LOAD BALANCING MECHANISM IN FOG COMPUTING TO MINIMIZE WORKLOAD ON SERVER

Ms. Sakshi Dinkar Junare [1], Dr. Manjiri Karande[2]

Student[1], Padm. Dr. V. B. Kolte College of Engineering, Malkapur,

Assistant professor[2], Padm. Dr. V. B. Kolte College of Engineering, Malkapur,

sakshijunare@gmail.com, manjirikarande@gmail.com

Abstract: *To mitigate the shortcomings of cloud computing for latency-aware applications, fog computing has been a recent distributed architecture. By pushing computation and storage out to the edge of the network, fog systems provide location awareness, mobility management, and considerably lower delays. Efficient load balancing is imperative in fog networks to avoid unequal resource allocation across fog nodes. Load balancing optimization can improve Quality of Service (QoS) aspects, such as resource utilization, throughput, cost, response time, performance, and energy efficiency. This paper gives a systematic solution of load balancing techniques in fog computing, classifying them into approximate, exact, fundamental, and hybrid approaches. In addition, it examines the loads balancing metrics, explaining the advantages and disadvantages of every method. The methodologies and tools used for evaluation in the analyzed studies are also discussed. Lastly, in this paper we determined the most significant open issues and future research directions for load balancing mechanisms in fog computing.*

Keywords: **Fog computing, load balancing, Quality of Service, Metrics, Mechanisms**

1. INTRODUCTION

Fog computing, a geographically dispersed architecture that extends the capabilities of clouds, advances computation and networking closer to the end-user and IoT devices by taking advantage of ubiquitous fog nodes. In conventional cloud-focused systems, data that needs processing and storage is sent to remote cloud servers, which may result in higher latency, security issues, and lower mobility and reliability. With the advent of location-aware and latency-sensitive applications, the native delays of cloud-exclusive solutions become an issue. Fog computing solves this by situating computational resources close to IoT devices, radically cutting latency and fulfilling strict real-time requirements. Seamlessly integrating and complementing cloud services, fog computing makes possible a new wave of applications.

In contemporary fog environments, clients require responsive and efficient applications. Load balancing is an important aspect in the provision of high Quality of Service (QoS) in fog networks. Although a lot of research has been carried out on load balancing in cloud computing because of its rising workloads, fog networks pose some special challenges. Their dynamism and heterogeneity make most cloud-based load balancing schemes inappropriate.

The goal of load balancing in fog computing is to distribute the incoming workloads across existing fog nodes or the cloud using an appropriate strategy to avoid node overload or underload. Load balancing optimization is to achieve maximum throughput, performance, and utilization of resources with minimal response time, cost, and energy usage.

2. LITERATURE REVIEW

As a result of the overwhelming volume of data and the Internet connection of more than 50 billion devices (according to Cisco estimation), processing that volume of data with the old models of computing, such as cloud computing, distributed computing, etc. is challenging. Sometimes privacy loopholes, elevated communication delay, associated network traffic loads connecting cloud computing to end-users for unforeseen reasons with recent growth in IoT-related services (such as smart cities, eHealth, industrial environments, smart transport systems, etc.) are some of the issues that impact cloud computing performance. To speak of a few of the limitations of cloud computing and to take cloud service characteristics so much closer to "Things," as it is termed, such as cars, mobile phones, embedded systems, sensors, etc., the research community has proposed the concept of fog computing.

Fog computing is also viewed as a platform that takes cloud computing to the proximity of end-users. "Fog" as a concept has a comparison with natural fog and was first proposed by Cisco. When the fog is closer to the ground, clouds are above in the sky and, curiously, fog computing uses the same idea, when the virtual fog platform is placed close to end users just between end-users' devices and the cloud. In the same definition, fog computing is proposed to enable computing at the network edge, to transmit new services and applications specially for Internet future. Bonomi, et al., to give a more appropriate definition of fog computing for the first time, said that fog computing was not exclusively located at the network edge. However, it was a virtualized platform providing networking services, storage, and computations among the data centers and end devices of conventional cloud computing.

Fog computing is most commonly confused with edge computing, but we have significant differences between the two.

