering and loss due to issues arising from virtualization and multi-tenancy.
- The data files are not fragmented and are handled as a single file.

### 3.2 PROPOSED SYSTEM:

- In this paper, we collectively approach the issue of security and performance as a secure data replication problem. We present Division and Replication of Data in the Cloud for Optimal Performance and Security (DROPS) that judicially fragments user files into pieces and replicates them at strategic locations within the cloud.
- The division of a file into fragments is performed based on a given user criteria such that the individual fragments do not contain any meaningful information. Each of the cloud nodes (we use the term node to represent computing, storage, physical, and virtual machines) contains a distinct fragment to increase the data security.
- We develop a scheme for outsourced data that takes into account both the security and performance. The proposed scheme fragments and replicates the data file over cloud nodes.

#### 3.2.1 ADVANTAGES OF PROPOSED SYSTEM:

- The proposed DROPS scheme ensures that even in the case of a successful attack, no meaningful information is revealed to the attacker.
- We do not rely on traditional cryptographic techniques for data security. The non-cryptographic nature of the proposed scheme makes it faster to perform the required operations (placement and retrieval) on the data.

- We ensure a controlled replication of the file fragments, where each of the fragments is replicated only once for the purpose of improved security

**SYSTEM ARCHITECTURE:**

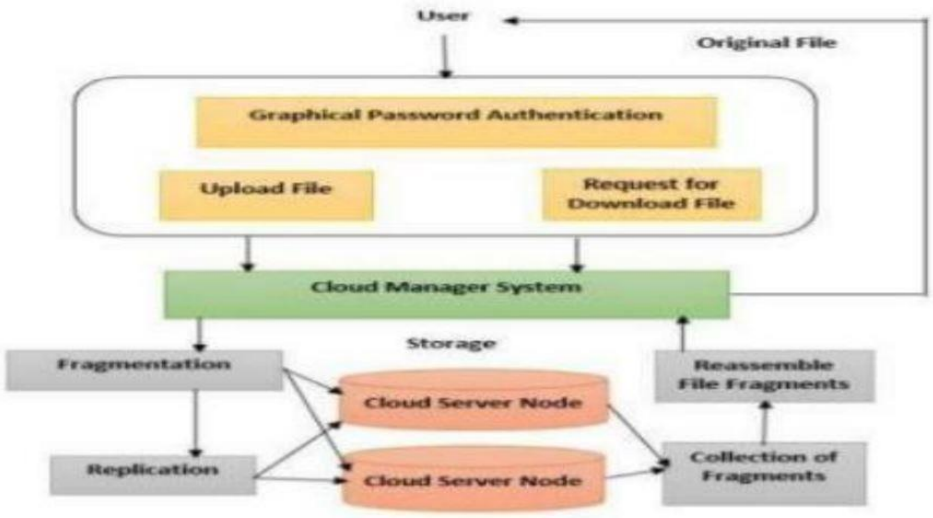



**Figure : DROPS System Architecture**

## IV. IMPLEMENTATION

**MODULES:**

- System Construction
- Data Fragmentation
- Centrality
- DROPS

**MODULES DESCSRIPTION:**

**System Construction:**

- In the first module we develop the System Construction module, to evaluate and implement Division and Replication of Data in Cloud for Optimal Performance and Security and propose an efficient construction called DROPS. For this purpose we develop User and Cloud entities. In User entity, a user can upload a new File, Update uploaded File blocks..
- Our system model considers two types of entities: the cloud server and users. For each file, *original user* is the user who uploaded the file to the cloud server, while *subsequent user* is the user who proved the ownership of the file but did not actually upload the file to the cloud server.
- In the Cloud entity, the cloud first check login authentication of users and then it gives permission for authenticated users and users datas are stored in blocks.
- The asymptotic performance of our scheme in comparison with related schemes, where n denotes the number of blocks, b denotes the number of the challenged blocks, and |m| denotes the size of one block.. Furthermore, the asymptotic performance of our scheme is better than the other schemes except which only provides weak security guarantee.

