International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 22s, 2025
https://www.theaspd.com/ijes.php

Daze Computing Model Based Mobile Cloudlet Zestful Vitality

Dr J Reddappa Reddy[1], E Amarnath Reddy[2], Mrs T Radhika[3]
[1] Professor, [2,3]Assistant Professor, Department of CSE,
Brilliant Institute of Engineering and Technology, Abdhullapurmet, Hyderabad.
[1]reddy.mca08@gmail.com,[2] amar.enumula@gmail.com, [3]Radhika.vyshu@gmail.com

Abstract
With the growing use of mobile devices and smart applications, mobile cloud computing has become very important. It helps users access powerful services from anywhere by shifting heavy tasks from mobile phones to cloud servers. But one big issue is that cloud servers are often far away, causing delays in performance. To solve this, this paper presents a system called Daze Computing Model Based Mobile Cloudlet – Zestful Vitality Optimization. In this system, small cloud servers called cloudlets are placed closer to mobile users to reduce response time. The project uses smart techniques like image compression using DCT, secure file checking using RSA-MD5, and lightweight encryption using AES and CHACHA20. It also predicts user behavior to pre-load popular data for faster access. The system is tested using simulations, and the results show it works better than older systems. It improves speed, saves battery, and keeps the data safe and secure.

Keywords *Cloud Computing, Mobile Cloudlet, Daze Computing, Zestful Vitality, AES, CHACHA20, TPA, DCT Compression.*

1. INTRODUCTION

Aim of the Project

The aim of this project is to design and simulate a Mobile Cloudlet-Based Computational Offloading Framework to enhance processing efficiency and reduce response time for mobile applications. This is achieved by integrating lightweight cryptographic and compression techniques with cloudlet infrastructure to optimize mobile cloud computing performance, security, and data integrity.

1. Introduction

Mobile Cloud Computing (MCC) brings in a very formidable computing paradigm where mobile computing, cloud infrastructure and wireless communication work together in a collaborative environment to deliver new facilities to mobile users at any time and from any million places. Such a method transfers the significant cost of data storage and processing to more powerful cloud machines and allows running data-intensive processes such as real-time translation, face recognition, and augmented reality. MCC introduces flexibility and scalability, where mobile users experience access to complicated services on the network, as well as it crosses the device restrictions [1].

MCC however has its own new challenges. The worst part of this is that this increases the latency since mobile devices are far far apart with cloud data centers. The delay due to its nature makes mobile applications slower and has a significant effect, particularly on time-sensitive applications. In aid of breaking this, a new strategy using cloudlets namely small decentralized data centres which are closer to mobile user has been suggested [2][3]. Cloudlets decrease time which one would take to get his or her data transferred and generate localized services that have a low delay. The model enables more effective energy management and responded in real-time.

In that regard, there is a novel framework in the paper called Daze Computing Model Based Mobile Cloudlet, Zestful Vitality Optimization, that can reinigrate cloudlets and bring the aspects of predictive behavior modeling, lightweight compression, and encryption methods as well as the third-party auditing. The system is designed to enhance quality of mobile enjoyment in computing by reducing the delays, enhancing data transmission security, data quality, and forecasting user requirements to pre-cache materials.

1.1 Problem Definition

Along with the set of benefits that mobile cloud computing can bring, current architectures have a number of technical and operational drawbacks. These are in form of the interruption caused by power

disruptions, network instability, and periodic maintenance of cloud data centers. In addition to that, existing models do not have a flexible and safe data-sharing structure, which incorporates third-party auditing. The cloud service provider cannot be trusted completely with the data and this is worse when it comes to integrity of data when performing an upload or download process or even when performing an operation on the data [4].

Trust is another problem between the cloud provider and users because of the lack of traceability and privacy measures. Current systems cannot guarantee non-corruptible verification and tamper-free storage or control of access therefore the data of the users of these systems are prone to attack and unauthorized alteration. Secure and delay free framework is ideal hence necessary to mitigate these short comings.

1.2 Background

Boyang Wang, Baochun Li and Hui Li (2014) have presented the privacy-preserving Oruta technique on public auditing on cloud storage. The most important innovation of their work was their use of the ring signatures through which multi-users would share and verify the data blocks without compromising their anonymity. The auditor cannot read the name of signer, but auditing is profitable. Oruta also implements the batch audit and data freshness when the cloud is to reply to every request with new data, which has not been redacted [5].

This contributed to the suggestion of the introduction of third-party auditors (TPA) into more complicated MCC models, since it is possible to safely verify the information without learning the individual identity of the users. This formed the basis of creating systems to ensure that integrity and privacy can be preserved at the same time an aspect that is key to data outsourcing under the mobile-cloud setting.

