

Android Offloading Computation over Cloud

Ayushi Srivastava,

Department of Computer Science & Engineering,
Chandigarh University, Mohali

K Divya,

Department of Computer Science & Engineering,
Chandigarh University, Mohali

Kavita,

Uttaranchal University, Dehradun Email: kavita@ieee.org

Anil Kumar Dixit

Uttaranchal University, Dehradun Email: anildixit@uumail.in

Abstract—Cloud is a field which comes up very fast in past years. Along with cloud, androids smartphone demand in market also booms. As Mobile cloud computing is a combination of Android and Cloud Computing so its demand in IT industries is very high. Mobile devices have experienced a lot of enhancement and evolution, along with that usage of smartphones has doubled to that of PC's but still they are assumed as limited computing devices. Nowadays, generation is more demanding and performance intensive applications on their smartphones. In this research paper I will be offloading the data of smartphones to the cloud by using Amazon's cloud portal i.e.; AWS and will also present a framework through which data will offload from devices.

Index Terms—Cloud computing, Offloading data, Frameworks, AWS portal, Android

I.

INTRODUCTION

1.1 Cloud defines as servers accessing the Internet as well as websites and software running on those servers. There are several data-centers all over the world where cloud servers are present. Cloud computing helps users and companies in such a way that they don't have to manage software applications on their own machine and also not to handle physical servers by themselves [21]. Cloud also allows users to access the files and applications anywhere from any device because all the computing and storage work takes place in data centers on servers, instead of locally on the user's device [1].

A. Cloud Computing

is defined as delivering services that are hosted by an individual or an organization over the network.

B. Architecture of Cloud

is a term in which components of technologies are combined to build cloud, where resources are combined through a virtual portal i.e.; cloud portals like AWS, GCP, Microsoft Azure, etc and these portal share resources across the network [22,23]. The architecture is shown in Fig 1.

To reduce reliance on storage, networking infrastructure and on-premise server; architecture of cloud helps organization a lot in every manner. As shown in fig 2.

Cloud architecture has three major models:-

SaaS is a Software as a Service; PaaS is a Platform as a Service; IaaS is a Infrastructure as a Service.

C. Computational Offloading

To reduce battery power consumption and improve performance, Computational Offloading is used as it sends time-consuming and heavy computational activities to nearby servers which are present in data centers. If there is a computing-powered-restricted environment

then Computation Offloading helps in providing content. Since there are still so many areas of improvement in this field like present approaches does not estimate execution costs accurately so many other techniques are also introduced which I will discuss later in paper.

D. Offloading frameworks

are type of application which are present in devices through which compute intensive & non-interactive parts of application are offloaded easily to the cloud by user [24].

As I will be using the AWS portal in this project so here I will be describing The AWS Well-Architecture Framework which shows key concepts, design principles, and best architectural practice for running workloads and designing in the cloud.

AWS Well-Architecture Framework is made up of six pillars i.e.; Operational Excellence, Security, Reliability, Performance Efficiency, Cost Optimization and Sustainability.

II. LITERATURE REVIEW

As I have written about cloud and its architecture in above introduction so here I will be telling about what other authors said about the need of cloud in real life and also concept of offloading which I have read from some recent research papers, journals and articles. There are so many offloading techniques in this fast moving world but there are also some cloud portals which helps a lot to use resources of cloud. Elgendy, Ibrahim A., et al. [11] described the closest existing method to the vision is cyber foraging, which aims to offload heavy tasks of a mobile application to the surrogate via WiFi.

Khoda, Mahbub E., et al. [12] described offloading tasks to second server if first server is busy and save copy from the file after processing for the first time to save battery and for less delay.

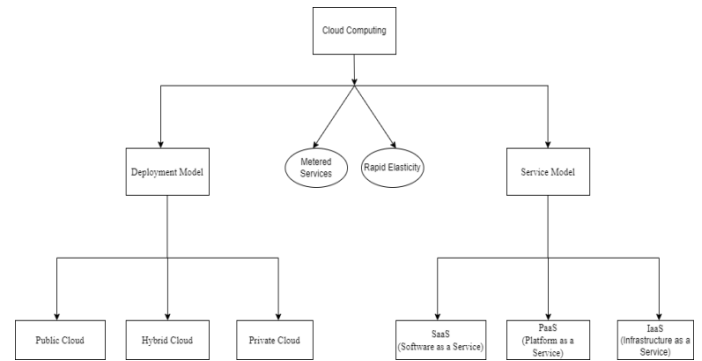


Fig. 1. Cloud Computing structure

Wu, Song., et al. [13] researched that proposed offloading framework methods can effectively adapt to the dynamic cloud computing and networking environment which will ease the working load of organizations.