**Data Fragmentation**

- In this module, we develop the Data Fragmentation. Compromising a single file will require the effort to penetrate only a single node. The amount of compromised data can be reduced by making fragments of a data file and storing them on separate nodes. A successful intrusion on a single or few nodes will only provide access to a

portion of data that might not be of any significance.

- Moreover, if an attacker is uncertain about the locations of the fragments, the probability of finding fragments on all of the nodes is very low. Therefore, we fragment the given Data file and upload in the Cloud so that no attacker will obtain the data file. In cloud systems, the probability for an attacker to obtain a considerable amount of data, reduces significantly. However, placing each fragment once in the system will increase the data retrieval time.
- To improve the data retrieval time, fragments can be replicated in a manner that reduces retrieval time to an extent that does not increase the aforesaid probability.

**Centrality**

- The centrality of a node in a graph provides the measure of the relative importance of a node in the network. The objective of improved retrieval time in replication makes the centrality measures more important.
- There are various centrality measures; for instance, closeness centrality, degree centrality, betweenness centrality, eccentricity centrality, and eigenvector centrality. We only elaborate on the closeness, betweenness, and eccentricity centralities because we are using the aforesaid three centralities in this work.

**DROPS**

- In the DROPS methodology, fragments the file and makes use of the cloud for replication. The fragments are distributed such that no node in a cloud holds more than a single fragment, so that even a successful attack on the node leaks no significant information. The DROPS methodology uses controlled replication where each of the fragments is replicated only once in the cloud to improve the security. Although, the controlled replication does not improve the retrieval time to the level of full-scale replication, it significantly improves the security.
- In the DROPS methodology, user sends the data file to cloud. The cloud manager system (a user facing server in the cloud that entertains user's requests) upon receiving the file performs: **(**a) fragmentation, **(**b) first cycle of nodes selection and stores one fragment over each of the selected node, and **(**c) second cycle of nodes selection for fragments replication. The cloud manager keeps record of the fragment placement and is assumed to be a secure entity.

**Algorithms:**

**AES:**

The Advanced Encryption Standard (AES), also known by its original name Rijndael (Dutch pronunciation: [ˈrɛindaːl]),[3] is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001.[4]

AES is a variant of the Rijndael block cipher[3] developed by two Belgian cryptographers, Vincent Rijmen and Joan Daemen, who submitted a proposal[5] to NIST during the AES selection process.[6] Rijndael is a family of ciphers with different key and block sizes. For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.

AES has been adopted by the U.S. government. It supersedes the Data Encryption Standard (DES),[7] which was published in 1977. The algorithm described by AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data.

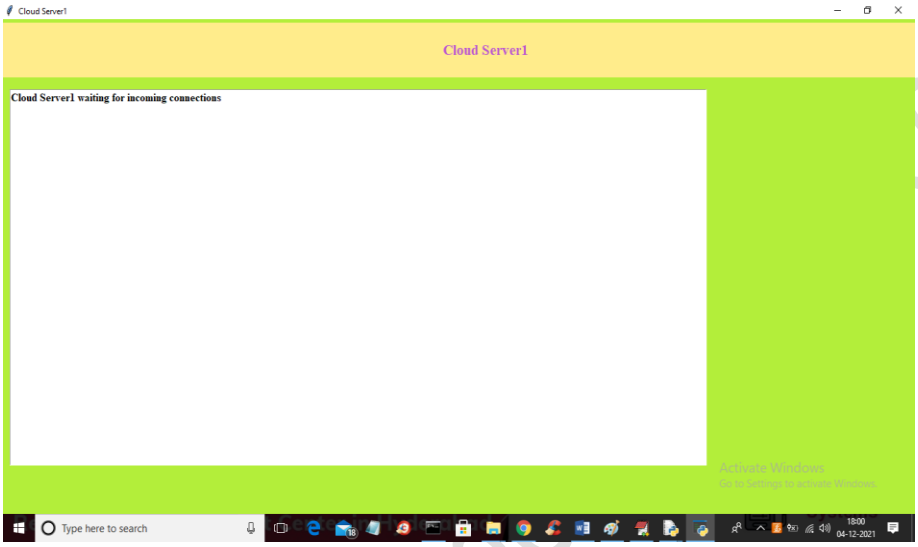
In the United States, AES was announced by the NIST as U.S. FIPS PUB 197 (FIPS 197) on November 26, 2001.[4] This announcement followed a five-year standardization process in which fifteen competing designs were presented and evaluated, before the Rijndael cipher was selected as the most suitable (see Advanced Encryption Standard process for more details).