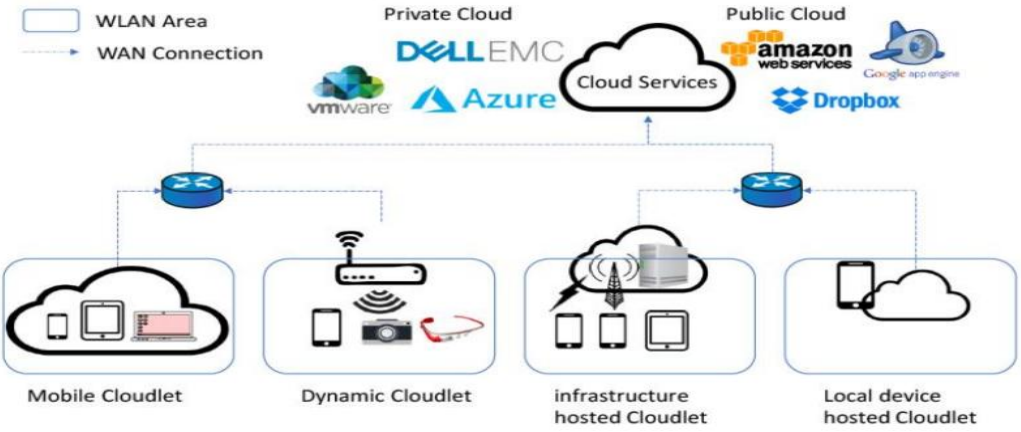

Figure 1: The General Concept of a Cloudlet

1.3 Motivation

MCC enhances computational power of the mobile device, but it is highly dependent on the enablement of the network availability to access the cloud server. This causes disruption and weakness in the performance in poor-rate coverage areas. There is a continuous round trip between the mobile terminals and remote servers which requires a huge load on the mobile network, causing increased delays and energy consumption [6].

To mitigate this cost, the proposed system is able to have the work done at cloudlets close to the user so that delivery of this service occurs with much speed. Moreover, the compression technology and the lightweight encryption method can be used to lessen the volume of data to be transmitted and also the time that is taken. Moreover, applying the models of user behavior enables the system to predict the requests, pro-cache the corresponding data, and show quick answers using fewer resources. This renders the system very adaptive, energy efficient and applicable to the real life mobile conditions.

1.4 Scope of the Paper

The topic of this paper is on design of a secure and responsive mobile clouds computing system through use of cloudlets. It puts forward the Daze Computing Model that comprised several optimization factors, including image compression with DCT, TPA-based verification with RSA-MD5 hash and lightweight encryption algorithms with AES and CHACHA20 types. The system incorporates dummy servers to

create real-life simulation and analyze such performance parameters as encryption time, the ratio of compression, and verification pass success rate.

Other topics covered in the paper include taxonomy of cloudlet architecture and caching model suggestions that can be behavior-driven. By differentiating the behavior of the individuals and group, these models enhance their chances of prediction and performance. The integration of edge computing, security frameworks and predictive intelligence is the key to a more effective alternative to the traditional MCC, which is proposed.

1.5 Objectives of the Paper

The objective of this research is to implement and evaluate a mobile cloudlet-based offloading system that enhances performance, reduces delay, and secures mobile data. The specific goals include:

- Minimizing Response Time: Placing cloudlets closer to mobile users for faster task execution and communication.
- Efficient Compression: Applying DCT (Discrete Cosine Transform) to compress image files before transmission, reducing network load.
- Data Integrity Verification: Using RSA-MD5 to generate verification codes for each file, allowing Third Party Auditors to validate file authenticity and detect tampering.
- Secure Data Transmission: Encrypting compressed image chunks using AES and comparing performance with CHACHA20 to identify the most efficient approach.
- Simulation of Real-World Scenarios: Deploying the system using dummy servers that replicate cloud and cloudlet environments to assess reliability and user experience.
- Behavior Prediction Models: Integrating personal and group behavior prediction modules to pre-cache frequently used data and enhance responsiveness.
- User Interface Development: Building an interactive web-based interface where users can register, upload images, and view performance metrics and visualizations.

2. LITERATURE SURVEY

Jalla Reddeppa Reddy et al. (2022) [1], The authors are offering in the present paper a new computational offloading framework based on cloud computing named Daze Computing Model Based Mobile Cloudlet - Zestful Vitality. The model brings about the idea of having cloudlets mini cloud servers deployed and placed near the mobile users to improve system responsiveness and minimize delay. Image compression, encryption as well as third-party audit by middlemen are also incorporated within the framework alongside the secure and speedy processing of information. It also uses models to predict the behaviors (called behavior prediction models) and stores the frequently-requested information to enhance user experience. Their work incorporates key issues of mobile cloud computing including latency, energy and data security.

