IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-6, June 2019 E-ISSN: 2347-2693

A Quality of Service Aware Model for Fog Computing

Shikha Kamboj[1*], Jitender Kumar[2]

[1, 2] Department of CSE, DCRUST, Murthal, India

*Corresponding Author: shikhakamboj08@gmail.com, Tel.: +91-8950394966

DOI: https://doi.org/10.26438/ijcse/v7i6.636640 | Available online at: www.ijcseonline.org

Accepted: 20/Jun/2019, Published: 30/Jun/2019

Abstract— Recent years have witnessed the extension of cloud computing paradigm to the IoT devices. In cloud computing, traffic rate and data transmission time are very high which causes delay and reduces Quality of Service. In order to counter these limitations, many researchers introduced a concept of fog layer. The concept of fog layer has many limitations such as the availability of limited number of servers for serving the incoming requests. To counter the resource limitation problem of fog layer, this paper proposes a mechanism which not only utilizes the cloud resources optimally but also helps to maintain the Quality of Service (QoS). The main purpose of this paper is to present Quality of Service aware model for fog computing and decreases SLA violation in fog computing environment. In this paper, we models and evaluate our proposed mechanism in CloudSim- a framework for modeling and simulation.

Keywords— *Fog Computing, Quality of service (QoS), Internet of things (IoT), Virtual Machines (VMs).*

I. INTRODUCTION

Internet of Things shapes future of connectivity, processing and reachability. With the help of Internet of Things (IoT) every "thing" is connected to internet. Internet of things is becoming very popular everywhere in this world. IoT devices increase rapidly, and we can assumed that IoT devices generated large amount of data in very less time, which may also require instant processing. Different IoT applications, like augmented facts, connected and autonomous automobiles, drones, and actual-time manufacturing have very illiberal latency necessities, and most cases it can be 10 ms [1] [2] but such applications does not allow these large and unpredictable latency of cloud, due to the fact these cloud resources deployed very far from wherein software statistics are generated. IoT devices helps to authorize things to see and sense the domain, to build synchronized selections, and based on these observations it can helps to perform different tasks [3]. For realizing the benefits of IoT devices, there is need of providing enough networking and computing infrastructure which helps to support less latency and quick response time for IoT application. Cloud computing paradigm seems to be the main reason of IoT applications or devices because it provides an ideal virtual environment for analyse the collected data and also provides computational resources. But, being very far from end users, IoT system that are supported by cloud, faces many challenges that includes quick response time, more load and absence of worldwide mobility. Instead of uploading data to the cloud paradigm, it can be more efficient to move the apps and processing abilities near by the data that is produced by the IoT devices. To overcome these problems a concept required to be proposed, and then, fog computing or edge computing proposed for these challenges by many researchers. Fog computing is defined as a concept whose main motive is to bring the cloud paradigm near to the end users to maintain quality of service [4], [5]. Fog computing paradigm is defined as a coating that sits among cloud coating and IoT coating and brings location awareness, lower latency and huge-spreaded geographical distribution for the IoT devices. The concept of fog computing was described to bring storage, network resources and methodical arrangement of computation between regular clouds and end users by CISCO in 2010 [6]. The concept of fog layer faces many limitations such as the availability of limited number of servers for serving the incoming requests. To overcome the resource limitation problem of fog layer, this paper proposes a mechanism which not only utilizes the cloud resources optimally but also maintains QoS.

In this paper, we focus to maintain the QoS at fog layer for end users and decreases SLA Violation in fog computing environment. The other segment of paper is structured as: In Section 2 we will discussed related work. Then, we will discuss the system model in section 3. Section 4 describes Simulation setup for proposed work and section 5 compare the results with existing research. Finally, section 6 consists of conclusions drawn.

II. RELATED WORK

Fog computing is an emerging field that enlarge the cloud computing to the edge of network and defines new apps and services. The traditional idea behind fog computing was firstly introduced by Bonomi et.al. [5].

Bonomi et al. [5] argues that the characteristics of Fog computing makes the Fog platform a suitable thing for a number of IoT apps and services, like Smart Cities, Connected Vehicle, and Smart Grid, etc. They visualize the Fog paradigm to be a consolidate platform, which helps to convey a new variety of emerging services and entitle the evolution of new applications.

Yousefpour et al. [7] introduces a general framework to evaluate, understand and model service delay in IoT-cloud-fog application scenarios and to minimize the service delay by sharing load for fog nodes they proposes a minimizing policy of delay.