Fog computing operations are executed within a multi-layered structure that decouples and interweaves the software and hardware operations, allowing the dynamic reconfigurations for various applications while performing transmission services and smart computations. Edge computing, however, establishes a point-to-point transmission service and handles special applications within a rigid logic position. Whereas Fog computing is hierarchical in nature, edge computing is confined to a limited number of peripheral devices. Apart from computing and networking, fog computing addresses control, storage, and speeding up data-processing. An IoT client or smart end-device, in order to identify fog computing from other computing standards, must use the following features but not necessarily all of them while consuming a fog computing service.

3. CHALLENGES IN FOG COMPUTING

Fog computing as the advanced form of the cloud computing system designed to manage IoT related issues and limitations at the network edge. But, in fog computing, processing nodes are heterogeneous and spread. Also, the services by fog technology need to operate with different perspectives of the controlled environment. Besides that, security assurance is overarching in fog computing. Thus, finding out the problems of fog computing from service-oriented, structural, security viewpoints in this technology can be enumerated as follows:

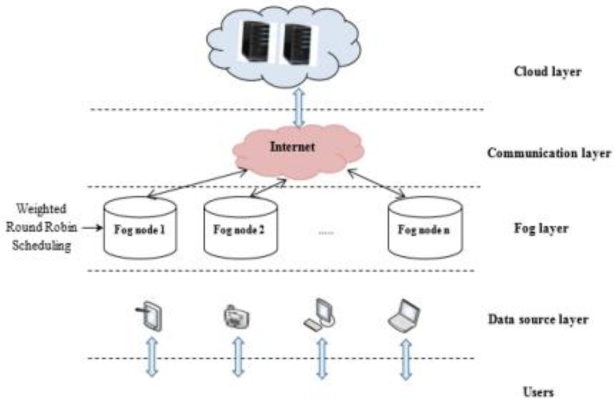

Figure.1 System Architecture

3.1 Service-Oriented

Resources enrich not every node of fog. Thus, complete scale application improvement in nodes with limited resources is not inherent with respect to conventional data centres. Thus, distributed application development requirements for possible programming platforms in Fog have to incorporate. Additionally, a fog administrator must demystify the policies to allocate necessary tasks between sensors/IoT devices, fog infrastructure.

3.2 Structural Issues

The fog computing infrastructure includes different components of core as well as verge networks. These kinds of components have a different computation but are not meant for common computing. So, redesign or modified the computation unit for the component is a very difficult section of the system setup. Furthermore, Depending on execution tasks and operational needs, choice of the appropriate device, deployment locations, and related resource planning are also important in fog computing. Computing devices are distributed over network borders in fog computing and can be virtualized or shared. Here, appropriate metrics need to be defined, inter-nodal collaboration strategies, as well as effective resource allocation.

3.3 Security Aspects

Fog computing depend on traditional network devices, it is extremely vulnerable to security attacks. Maintenance of privacy and authorized access to computing and storage resources in a highly distributed architecture, like fog computing, is tough to guarantee. Thus, preserving QoS is tough during the deployment of security, where the integrity of data-centre is sufficient and makes security subject in fog computing difficult.

4. RESEARCH OBJEVTIVES

4.1 Education on Load Balancing through Simulation

The primary aim is to have a good grasp of the concept of load balancing in a distributed system. This shall be accomplished through simulation where computational load is being fed into a server. Observing how the server processes the incoming load, the researchers shall assess its efficiency, performance, and capacity. This assists in detecting the necessity for load distribution between many systems, particularly if one server acts as a bottleneck.

4.2 Load Distribution between Two Servers with the Assistance of Fog Computing

The second aim expands the research to the idea of fog computing, which consists of processing data near the data source (i.e., network edge). In this configuration, the computational burden is split and shared between two distinct servers or fog nodes. This split lessens latency, enhances response time, and levels out resource utilization. The application of fog computing showcases how data and tasks can be carried out efficiently without complete dependence on centralized cloud systems.

4.3 Storing Independent Files on Other Servers to Reproduce Decentralization

Thirdly, the goal targets the decentralization of storage. For example, the study entailed saving two disparate files on two isolated servers. This arrangement is meant to mimic a distributed computing scenario in which no one server contains all the data. It supports the concept of fault tolerance and data availability, and fits with contemporary trends like edge computing and blockchain stores.

4.4 Processing Decentralized Files for Quicker Solutions

The ultimate goal is to process the files distributed here separately on their own servers. As each server is processing a portion of the total data or load, the processing is quicker and more efficient. Decentralization assists in avoiding delays, improves system scalability, and yields faster responses to user requests or system operations.