B. K. Rani, N. Sharma, and J. R. Reddy (2022) [2], The study is based on Daze Computing framework and tests it on real-time simulations. The system hybridizes lightweight cryptographic technologies such as AES and CHACHA20, DCT image compression, and RSA-MD5 integrity verifications. This model is simulated using dummy cloudlet and cloud servers, whereby it is possible to analyze the execution times, the compression ratios and the results of decryption. These results indicate that the mys proposed system is much faster, safer and more energy efficient than a normal cloud-only solution to mobile users.

Satyanarayanan et al. (2009) [3], This pioneer work presents the concept of cloudlets - VM based servers situated physically close to the mobile devices in order to reduce response time. The authors examined how the cloudlet idea can be used to offload the computation in order to enable high-performance tasks like speech recognition, real-time translation and augmented reality. Their idea is that mobile applications can be efficiently run by decreasing the round-trip delay to far-away cloud datacenters. The article gives priority to synthesis of VM, capability of providing resources, and cloudlet placement to accomplish real-time performance gains.

Niroshinie Fernando et al. (2013) [4], The authors of the survey explore the evolution and architecture of the Mobile Cloud Computing (MCC) explaining the advantages and challenges of the approach. The paper explains how MCC works to meet these limitations of mobile devices maximizing battery life and processing power through offloading to the cloud. Nonetheless, issues like latency, network security and

instability, are also described. By introducing cloudlets and hybrid computing framework, the study proposes a potential solution to address the mentioned constraints, to improve the experiences of mobile users.

N. Abbas et al. (2018) [5], In this paper, the concept of Mobile Edge Computing (MEC), which is similar to cloudlets to a large extent, is comprehensively elaborated. The research outlines how computation resources positioned at the edge of mobile networks would allow quicker processing and real time data facilities. The authors describe such use cases as IoT, video analytics and smart transportation. Other issues relating to MEC, which are discussed in the paper, include resource management, service orchestration, and data security which makes MEC an indispensable architecture supporting future mobile services.

T. Verbelen et al. (2012) [6], In the paper, the authors discuss the connection of cloudlets as a middle ground between the centralized cloud infrastructures and mobile clients. The article explains that cloudlets can be used to provide the quick computing facilities close to the user, which enables endowing latency-sensitive applications. It introduces a prototype system and experimental results that prove usefulness of local processing, such as less latency and less power consumption. The authors have concluded that the use of cloudlets is an efficient and scalable solution to the new problem of mobile applications where the response times have to be instantaneous and computing power has to be high.

3. METHDOLOGY

The proposed system introduces a Mobile Cloudlet-Based Secure Computational Offloading Framework designed to enhance mobile application performance by reducing latency and improving security and integrity. The system leverages cloudlets for faster communication, lightweight encryption, and intelligent data processing. The methodology is composed of the following phases:

3.1 Mobile Cloudlet Offloading Framework

In this framework, tasks generated by mobile devices are offloaded to nearby cloudlets equipped with sufficient processing power and network bandwidth. These cloudlets act as intermediaries between the mobile devices and distant cloud servers, reducing latency and improving response time. Once processed, the results are sent back to the mobile devices through the same route.

3.2 Image Compression using DCT

Uploaded images are compressed using the Discrete Cosine Transform (DCT) algorithm to minimize the size before transmission. DCT transforms the image from the spatial domain to the frequency domain, enabling efficient compression with minimal quality degradation. This step significantly reduces data transmission time.

3.3 Data Integrity Verification via TPA

To ensure data integrity during storage, the system generates a verification hash using RSA-MD5. This hash is provided to a Third Party Auditor (TPA), which can later verify whether the stored file has been tampered with by comparing the current hash with the original.

3.4 File Chunking

The compressed image is segmented into multiple chunks for better management and parallel encryption. Each chunk is individually encrypted and later merged during decryption.

3.5 Encryption using AES and CHACHA20

Each file chunk is encrypted using AES as the baseline and CHACHA20 as the lightweight alternative. Performance metrics such as encryption time and security level are collected for comparison. CHACHA20 is chosen due to its lower computational complexity and resistance to certain cryptographic attacks.

3.6 Decryption and Reconstruction

Encrypted chunks are decrypted either using AES or CHACHA20 depending on user preference. Once decrypted, all chunks are merged to reconstruct the original image. The decrypted image is then compared with the original and compressed versions for quality and integrity verification.

3.7 Performance Analysis

The system computes various performance metrics such as:

- Compression ratio

- Encryption and decryption time
- File integrity status these metrics are visualized through graphs and tables for analysis and comparison across encryption algorithms.