AES is included in the ISO/IEC 18033-3 standard. AES became effective as a U.S. federal government standard on May 26, 2002, after approval by the U.S. Secretary of

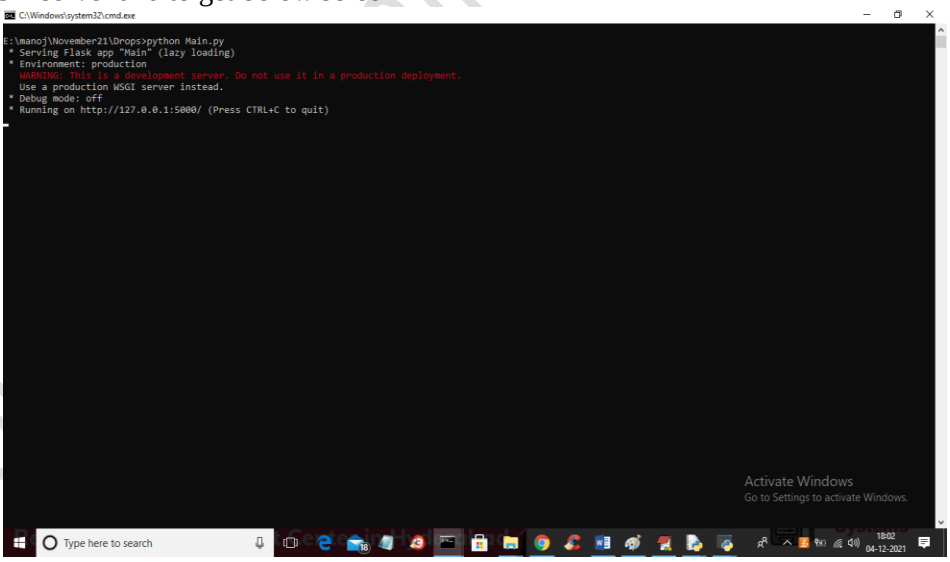
Commerce. AES is available in many different encryption packages, and is the first (and only) publicly accessible cipher approved by the U.S. National Security Agency (NSA) for top secret information when used in an NSA approved cryptographic module (see Security of AES, below).

**V. SCREEN SHOTS**

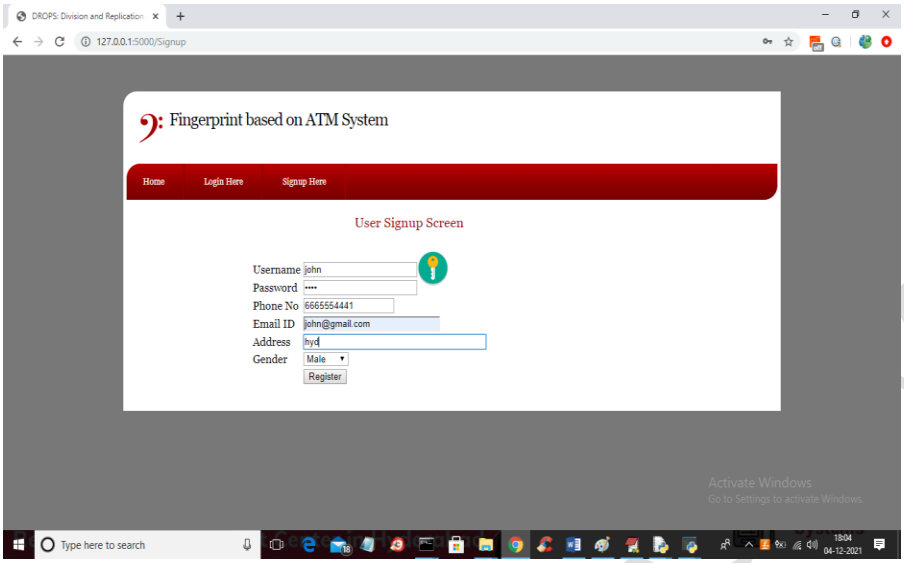
First double click on 'run.bat' file from 'CloudServer1' folder to get below screen and to start first cloud server