5. PROPOSED SYSTEM

Cloud load balancing is referred to as the distribution of workload and computing characteristics in cloud computing. It allows businesses to handle workload demands or application demands by distributing resources across several computers, networks or servers. Cloud load balancing entails controlling the flow of workload traffic and demands across the Internet. Traffic on the Internet is increasing extremely fast, representing nearly 100% of existing traffic each year. Thus, the load on the servers is growing so fast, resulting in overloading of the servers, primarily for the favorite web servers. Two main solutions to eliminate the issue of overloading on the server are-

•The first one is a single-server approach where the server is replaced with a higher-performance server. The new server will likely get overloaded in a short period as well, requiring another upgrade. Additionally, upgrading is cumbersome and costly.

• The second is a multi-server solution where an expandable service system on a server cluster is constructed. That is why it is more scalable and cheaper to construct a server cluster system for network services.

• Cloud servers can obtain more accurate scalability and availability through the utilization of farm server load balancing. Load balancing is effective with nearly any kind of service, including HTTP, SMTP, DNS, FTP, and POP/IMAP.

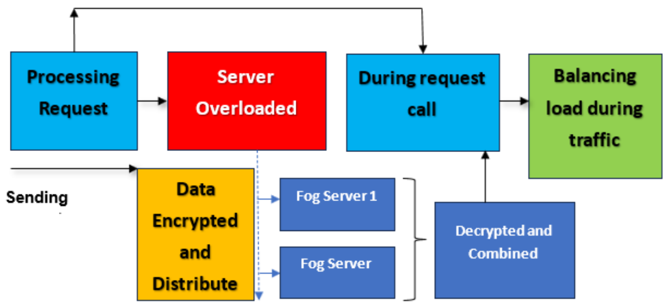

Figure.2 Flow of the system

6. RESULT

The experimental environment was used to mimic and study the performance of a decentralized fog computing system with emphasis on load balancing and safe file management. The system was set to upload, distribute, encrypt, share, and decrypt files between two servers. The findings are illustrated in the subsequent figures:

6.1 Dashboard for Uploading Files on Server

This image depicts the user interface or dashboard created to upload files. It provides users with the ability to choose and send files to the system. This dashboard serves as the point of entry for the data that will subsequently be shared and processed on many servers.

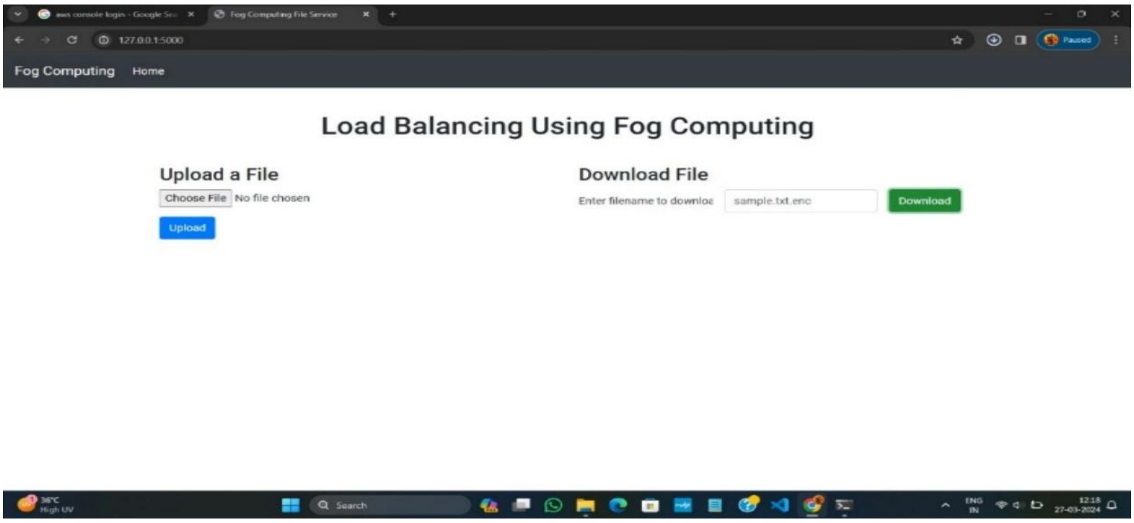

Figure.3 Dashboard for uploading files on server

6.2 Two Servers Created for Sharing the Data

Upon file selection, the data are dynamically distributed to two separate servers by the system. This configuration mimics a decentralized system, where the load is divided among several fog nodes. The figure indicates how the architecture prevents any one server from dealing with all the data, thus facilitating load balancing.