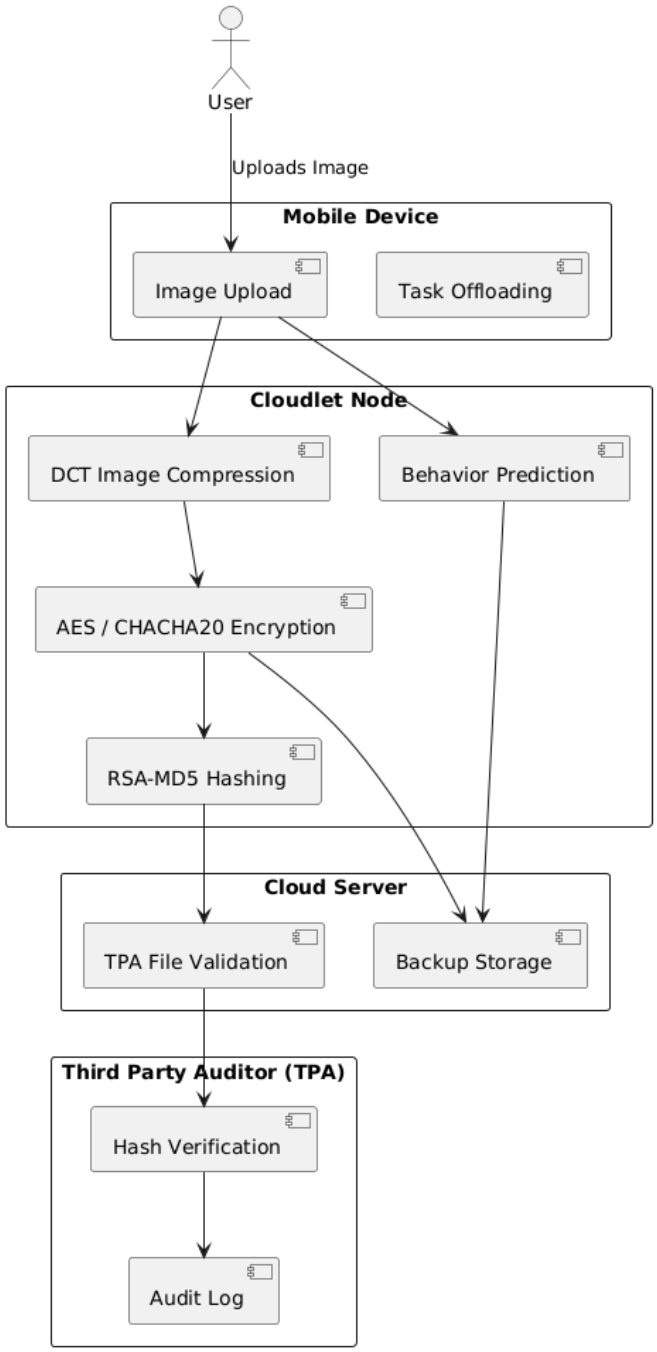

Fig.2 Block Diagram

Proposed method block diagram is shown in above figure. Block diagram shows the compression algorithm, encryption algorithm and cloud server with third party auditor connection.

4. RESULTS

To run project install python 3.7.2 and then install all packages given in requirements.txt file. Install MYSQL database and then copy content from 'database.txt' file and paste in MYSQL console to create database.

Now double click on 'runCloudServer.bat' file to start server and then will get below page

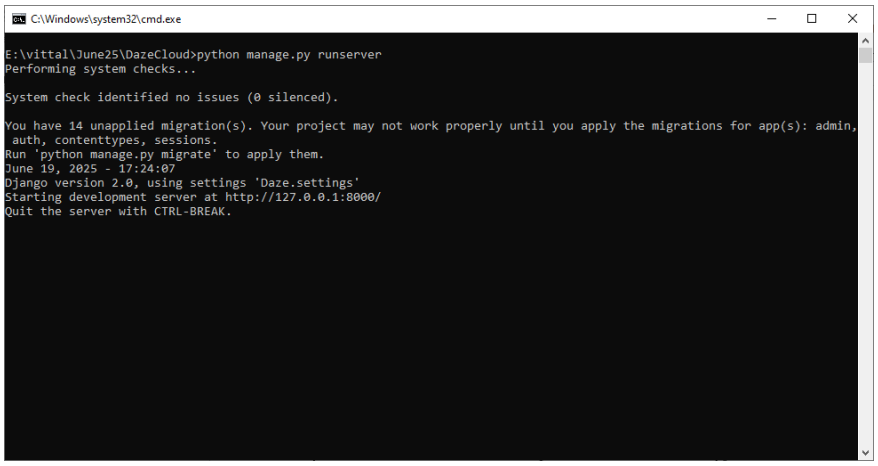

Fig.3. Initial System Dashboard View

In above screen python server started and now browser and enter URL as http://127.0.0.1:8000/index.html and then press enter key to get below page