B.B Gupta., et al. [14] describes how to develop portal that authenticate old users and then create new one where each one will get separate storage space in AWS S3. Through this all users will get there containers to offload there smart-devices [25]. Khaled Boussetta., et al. [15] disclosed a method of real testbed which composed of client terminal and offloading server located remotely.

Song Wu, Chao Mei., et al. [16] gives the concept in which aim is to handle several application in one unikernel and along with it avoids time-consuming recompilation. This whole concept name is Rich-Unikernel which is light-weight and also has flexible runtime during offloaded codes called Android Unikernel. This whole process integrates libraries of basis android system into OSv unikernel.

Joaquim Silva, Eduardo., et al. [17] introduced Edge computing which brings up requirement of applications in mobile in terms of communication and computation and along with this emerging IOT with millions of devices.

Robin. Manmohan Sharma., et al. [18] has given some techniques which are based on decision engine components and profiling.

Ming LI, Yun Ma., et al. [19] designed a framework that uses an optimization model for dynamically determining the decision of offloading based on some of parameters called, storage usage, time of execution, consumption of energy and at last utilization of CPU.

Hina Akram, Sukhpal Singh Gill., et al. [20] proposed a symmetric approach named EFFORT for offload communication in the cloud. In this approach energy consumption issue during communication in intense application of mobile devices has be raised and solved.

III. METHODOLOGY

A. Offloading Approaches:

In this section I will be mentioning some of the approaches which are used by offloading techniques.

B. Offloading Steps:

Offloading performs transferring of compute intensive task to remote server. It is performed by three basic steps:

1. Application Partitioning - In this partitioning, applications are divided into two components from which one will stay on the mobile device and one will migrate to the server of cloud [2,26].
2. Preparation - The preparatory step performs all the necessary actions to extract the downloadable components to use it in smartphones. These consist of remote server selection, transfer and coding, as a starting point for processes which are proxy which detect and perform tasks on the place of SMD [3]. Without code sending, data also may be send to remotely control the remote.
3. Offloading Decision - The decision to remove is the final step before the remote application can be started in non- loading sections. Whether the installed part of the remote control gets used in the SMD system or did not depends on the operating environment. If the conclusion is made during operation, the available information will be more accurate, for

example, may be SMD does not have a wireless connection or the use of remote data transmission power may be very excessive. Uploads can be changed if situation changes [4,27]. Such a time-consuming decision attracts the most common shortcomings in the design process.

C. Framework Classes:

There are two types of offloading frameworks for offloading computation on a remote server [5,28].

1. Static Offloading Frameworks - In this, at design time, all presented steps are performed, before the application is started on smartphones.
2. Dynamic Offloading Frameworks - In this final decision to offload a computation is taken at runtime.

Framework Mechanism: There are two major categories which define these mechanisms

1. Frameworks based on VM (Virtual Machine) cloning - The full image of mobile device is captured and stored on the cloud server.
2. Framework based on code offloading - It removes parts of the in-app application by using binary conversions & annotations.

D. Existing Techniques:

It seems very hard to recognize the right technique which can be used for our work so as I mentioned above that there are many other techniques to offload data to cloud, so in this section I will be discussing some offloading techniques.

- 1) MAUI :: A structure that combines the benefits of these

two approaches: increases energy efficiency by generating well-refined code while minimizing the necessary changes to operating systems [6,29]. As MAUI makes its decision during operation it is a type of real-time framework. MAUI achieves these benefits by using a few features of today's managed code environment.

- 2) Mirror Server : : This framework uses TSP based re-

remote services. Mirror server extends smartphones capabilities by providing services like computation offloading, storage and security. For different mobile device platform, mirror server retains virtual machines templates. In this framework there is no need of partitioning because it offloads the whole application. For offloading, this framework employs an optimized mechanism.

3) Cuckoo : : This framework use Java stub model to

offload smartphone applications onto a remote cloud server. Its aim is to reduce battery usage and enhance performance of device [7,30]. In this framework Eclipse development tool and open source Android framework are integrated. Cuckoo works on dynamic offloading framework.