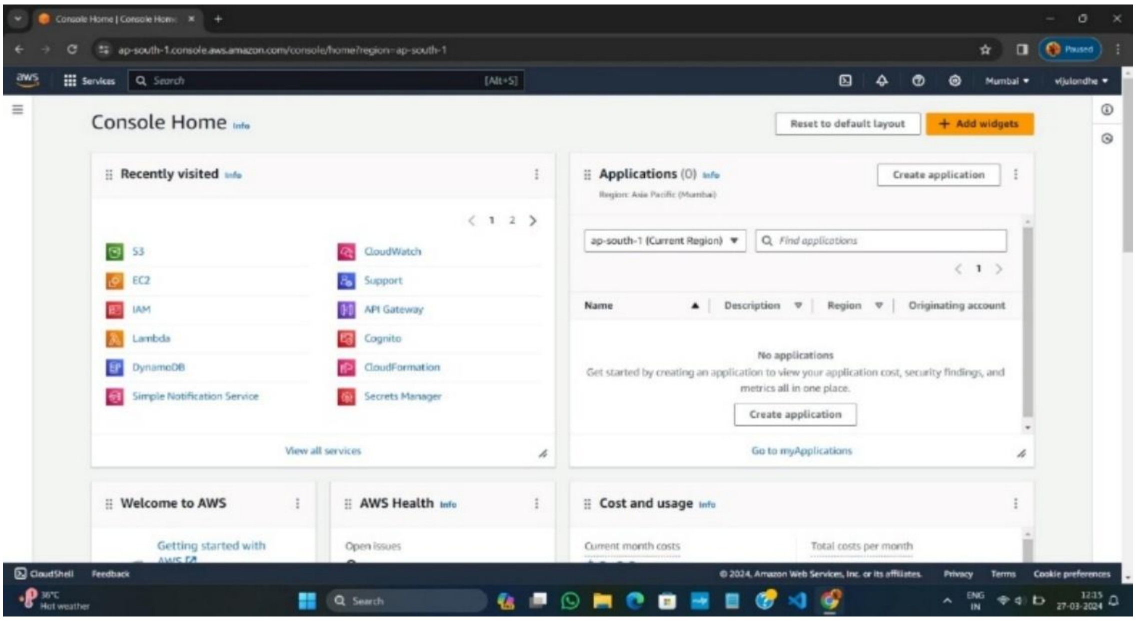

Figure.4 Two server created for sharing the data

6.3 Upon Uploading, File is Encrypted and Shared

After a file is uploaded, it is encrypted to provide security for the data while in transit and storage. The encrypted file is then shared between the two servers. This process illustrates how security has been integrated into the decentralized system to keep sensitive data out of unauthorized hands