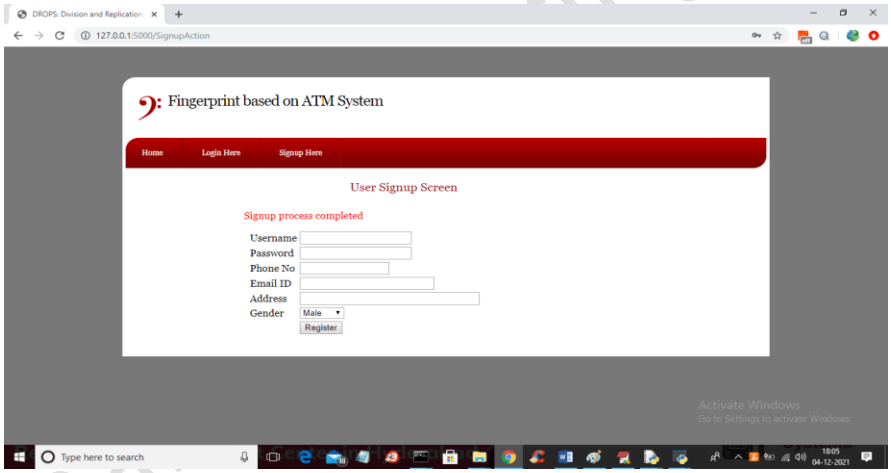

In above screen cloud server3 started. Now double Click on 'run.bat' file from 'Drops' main folder to start python FLASK server and to get below screen



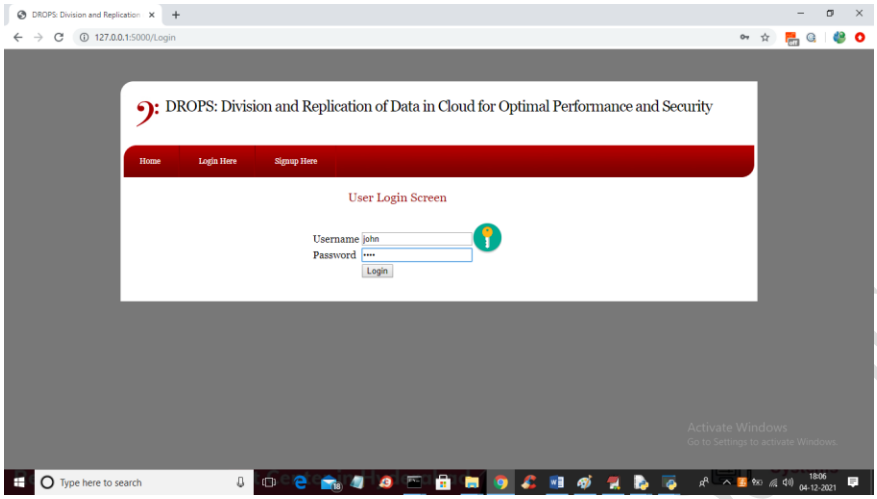
Now click on 'Signup Here' link to add new user and to get below screen