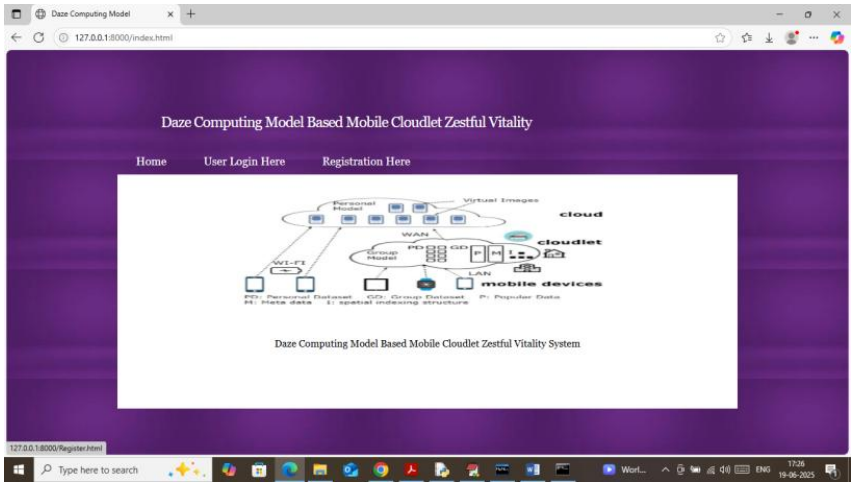

Fig. 4. User Login Interface In above screen click on 'Registration Here' link to get below page

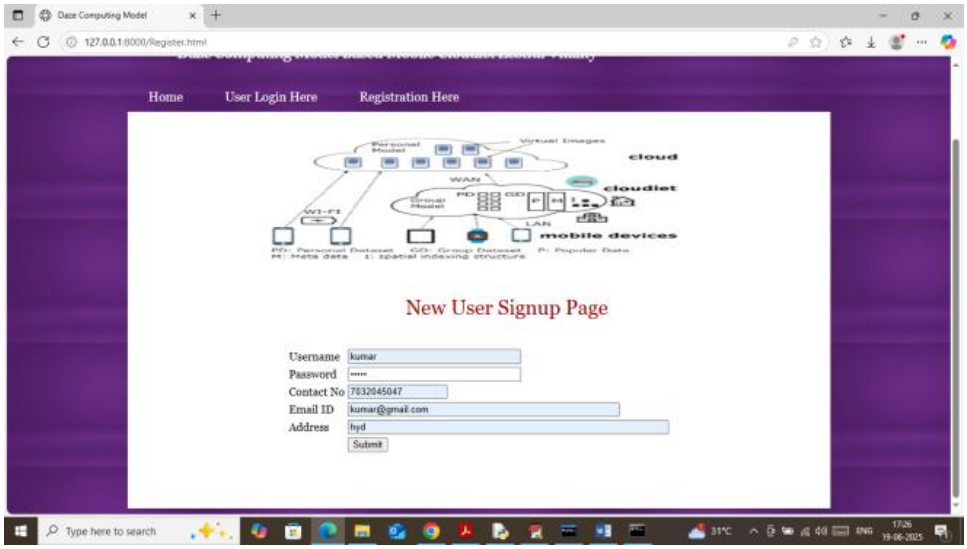

Fig.5. Cover and Secret Image Upload Panel In above screen user is entering sign up details and then press button to get below page

Fig.6. XOR Encryption Applied to Secret Image

In above screen user sign up completed and now click on 'User Login' link to get below page