Yousefpour et al. [8] introduce dynamic fog service provisioning problem to meet Quality of Service constraints that is they brings low latency, reduces bandwidth and location awareness to the IoT devices.

Maiti et al. [9] presents a mechanism to decrease service latency and consumption of energy with the help fog computing. To reduce service response time for overall network traffic they also model an inexpensive architecture.

Hong et al. [10] proposed a fog computing platform for dynamically locating modules on fog devices and also proposes an efficient Module algorithm for deploying that helps in maximizing number of satisfied requests but there exists many disputes of their proposed platform. Firstly, creating smaller modules by decaying larger modules is difficult because of heterogeneous devices. Secondly, it is difficult to connect smaller modules with dynamic flows is a critical problem which we have to solve.

Kapsalis et al. [11] introduces a fog computing architecture that helps to serve computational needs by using both stable and mobile devices. They also discussed the concept of fog message, that uses a specific MQTT (Message Queue Telemetry Transport) message format that contains all information for task allocation and they also evaluated workload balancer module.

Souza et al. [12] proposes a service allocation problem for fog platform called as “QoS- aware” which solves basic problem of optimization.

Skarlat et al. [13] and Skarlat et al. [14] proposes a framework for the service provisioning problem in the fog computing paradigm. They proposes a concept of fog cells from these simulation studies that is defined as a software running on IoT nodes for handling different services.

Sarkar et al. [15] lights upon a theoretical modelling of fog computing architectures, in particularly, power consumption, service delay, and cost. In basic models, they had formulated power consumption and service delay, and they does not introduced any policy for minimizing service delay.

Buyya et al. [16] proposes a policy in fog computing domain for latency aware application module management that considers distinct elements of dispersed applications in frame with fragmented coordination.

Buyya et al. [17] proposes a technique that attempts to meet QoS (Quality of Service) constraints, which request processing time (response time) and service requests rejection, even when preventing from over-provisioning of IT resources.

Mahmud et al. [18] defines the policy that optimize resource usage and ensure applications, QoS satisfying service cut-off times in the fog computing domain. They model and evaluated their proposed policy in an iFogSim tool - simulated fog domain.

III. SYSTEM MODEL

This section, present an overview of proposed system model whose key entities are IoT Devices, Broker, and Fog layer shown in fig 1. We also present an overview of sub entities of fog layer which is illustrated in Fig 2.

IoT devices are defined as the devices that transmits data from one object to another object with the help of internet. Sensors are also attached with these IoT devices with the help of which they can transmit data. IoT devices includes wireless sensors, actuators and computer devices.

Fog Broker: For task allocation and management as well as message received from lower layers fog broker is responsible [11]. In our proposed model broker acts as scheduler and QoS manager which is repository for tracking SLA Violation. IoT devices sends request to broker to get VM (Virtual Machine) then broker check all VMs busy or not if all VM are busy then broker add the request in waiting queue if not then immediately return VMs to IoT devices as they request for get VMs. After that if IoT devices get all VMs which they requested than these VMs are send to fog layer for processing. But if all requested VMs are not received then wait for some VMs to free after that sends to fog layer for processing.

Fog Layer have computing edges that operates in a Cooperative manner to execute different tasks [11]. In this proposed model, fog layer consists of two sub-components namely a Quality of Service (QoS) Manager and Virtual machines. QoS manager manages Quality of Service for proposed algorithm. Virtual machines are used for decreases the workload of proposed system.

Cloud Layer is mainly responsible for batch analysis and processing but it process data in longer-term data storage instead of processing it online [11].

In proposed work, we assume that all the request need to be satisfy at fog layer. So this work, is limited to IoT devices, broker, and fog layer.