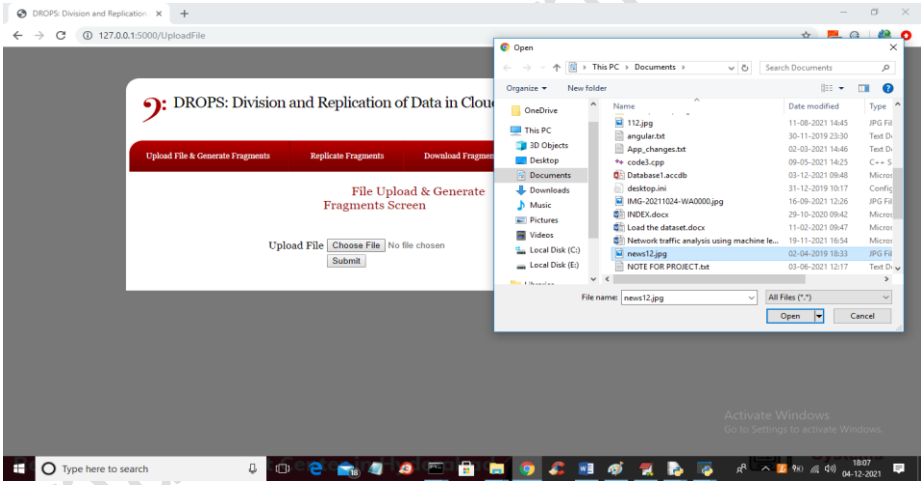

In above screen one user is signing and after signing will get below screen



In above screen user signup process completed and now click on 'Login Here' link to get below login screen

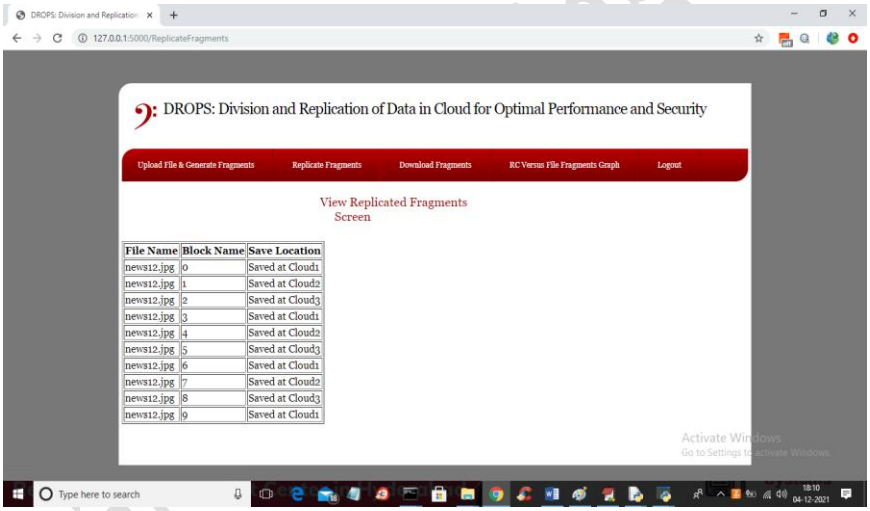



In above screen click on 'Upload File & Generate Fragments' link to upload file and to get below screen

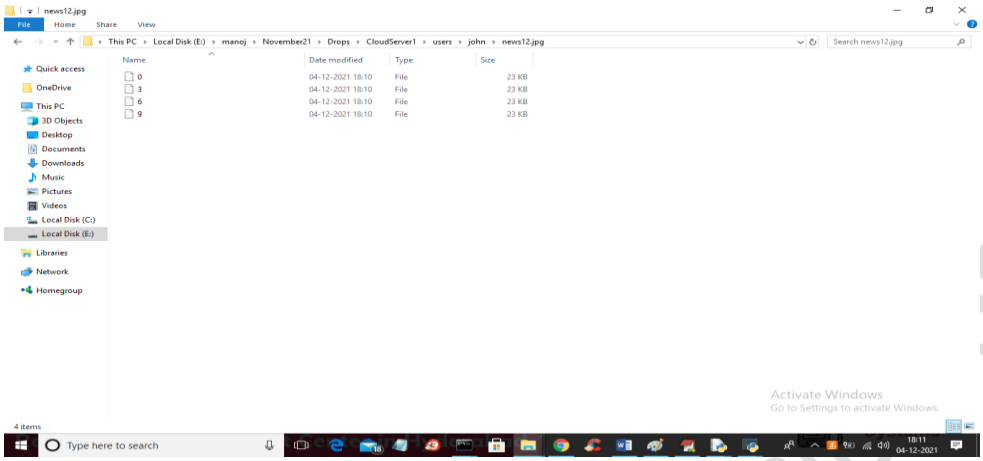


In above screen click on 'Choose File' option and then select any type of file to upload and in above I selected 'news12.jpg' file and then click on 'Open' button to upload file and to generate fragments