Apart from all mentioned techniques above I will be introducing the simple method of offloading data to the cloud directly by an android application that will offload all selected device data to cloud. For this I will be making a framework on android device which will help in offloading data to the S3 resource of AWS portal [8,31]. This framework will be made by using Visual Studio and from there only I will be connecting the AWS portal to my framework.

On the web and in mobile applications, it often gives users the function for data uploading. Applications may also give access to users for uploading documents or media. All modern web server technologies have ways to enable this feature. The user interface screens are shown in Fig 4, 5, 6, 7 and 8

Server-based environment, follows processes mentioned below in a form of flow-chart:

- 1) In the Application server, files are uploaded by the users
- 2) For the further processing, application server saves file to temporary storage
- 3) 3. Now files get transferred to the places like file server, database and object store by application.

Although the process has its proper valid side effects on function of web-serve in busy applications. Maximum files are large, so

uploading them shows a sensitive portion of the network inputs/outputs and server CPU time. Tracking status is also important to be sure about whole files are uploaded successfully and retries & errors are settled.

This challenge of traffic patterns will be very hard for applications to handle like if a special move of a web application is to send marriage invitation will face lot of traffic during season. It required good scalability and network bandwidth available on that season for good and smooth working.

With the help of application server, we can stop proxy of requests during uploading files directly to Amazon S3. This will tends to reduce of network traffic and server CPU usage during heavy. The S3 is highly accessible and long-lasting, [9] so we can make it suitable for store file permanently and user uploads.

4) Process of Uploading to Server-less S3 bucket: Requesting a signed URL from AWS S3 service is important for directly uploading to S3 bucket. From this signed URL it will be easy to upload directly. [10] For front-end of application these two-steps will be processed as shown in Fig 3.

5) Testing the application: Let's see two ways of application testing.

First with Postman, which calls API directly and with the help of signed URL upload binary file [20].

Second with a frontend application that integrates the API.

- To test using Postman:

- 1) From the API endpoint copy expansion output
- 2) Paste the API endpoint in Postman interface where labeled box exist. Then enter Request URL.
- 3) Select Send
- 4) When request is completed, the JSON response will be shown on body. The upload URL attribute hold the signed URL.

- 5) Copy feature to clipboard.
 - 6) Tap + icon next to tab for generating new request
 - 7) Select Dropdown, from there switch the method PUT instead of GET. Enter the URL and paste it to the Request URL box.
 - 8) Select the tab named 'BODY', then enter the Binary Radio button
 - 9) Now Select file and go for a JPG file you wanted to upload.
 - 10) Choose Send. You see 200 OK response after the file is uploaded.
 - 11) Navigate to S3 bucket from S3 console, created during deployment. There the JPG file we uploaded using Postman will be appear.
- 6) To test with a sample frontend application::
From example repo, copy index.html to the S3 bucket.

- To make objects public for read update permissions.

- In browser, navigate to the index.html file public URL. Now to upload in file picker, tap 'Choose file' then browse

JPG file. A pop up message will display of confirmation after

we will choose image to upload.

The second image we uploaded by browsing will reflect in S3 bucket which was deployed in S3 console.

IV. CHALLENGES IN COMPUTATION

A. OFFLOADING

There are several perspectives to see issues but here I will be discussing some of the major issues which are faced during offloading.