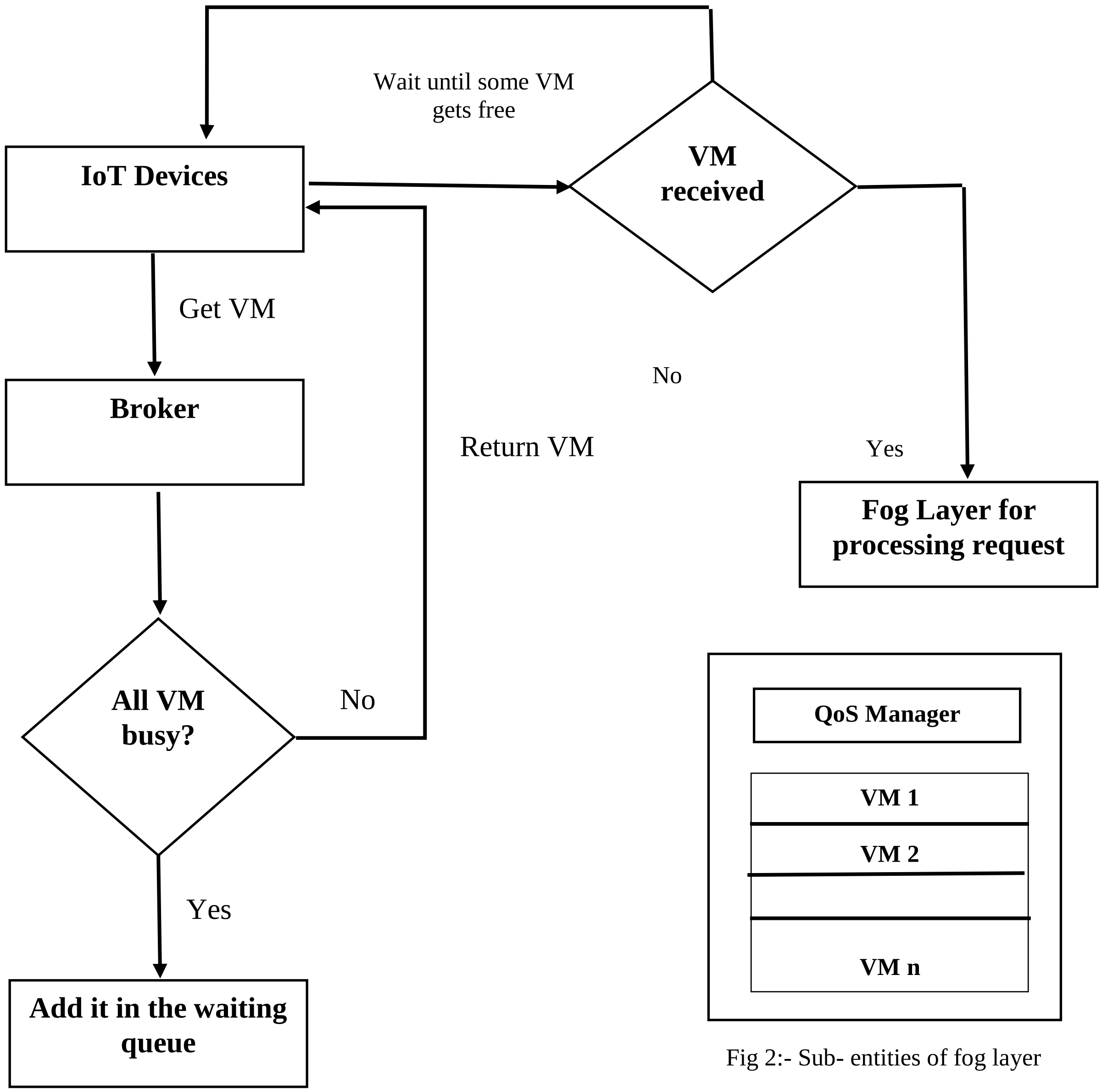

Fig 2:- Sub- entities of fog layer

Fig 1:- Working of our proposed model

QoS Manager: - The number of pragmatic application instances that are required to maintain Quality of Services is decided by QoS manager. It maintains the record of request arrival rate (λ) and service rate request (μ). This component serves the request like M/M/c queuing system on first come first serve basis. It estimates the number of virtual machines (VMs) according to SLA negotiated waiting time by using algorithm 1.

Algorithm 1

QoS Algorithm (λ, μ, max w)

1. int temp =1
2. ρ = λ / (temp * μ)
3. if (ρ >1) then
4. While (ρ>1) do

5. temp ++
6. Calculate $\rho = \lambda / (temp * \mu)$
7. end while
8. end if
9. Calculate W_q
10. if (W_q > max w) then
11. While (W_q > max w)
12. temp ++
13. Update W_q
14. end while
15. end if
16. Return temp
17. end QoS Algorithm

IV. SIMULATION SETUP

This section, present the experiments aimed to maintain Quality of Service (QoS) and decreases SLA violation in fog environment. Cloud-Sim tool is used to model our proposed work.

The simulation model is composed with one data center which contains 50 hosts. Every single host have one octacore CPU and 16 GB of RAM. Virtual machine for the application requires one core CPU and configuration 2.4GHz. We follow a space shared policy to maintain Quality of Service. A process owns the virtual machine until it finishes its execution and it minimizes the free RAM according to the number of processing elements. Output parameters collected for every individual scenario are: VM hours, VMs utilization and SLA Violation simulation of each scenario.