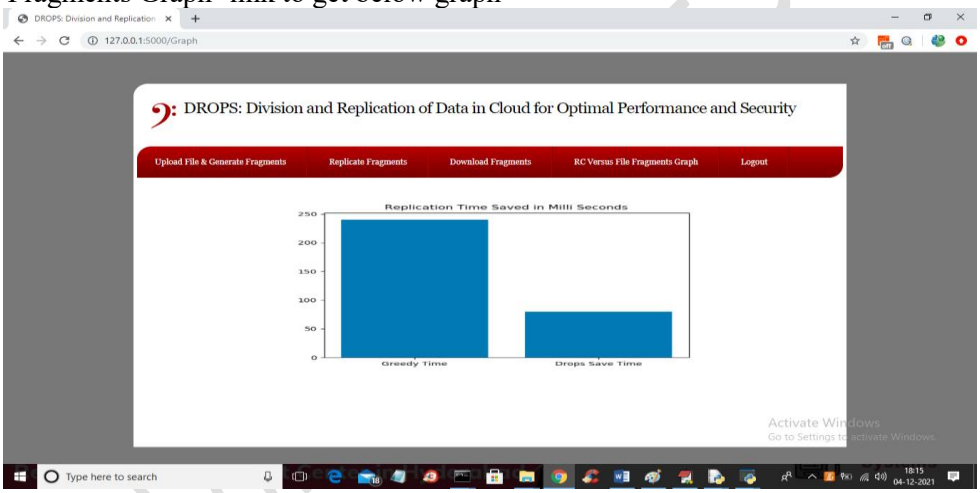

In above screen file uploaded and total 10 fragments generated and now click on 'Replicate Fragments' link to send all fragments to 3 cloud servers and to get below screen



In above screen we can see 10 fragments with file name, block name and cloud server name where this fragments is store. In below cloud server1 ‘users’ folder we can see this blocks



In above screen in E directory in last file we can see news12.jpg file downloaded and now click on ‘RC Versus File Fragments Graph’ link to get below graph




In above graph x-axis represents algorithm names and y-axis represents block saving time and drop is the propose work which took less time compare to existing Greedy Algorithm. Drop will save storage time by sending blocks/replications to multiple servers

## V. CONCLUSION

We proposed the DROPS methodology, a cloud storage security scheme that collectively deals with the security and performance in terms of retrieval time.The data file was fragmented and the fragments are dispersed over multiple nodes. The nodes were separated by means of T-coloring. The fragmentation and dispersal ensured that no significant information was obtainable by an adversary in case of a successful attack. No node in the cloud, stored more than a single fragment of the same file. The performance of the DROPS methodology was compared with full-scale replication techniques. The results of the simulations revealed that the simultaneous focus on the security and performance, resulted in increased security levelof data accompanied by a slight performance drop. Currently with the DROPS methodology, a user hasto download the file, update the contents, and uploadit again. It is strategic to develop an automatic update emechanism that can identify and update the required fragments only. The aforesaid future work will savethe time and resources utilized in downloading, updating and uploading the file again. Moreover, the implications of TCP in cast

over the DROPS methodology need to be studied that is relevant to distributed data storage and access.

## 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] 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.

[5] B. Grobauer, T.Walloschek, and E. Stocker, "Understandingcloud computing vulnerabilities," IEEE Security and Privacy, Vol.9, No. 2, 2011, pp. 50-57.

[6] W. K. Hale, "Frequency assignment: Theory and applications,"Proceedings of the IEEE, Vol. 68, No. 12, 1980, pp. 1497-1514.

[7] 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.

[8] M. Hogan, F. Liu, A.Sokol, and J. Tong, "NIST cloud computingstandards roadmap," NIST Special Publication, July 2011.

[9] W. A. Jansen, "Cloud hooks: Security and privacy issues incloud computing," In 44th Hawaii IEEE International ConferenceonSystem Sciences (HICSS), 2011, pp. 1-10.

[10] 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.

[11] 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.

[12] L. M. Kaufman, "Data security in the world of cloud computing,"IEEE Security and Privacy, Vol. 7, No. 4, 2009, pp. 61-64.

[13] 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.

[14] 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.

[15] 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 .