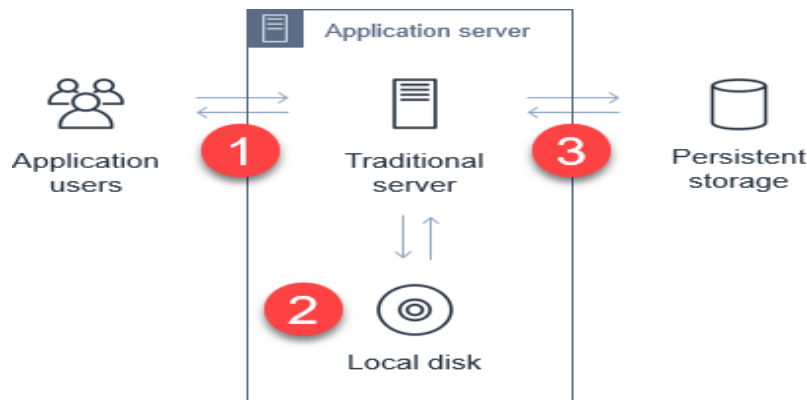
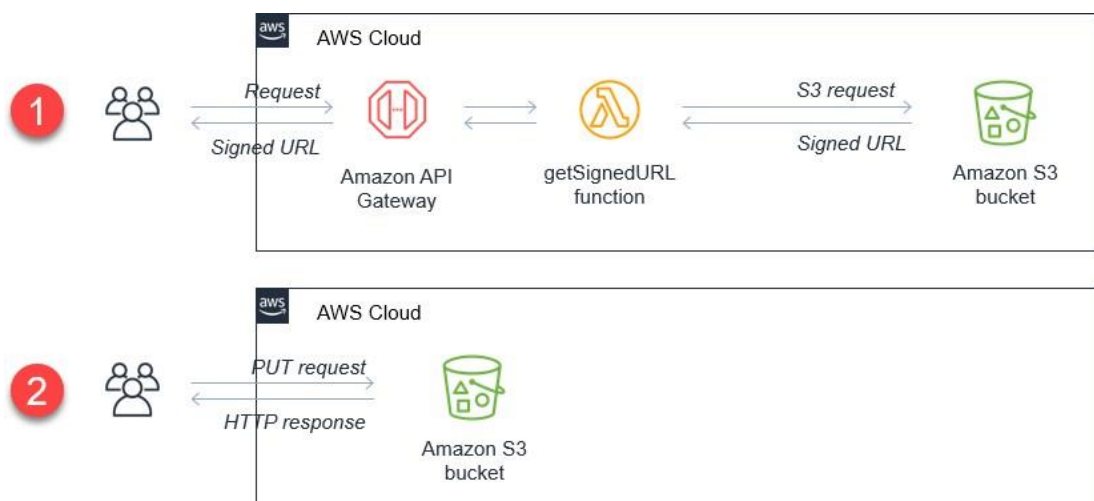


Fig. 2. Server based Environment



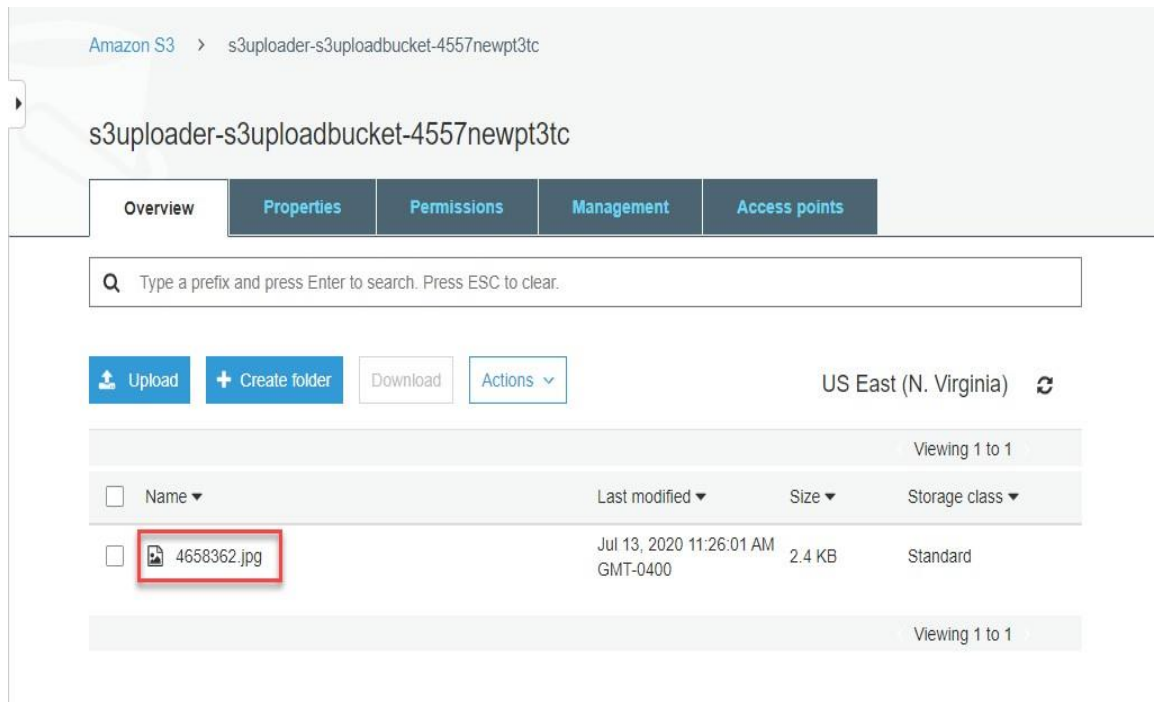


Fig. 7. Deployment phase

B.