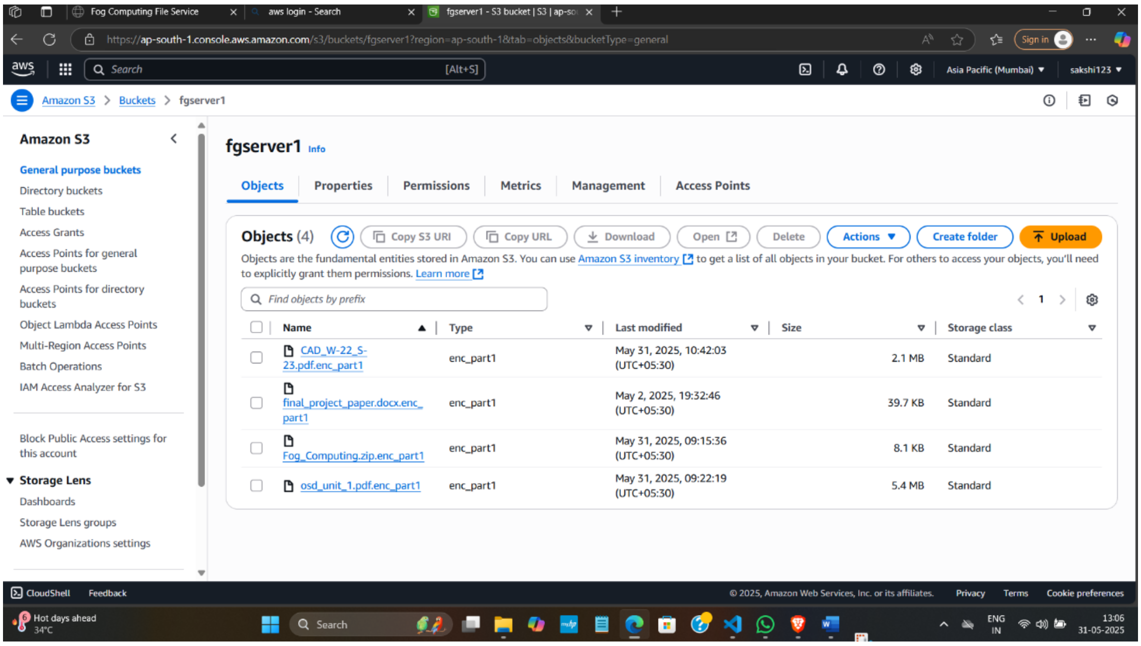

Figure.5 After uploading file encrypted and shared

6.4 During Decryption, File Was Restored

In the last step, the file is decrypted and successfully retrieved on the client side or desired destination. This ensures that the system not only encrypts and disseminates the data but also provides solid retrieval and decryption of the data when required. The retrieved file verifies the end-to-end operation of the system.

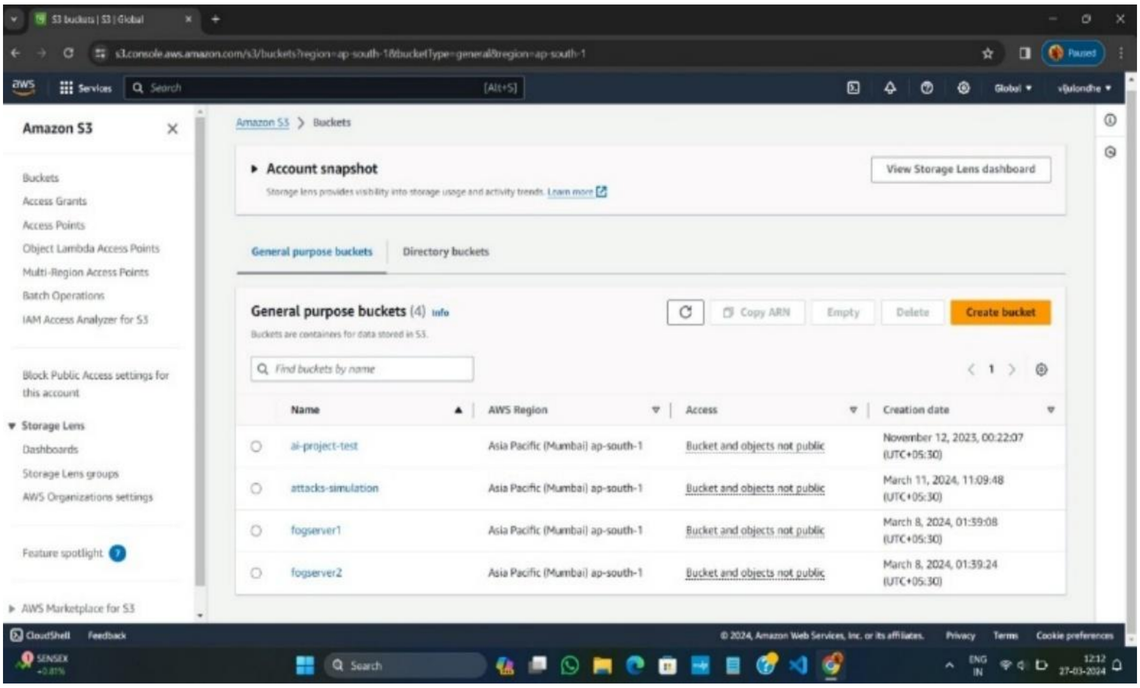

Figure.6 During decryption file was restored

7. CONCLUSION

The provisioning of services and processing capabilities across edge and cloud networks remains an area of focus in industry and academia alike because of the large potential it holds. This research tackles the problem of task scheduling in this emerging environment so that tasks are efficiently executed based on the processing capacity and energy available. Web applications today often endure simultaneous access from many clients, posing significant load handling challenges that result in system instability. To counter this problem, the proposed system utilizes fog computing to dynamically adjust workloads, thus allowing for more seamless request handling and less latency.