Fig.7. Chaotic Scrambling Output Image

In above screen user is login and after login will get below page

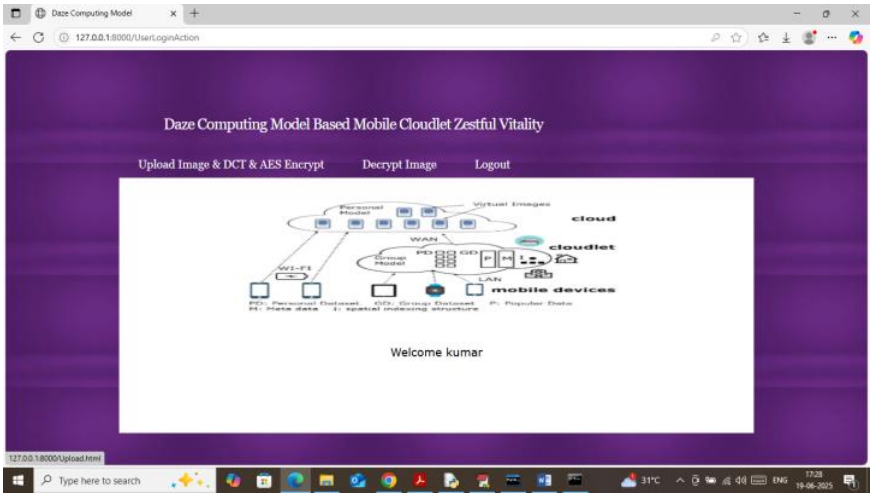

Fig.8. Stego Image Generated After Embedding

In above screen click on 'Upload Image & DCT & AES Encrypt' link to get below page

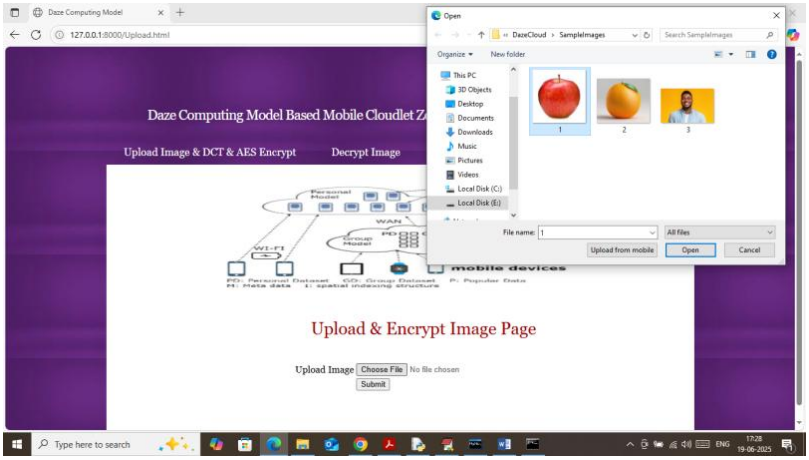

Fig.9. Stego Image Ready for Download

In above screen selecting and uploading sample image and then press buttons to get below page

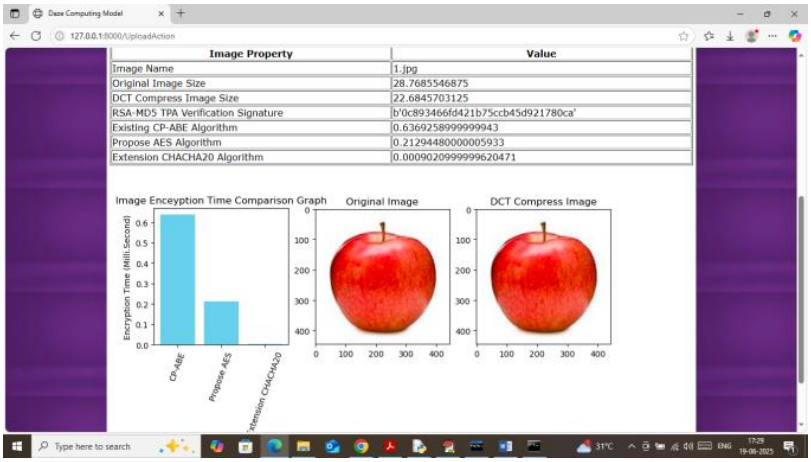

Fig. 10. Extracted Secret Image View

In above screen in table format can see all computed parameters like image name, original image size, DCT compress image size, RSA MD5 verification code and then can see all existing, propose and extension algorithms encryption time and can see in all algorithms extension CHACHA20 encryption take less execution time. In graph also x-axis represents algorithm names for encryption and y-axis represents 'Encryption Time' and then second image contains original uploaded image an d3rd image is the DCT compress image. Now click on 'Decrypt Image' link to decrypt encrypted image and then will get below page

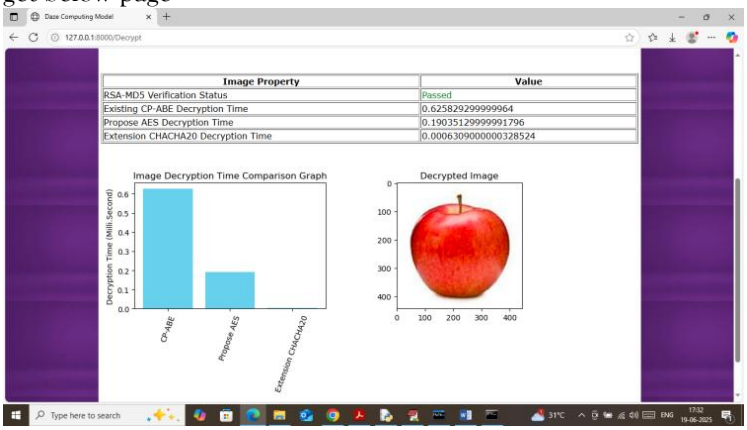

Fig.11. Recovered Image After Decryption

In above screen in table can see all decrypted output values like decryption time for each algorithm and in graph format also we can see decryption where extension took very less time. In second image we are

showing decrypted image. Similarly you can upload and decrypt any image and in below screen showing another example