Platform variation

One of the crucial challenges of offloading in current is the diversity and diversity of mobile phones designs and their OS. This variation comes from the following example: MAUI [11] is a download framework running on the Net Framework while Mirror Server [12] is a compiler of Android platform. Regular access to services of cloud is believed that SMDs will be able to operate services of cloud regardless of operating system. The standard loading of various smartphone platforms remains a challenge for the MCC sector.

Security and Privacy in smartphone's applications of cloud Security of Data transmission seems as high concerned aspect of application processing based on cloud. Security and privacy are two major aspects of the offloading process. There can be different angles to address these concepts like mobile devices, cloud data centers and during transmission of data across networks. Therefore, both mobiles are expected to have a high level of security.

C. Automatic system

Available frameworks for compute offloading need to be automated which will help in smoothing the process of offloading while exploring the surroundings. Achieving this type of automation is not easy process as it requires the implementation of some important protocols for identifying and identifying services based on the current context and its limitations [15].

D. Cost of offloading

The use of cloud infrastructure resources imposes financial costs on end users, which must be paid in accordance with the service level agreement (SLA) agreed with the cloud seller providing the service. Typically, managing content offloading and data transfer between cloud providers incurs additional costs on end users. [16] Therefore, consideration of financial factors is important when taking offloading decisions.

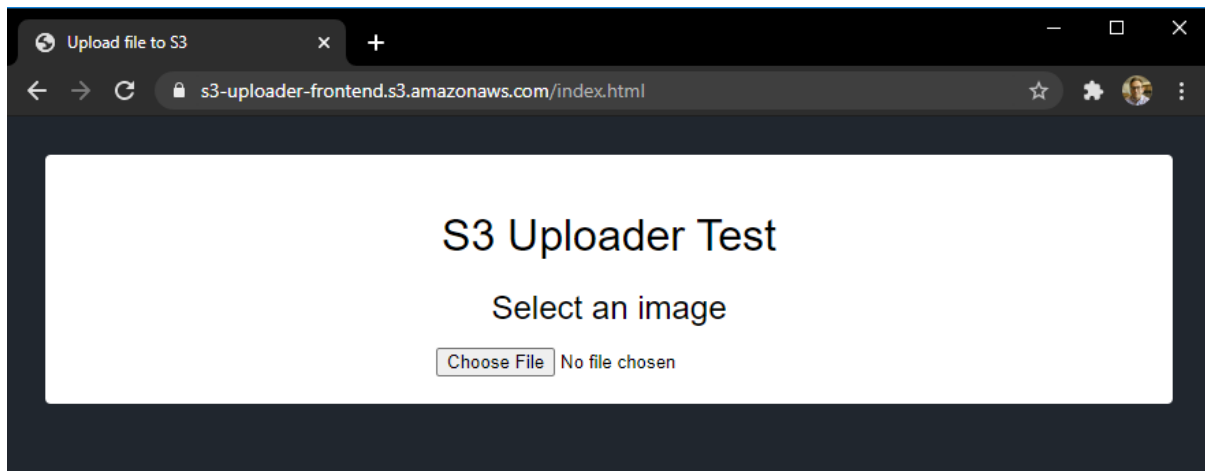


Fig. 8. To test the file, S3 Uploader

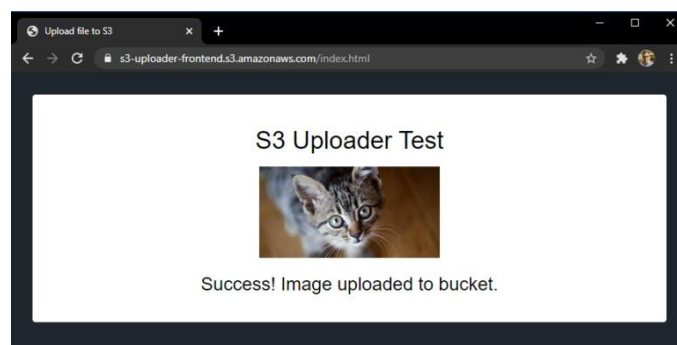


Fig. 9. Image uploaded successfully

E.