REFERENCES

11.1. Journal Article

[1] *Yousefpour, A., et al. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. Journal of Systems Architecture, 98, 289-330.*

[2] *Auluck, N., Azim, A., & Fizza, K. (2019). Improving the Schedulability of Real-Time Tasks using Fog Computing. IEEE Transactions on Services Computing.*

[3] *Brereton, P., et al. (2007). Lessons from applying the systematic literature review process within the software engineering domain. Journal of Systems and Software, 80(4), 571-583.*

[4] *Kitchenham, B., et al. (2009). Systematic literature reviews in software engineering – A systematic literature review. Information and Software Technology, 51(1), 7-15.*

[5] *Jatoth, C., Gangadharan, G. R., & Buyya, R. (2017). Computational Intelligence Based QoS-Aware Web Service Composition: A Systematic Literature Review. IEEE Transactions on Services Computing, 10(3), 475-492.*

[6] *Wieringa, R., et al. (2005). Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. Requirements Engineering, 11(1), 102–107.*

[7] *Kashani, M. H., Rahmani, A. M., & Navimipour, N. J. (2020). Quality of service-aware approaches in fog computing. International Journal of Communication Systems.*

[8] *Rahimi, M., Songhorabadi, M., & Kashani, M. H. (2020). Fog-based smart homes: A systematic review. Journal of Network and Computer Applications, 153, 102531.*

[9] *Abkenar, S. B., et al. (2020). Big data analytics meets social media: A systematic review of techniques, open issues, and future directions. Telematics and Informatics, 101517.*

[10] *Hu, P., et al. (2017). Survey on fog computing: architecture, key technologies, applications and open issues. Journal of Network and Computer Applications, 98, 27-42.*

[11] *Asghari, P., Rahmani, A. M., & Javadi, H. H. S. (2019). Internet of Things applications: A systematic review. Computer Networks, 148, 241-261.*

[12] *Marín-Tordera, E., et al. (2017). Do we all really know what a fog node is? Current trends towards an open definition. Computer Communications, 109, 117-130.*

11.2. Book

[13] *Buyya, R., Vecchiola, C., & Selvi, S. T. (2013). Mastering Cloud Computing: Foundations and Applications Programming.* ***Morgan Kaufmann****.*

[14] *Erl, T., Puttini, R., & Mahmood, Z. (2013). Cloud Computing: Concepts, Technology & Architecture. Prentice Hall.*

11.3. Chapter in a Book

[15] *Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2014). Fog Computing: A Platform for Internet of Things and Analytics. In Bessis, N., & Dobre, C. (Eds.), Big Data and Internet of Things: A Roadmap for Smart Environments (pp. 169–186). Springer.*

[16] *Mahmud, R., Kotagiri, R., & Buyya, R. (2018). Fog Computing: A Taxonomy, Survey and Future Directions. In Buyya, R., & Dastjerdi, A. V. (Eds.), Internet of Things: Principles and Paradigms (pp. 103–130). Morgan Kaufmann.*

11.4. Conference Proceedings

[17] *Manjoth, Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog computing and its role in the internet of things. In Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing (pp. 13-16). ACM..*

[18] *Aslam, S., & Shah, M. A. (2015). Load balancing algorithms in cloud computing: A survey of modern techniques. In 2015 National Software Engineering Conference (NSEC), IEEE (pp. 30-35).*

[19] *Jamshidi, P., Ahmad, A., & Pahl, C. (2013). Cloud Migration Research: A Systematic Review. IEEE Transactions on Cloud Computing, 1(2), 142-157.*

[20] *Mouradian, C., et al. (2017). A comprehensive survey on fog computing: State-of-the-art and research challenges. IEEE Communications Surveys & Tutorials, 20(1), 416-464.*

11.5. Patent

[21] *Mahmud, R., Kotagiri, R., & Buyya, R. (2018). Fog Computing: A Taxonomy, Survey and Future Directions. In Buyya, R., & Dastjerdi, A. V. (Eds.), Internet of Things: Principles and Paradigms (pp. 103–130). Morgan Kaufmann.*

[22] *Mahmud, R., Kotagiri, R., & Buyya, R. (2018). Fog Computing: A Taxonomy, Survey and Future Directions. In Buyya, R., & Dastjerdi, A. V. (Eds.), Internet of Things: Principles and Paradigms (pp. 103–130). Morgan Kaufmann.*