We simulated the proposed scenario for 10 hours where request arrival rate is normally distributed. During the peak hours, the minimum requests arrival rate is 2 per minute and maximum requests arrival rate is 10 request per minute. However, during the low request hour's minimum requests arrival rate is 2 requests per minute and maximum request arrival rate is 5 requests per minute. 2409 requests are the total number of requests during 10 hours of simulation. We took each request length is 30,000 MIs (Million Instructions). The input and output data of each request is taken as 1000 Kbs. To maintain Quality of Service, we kept maximum waiting time as 5sec. We compared our proposed model with fixed VM provisioning schemes using fixed 17, fixed 20, fixed 23 and fixed 26 VMs.

V. RESULTS AND DISCUSSION

Below figures i.e. Fig 3, Fig 4, and Fig 5, shows the results where QoS is labeled to represent our proposed mechanism, whereas fixed 17, fixed 20, fixed 23 and fixed 26 represents different VM provisioning schemes.

From below Fig 3, it is depicted that system operating at fixed 17 VMs results in best utilization of fog resources at the cost of highest SLA violations. On the other end, the system operating at fixed 26 avoids SLA Violation at the cost of least utilization resources. However, the system proposed in this work achieves utilization rate near to the fixed 17 VMs with very lesser SLA Violation rate of 2.32 %. So, the proposed work is not only beneficial for the service providers but also maintains the QoS for the end users without suffering from under or over utilization.

We also estimated VM hours in our work for all of the system. Figure 5 depicts that the proposed system spans VM hours near to the fixed 17 VMs system. Such type of system are also beneficial for data centers as it reduces the energy utilization in each data centers. In addition, such systems can be used by SaaS (System as a software) provider for reducing the VMs releasing cost.

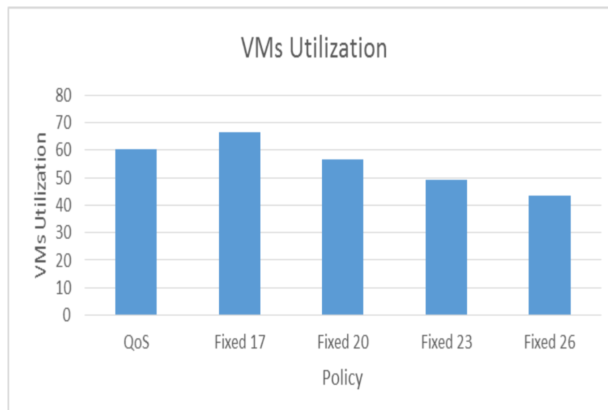

Fig 3:- Comparison of QoS with fixed VM provisioning schemes for calculating VMs utilization.

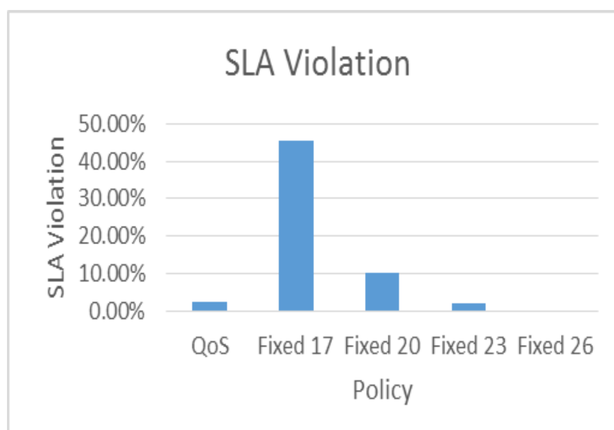

Fig 4:- Comparison of QoS with fixed VM provisioning schemes for predicting SLA Violation.

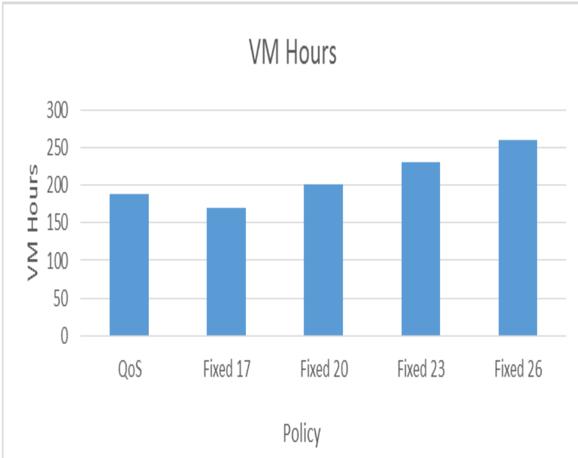

Fig 5:- Comparison of QoS with fixed VM provisioning Schemes for estimated VM hours.