Partition offloading and external data input

During runtime as shown in fig 8 and fig 9, deciding which application components to offload and finding the appropriate server is a challenge [17]. Algorithms that answer this problem require intensive effort regarding resources, which gives problem to the execution time of the off-loaded partition of application. Although present application segmentation algorithm allows optimal application of applications between smart devices and servers of cloud, they don't provide solutions on how to use and utilize re- sources which are elastic in the cloud. Scalable applications are especially needed when they need to serve a large number of mobile users and when any application needs to input data stored on other remote servers [18].

V. CONCLUSION

In this paper I have discussed about Cloud Computing, Mo- bile cloud computing,

Computational Offloading: Techniques and their approaches, also the method of offloading data using

AWS portal. In approaches paper have information of dynamic and static approach.

Further in paper I have also shown the issues and chal- lenges which are faced during offloading in which security is the important challenge which programmer face. Along with Security, Cost is one of the basic issues which users face.

Basically I have shown the existing framework and tech- niques which are used for offloading data to the cloud.

REFERENCES

[1] T. Puri, R. K. Challa, and N. K. Sehgal, "Energy-efficient delay-aware preemptive variable-length time slot allocation scheme for WBASN (ed- pvt)," Proceedings of 2nd

International Conference on Communication, Computing and Networking, pp. 183–194, 2018.

[2] N. Kaur, "Thermal aware routing protocols in WBAN," 2021 4th International Conference on Signal Processing and Information Security (ICSPIS), 2021.

[3] T. Kaur, N. Kaur, and G. Sidhu, "Optimized energy efficient and QoS aware routing protocol for WBAN," Recent Patents on Engineering, vol. 14, no. 3, pp. 286–293, 2021.

[4] N. Chauhan and P. Tekta, "Fraud detection and verification system for online transactions: A brief overview," International Journal of Electronic Banking, vol. 2, no. 4, pp. 267–267, 2020.

[5] N. Chauhan, N. Kaur, and K. S. Saini, "Energy Efficient Resource Allocation in cloud data center: A comparative analysis," 2022 International Conference on Computational Modelling, Simulation and Optimization (ICCMO), 2022.

[6] P. Rani, S. Verma, N. Kaur, M. Wozniak, J. Shafi, and M. F. Ijaz, "Year). Robust and secure data transmission using artificial intelligence techniques in ad-hoc networks," Sensors, no. 1, pp. 22–22.

[7] Sood, M., Verma, S., Panchal, V.K.: Optimal path planning using hybrid bat algorithm and cuckoo search. Int. J. Eng. Technol. 7(4.12), 30–33 (2018).

[8] Kumar, P.; Verma, S. Detection of wormhole attack in VANET. Natl. J. Syst. Inf. Technol. 2017, 10, 71.

[9] Batra, Isha, Sahil Verma, Arun Malik, Kavita, Uttam Ghosh, Joel J. P. C. Rodrigues, Gia Nhu Nguyen, A. S. M. Sanwar Hosen, and Vinayagam Mariappan. 2020. "Hybrid Logical Security Framework for Privacy Preservation in the Green Internet of Things" Sustainability 12, no. 14: 5542.

[10] Gaba S, Verma S. Analysis on Fog Computing Enabled Vehicular Ad hoc Networks. Journal of Computational and Theoretical Nanoscience, 2019, 16(10), pp. 4356–4361.

[11] N. Kaur and S. Verma, "Detection of plant leaf diseases by applying image processing schemes," Journal of computational and theoretical nanoscience (JCTN), vol. 16, no. 9, pp. 3728–3734, 2019.

[12] M. Shafiq, H. Ashraf, A. Ullah, M. Masud, M. Azeem, N. Jhanjhi, and

M. Humayun, "Robust cluster-based routing protocol for IoT-assisted smart devices in WSN," Computers, Materials & Continua, vol. 67, no. 3, pp. 3505–3521, 2021.

[13] M. Lim, A. Abdullah, and N. Z. Jhanjhi, "Performance optimization of criminal network hidden link prediction model with deep reinforcement learning," Journal of King Saud University-Computer and Information Sciences, vol. 33, no. 10, pp. 1202–1210, 2021.