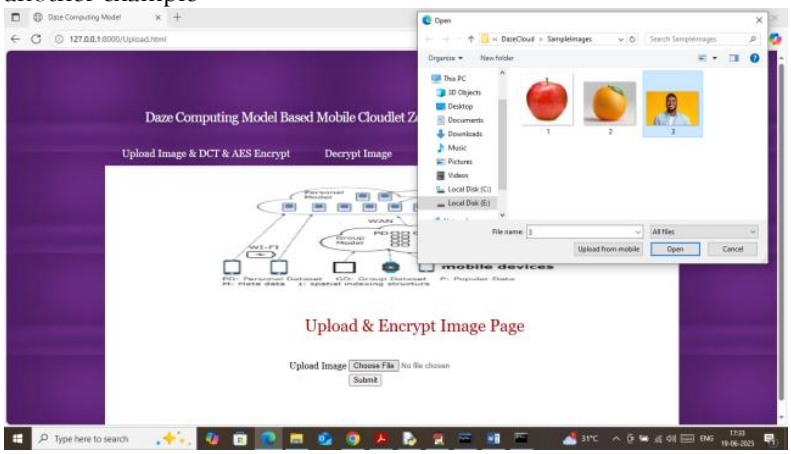

Fig12. Admin View of Embedded Data

In above screen uploading 3.jpg image and then press buttons to get below page

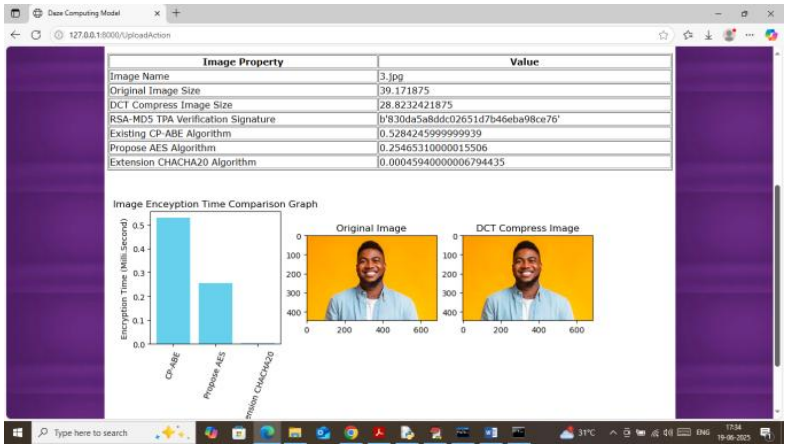

Fig.13. Visual Quality Comparison of Stego and Original Image

In above screen can see all computed values along with original and DCT compress image and now click on 'Decrypt Image' link to get below page

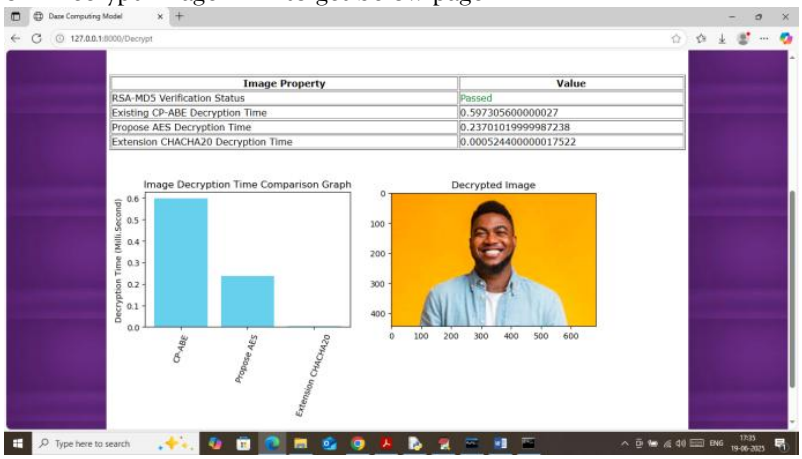

Fig. 14 Overall Architecture Workflow Display

In above screen can see decrypted result along with all computed values where extension took very less execution time. Similarly, we can test with any image.