VI. Conclusion

In this paper, a mechanism has been proposed that helps to overcome the resource utilization problem and maintains Quality of Service for end users and decreases SLA Violation in fog environment. The VM hours were estimated in this work to depict the total span. The simulation results of the proposed work achieved enhanced VMs utilization along with lesser SLA Violation as compared with the existing approaches. The system proposed in this work achieved utilization rate equivalent approximately to the method namely fixed 17 VMs with very lesser SLA Violation rate of 2.32 %. With the help of proposed approach one can easily maintain Quality of service for end users.

References

[1] C. C. Byers, "Architectural imperatives for fog computing: Use cases, requirements, and architectural techniques for fog-enabled IoT networks," IEEE Communications Magazine, vol. 55, no. 8, pp. 14–20, 2017.

[2] R. S. Montero et al., "Extending the cloud to the network edge," IEEE Computer, vol. 50, no. 4, pp. 91–95, 2017.

[3] A. Al-Fuqaha et al., "Internet of things: A survey on enabling technologies, protocols and applications," *IEEE Communication Surveys Tutorials*, vol. 17, no. 4, pp. 2347-2376, 2015.

[4] L. M. Vaquero and L. Rodero-Merino, "Finding your way in the fog: Towards a comprehensive definition of fog computing," *ACM SIGCOMM Computer Communications Review*, vol. 44, no. 5, pp. 27–32, 2014.

[5] F. Bonomi et.al. , "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, 2012.

[6] Chiang et.al. , "Fog computing and Networking: Part 1 [Guest editorial]" *IEEE Communications Magazine*, 55(4), pp.16-17, 2017.

[7] Yousefpour et.al. , "Fog Computing: Towards Minimizing Delay in the internet of things" *at IEEE 1st international conference on edge computing*, PP. 17-24, 2017.

[8] Yousefpour et.al. , "QoS-aware dynamic fog service provisioning" arXiv: 1802.00800v1 [cs.NI], pp. 1-10, 2018.

[9] Maiti et.al., "Mathematical modeling of QoS-Aware fog computing Architecture for IoT services" *at emerging technologies in data mining and information security, Advances in Intelligent Systems and Computing 814*, pp. 13-19, 2019.

[10] Hong et al., "Dynamic module deployment in a fog computing platform*" in Network Operations and management symposium (APNOMS)*, 2016 18th Asia-Pacific, pp. 1-6, 2016.

[11] Kapsalis et al., "A Cooperative Fog Approach for Effective Workload Balancing" *IEEE Cloud Computing*, vol. 4, no. 2, pp. 36-45, 2017.

[12] Souza et.al., "Handling service allocation in combined fog-cloud scenarios", in Communications (ICC), 2016 IEEE International Conference on, pp. 1–5, 2016.

[13] Skarlat et.al. , "Resource provisioning for IoT services in the fog," in Service-Oriented Computing and Applications (SOCA), 2016 IEEE 9th International Conference on, pp. 32–39, 2016.

[14] Skarlat et.al. , "Towards QoS-aware fog service placement," in Fog and Edge Computing (ICFEC), *IEEE 1st International Conference,* pp. 89–96, 2017.

[15] Sarkar et.al. , "Assessment of the suitability of fog computing in the context of internet of things," *IEEE Transactions on Cloud Computing*, no. 99, pp. 1–1, 2015.

[16] Buyya et.al. , "Latency-Aware Application Module Management for Fog Computing Environments", *ACM Trans. Internet Technol.* 19, 1, Article 9 (November 2018), 21 pages, 2018.

[17] Buyya et.al. , "Virtual Machine Provisioning Based on Analytical Performance and QoS in Cloud Computing Environments", *in International Conference on Parallel Processing*, pp. 296-304, 2011.

[18] Mahmud et.al. , "Latency aware application module management for fog computing environments", at article 9, November, 2018.

Authors Profile

Ms. Shikha Kamboj pursed Bachelor of Technology from University Institute of Engineering and Technology, Kurukshetra University, Kurukshetra in 2017 and currently pursuing Master of Technology from Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Haryana India. Her main research work focuses on Fog Computing.

Dr. Jitender Bhardwaj is working as Assistant Professor in Deenbandhu Chhotu Ram University of Science and Technology, from Deenbandhu Chhotu Ram University of Science and Technology, Haryana, India. He has completed his B. Tech., M. Tech. and Ph.D. degree in the past and working as Assistant Professor for last few years.