[14] V. E. Adeyemo, A. Abdullah, N. Z. Jhanjhi, M. Supramaniam, and A. O. Balogun, "Ensemble and deep-learning methods for two-class and multi- attack anomaly intrusion detection: an empirical study," International Journal of Advanced Computer Science and Applications, vol. 10, no. 9, 2019.

[15] Babbar, H.; Rani, S.; Masud, M.; Verma, S.; Anand, D.; Jhanjhi, N. Load balancing algorithm for migrating switches in software-defined vehicular networks. Comput. Mater. Contin. 2021.

[16] Dogra, V.; Verma, S.; Kavita; Chatterjee, P.; Shafi, J.; Choi, J.; Ijaz, M.F. A Complete Process of Text Classification System Using State-of-the-Art NLP Models. Comput. Intell. Neurosci. 2022, 2022, 1883698.

[17] Ghosh, Gopal, Kavita, Divya Anand, Sahil Verma, Danda B. Rawat, Jana Shafi, Zbigniew Marszałek, and Marcin Woźniak. 2021. "Secure Surveillance Systems Using Partial-Regeneration-Based Non-Dominated Optimization and 5D-Chaotic Map" Symmetry 13, no. 8: 1447. <https://doi.org/10.3390/sym13081447Y>.

[18] Miao, H. Jia, X. Liu, Y. Zhang, and W. Tan, "The Research and Application of Cloud Printing Platform Based on Improved AES-RSA Encryption Algorithm," 2019 Scientific

Conference on Network, Power Systems and Computing, 2019.

[19] M. L. Neelima, "Cloud Printing: An Innovative Technology Using Mobile Phone," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 5, pp. 966–971, 2014.

[20] Kumar, M., Raju, K.S., Kumar, D. et al. An efficient framework using visual recognition for IoT based smart city surveillance. *Multimed Tools Appl* 80, 31277–31295 (2021). <https://doi.org/10.1007/s11042-020-10471-x>

[21] S, K, S. R, F, J. F, & M, and A, "A Study on Three Step Multifactor Authentication System for Modern Security," *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, no. 3, pp. 45–49, 2022.

[22] S. H. Kok, A. Abdullah, and N. Z. Jhanjhi, "Early detection of crypto- ransomware using pre-encryption detection algorithm," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 5, pp. 1984–1999, 2022.

[23] Sennan, S., Somula, R., Luhach, A. K., Deverajan, G. G., Alnumay, W., Jhanjhi, N. Z., ... & Sharma, P. (2021). Energy efficient optimal parent selection based routing protocol for Internet of Things using firefly optimization algorithm. *Transactions on Emerging Telecommunications Technologies*, 32(8), e4171.

[24] A. Almusaylim, Z., Jhanjhi, N. Z., & Alhumam, A. (2020). Detection and mitigation of RPL rank and version number attacks in the internet of things: SRPL-RP. *Sensors*, 20(21), 5997..

[25] Khalil, M. I., Jhanjhi, N. Z., Humayun, M., Sivanesan, S., Masud, M., & Hossain, M. S. (2021). Hybrid smart grid with sustainable energy efficient resources for smart cities. *sustainable energy technologies and assessments*, 46, 101211.

[26] Amal F.Abd El-Gawad , Shereen Zaki , Esraa Kamal, A Survey on Machine Learning Techniques for Supply Chain Management, *American Journal of Business and Operations Research*, Vol. 2 , No. 1 , (2021) : 24-38

[27] Noura Metawa , Mohamed Elhoseny, Feature Selection Optimization Model for Business Risk Assessment Model, *American Journal of Business and Operations Research*, Vol. 2 , No. 1 , (2021) : 51-64

[28] Kopal Sharma , Vidhi Kapoor , Kartikay Laddha, Ethics in the Financial Industry, *American Journal of Business and Operations Research*, Vol. 3 , No. 1 , (2021) : 39-47

[29] Ali Alfartoosi , Mohd Abdullah Jusoh , Hussein Jaleel Mohsin , Harith Yas, The effect of e-accounting and mediated by internal control system on the performance of SME in Iraq, *American Journal of Business and Operations Research*, Vol. 3 , No. 1 , (2021) : 05-38

Noura Metawa , Saad Metawa, Internet Financial Risk Early Warning Based on Big Data Analysis, *American Journal of Business and Operations Research*, Vol. 3 , No. 1 , (2021) : 48-60