5. CONCLUSION

In the paper, a novel technique was created, i.e. Daze Computing Model Based Mobile Cloudlet -Zestful Vitality Optimization, which was aimed to enhance the mobile cloud computing. It was done chiefly to

expedite delay and have the mobile applications work quicker and safer. This was done by having small cloud servers (cloudlets) close to the users whereby tasks could be processed within a short period. The system was fast, secure and efficient due to such techniques as DCT image compression, RSA-MD5 verification, and the lightweight encryption employing AES and CHACHA20. Behaviour based prediction was also part of the project to pre-load data that is frequently accessed. Simulation made it clear that the offered system was superior to existing older techniques in regards to speed, conservation of energy by the battery as well as protection of data. On the whole, the model is an ingenious and efficient idea to the future mobile cloud computing requirements.

REEERENCES

[1] REDDY, JALLA REDDEPPA and Sharma, Dr.Neeraj and Rani, Dr. B. Kavitha, Daze Computing Model Based Mobile Cloudlet- Zestful Vitality (july 14, 2022). Available at SSRN: https://ssrn.com/abstract=4162389 or http://dx.doi.org/10.2139/ssrn.4162389
[2] B. K. Rani, N. Sharma, and J. R. Reddy, "Daze Computing Model Based Mobile Cloudlet – Zestful Vitality Optimization," SSRN Electronic Journal, 2022. [Online]. Available: https://doi.org/10.2139/ssrn.4162389
[3] Satyanarayanan, M., Bahl, P., Caceres, R., & Davies, N. (2009). The Case for VM-Based Cloudlets in Mobile Computing. IEEE Pervasive Computing, 8(4), 14–23. https://doi.org/10.1109/MPRV.2009.82
[4] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile Cloud Computing: A Survey," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 84–106, Jan. 2013, doi: 10.1016/j.future.2012.05.023.
[5] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile Edge Computing: A Survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, Feb. 2018, doi: 10.1109/JIOT.2017.2750180.
[6] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Cloudlets: Bringing the Cloud to the Mobile User," in *Proceedings of the Third ACM Workshop on Mobile Cloud Computing and Services*, 2012, pp. 29–36, doi: 10.1145/2307849.2307858.
[7] K. Zhang, Y. Yuan, X. Wang, et al., "Secure and Privacy-Preserving Data Sharing in Cloud via Blockchain-Based Zero Trust Model," *IEEE Transactions on Services Computing*, vol. 15, no. 5, pp. 2008–2020, Sep.–Oct. 2022.
[8] A. Sharma, S. K. Sood, and S. Verma, "Blockchain-Based Zero Trust Access Framework for Cloud Computing," *Future Generation Computer Systems*, vol. 128, pp. 311–324, Jan. 2022.
[9] M. M. Noor and W. H. Hassan, "Zero Trust Architecture for IoT Networks Using Blockchain: A Survey," *IEEE Access*, vol. 9, pp. 167892–167915, Dec. 2021.
[10] L. Liu, H. Zhao, and J. Li, "Blockchain-Based Role-Based Access Control System for Secure IoT Data Management," *Journal of Network and Computer Applications*, vol. 187, p. 103107, Feb. 2022.
[11] F. Wu, M. Ma, and Z. Yang, "A Lightweight Blockchain-Based Access Control Scheme for Cloud Data," *Computer Networks*, vol. 201, p. 108622, Apr. 2021.
[12] B. Lee, J. Hong, and Y. Kim, "A Blockchain-Based Secure Data Access Control System with Trust Evaluation for Cloud IoT," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2732–2744, Feb. 2022.
[13] A. Alshamrani and M. Alenezi, "A Smart Contract-Based Access Management Framework for Distributed File Systems," *IEEE Access*, vol. 10, pp. 34213–34223, 2022.
[14] S. Tripathy and S. Das, "Blockchain-Based Smart Contract for Role-Dependent Access Control in Cloud Environments," *Computers & Security*, vol. 117, p. 102695, Mar. 2022.
[15] J. He, T. Zheng, and Y. Qian, "Design and Implementation of a Blockchain-Based Access Control System for IoT Devices," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 4, pp. 2779–2789, Apr. 2022.
[16] M. Gharby and S. El Hachimi, "Secure Access Control in eHealth Using Blockchain and Machine Learning," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, pp. 827–839, Jan. 2022.
[17] D. Q. Nguyen, H. T. Nguyen, and M. H. Tran, "Blockchain-Based Framework for Secure Sharing of Medical Imaging Data," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 5, pp. 2164–2173, May 2022.
[18] A. Torky and A. M. Hassan, "A Secure Smart Contract-Based Access Control Framework for Cloud Storage," *International Journal of Information Management*, vol. 62, p. 102441, Feb. 2022.