Smart Ambulance Using Fog Computing

Kunal Meher, Saurabh Patil
Department of Computer Engineering, Xavier Institute of Engineering, Mumbai

Abstract--- **Fog is a recent and emergent architecture for storing, computing, controlling devices and their data while transferring this through networks across different end distributors and clients. It covers various wired as well as mobile scenarios for computing of data or information that is residing across various hardware and software devices making human life more comfortable. Here in particular, we discuss two scenarios in smart ambulance where we demonstrate the efficient storing of data at the hospital using network edge technology called fog computing and using sensors to control traffic signal. It reduces latency which is one of the drawbacks of cloud computing.**

Keywords – fog computing, edge computing, cloud computing, smart ambulance, latency

I. Introduction

Over the years, we seem to fancy our chances using the concepts of cloud for computing and storing information locally. Recently, cloud computing has paved way for fog computing erasing various problems that existed in the past. The current situation of traffic on the busy roads these days is drastic. Also in accidental cases the patient is delayed in the hospital as entry of details into the hospital cloud takes time. Based on these issues, we have given the overview regarding the redundancy and latency that was reduced at the end of the demonstration of this project.

The paper then discusses why we need the fog concepts, system working and various statistics regarding the computing and storing of the data. The main goal of this paper is to illustrate by making use of coding as to how efficiently data computing of patients in the moving ambulance can be done reducing latency. The paper also illustrates the controlling and storing of data at the hospital database reducing the time needed for storing locally. Previously, there were a surging number of death rates due to the delayed ambulance facilities as the traffic was not well controlled. This paper covers the problems like ambulance stuck in traffic, delay in data storing in the hospital which might delay the treatment in case of fatal accidents. The system is dynamic which changes based on the schedule of the transportation. This dynamic system which is the ambulance in this case when comes in the range of the traffic signal; the signal turns green making way for vehicles to move smoothly. Fog computing is fast and responsive and can be used in hospital for storing data in case of emergency.

The paper demonstrates the use of wifi enabled device for this local mobile networking. Further, the paper illustrates the use of network edge simulator for giving the real time execution on basis of speed of transfer, security and latency of the data.

A. Scope of the Project

Ambulances are often delayed to reach the hospital due to traffic jams because it is not possible for the vehicles to make way for the ambulance. The system consists of sensors, one of which will be installed at the traffic signal and the other one installed inside the ambulance. When the ambulance's sensor is within the range of the sensor which is installed in the traffic signal, the signal will stay green till the ambulance is out of the sensor's range. When the ambulance is outside the network, the traffic signal will work normally. Hence, the ambulance will not be delayed to reach the hospital due to traffic jam. The patient's diagnosis details carried out in the ambulance will be directly available to the doctor in the hospital through the local server working as fog computing agent. Hence, this will not require re-entering of patient details by the doctor in the ambulance and the patient's treatment will not be delayed.

B. Aim and Objective

To ensure that ambulances are not delayed to reach the hospital due to traffic jams and to make the patient's diagnosis details directly available to the doctors in the hospital which will prevent delay in the patient's treatment.

II. SYSTEM OVERVIEW

The system presented here is based on the concept of "IoT" and it follows "Waterfall Model" comprising of following functionalities:

(a) Ambulance and traffic signal interaction (b) Ambulance and hospital interaction (c) Prompt for connecting the traffic signal sensor with the ambulance sensor (d) Display patient diagnosis details.

(a) Ambulance and traffic signal interaction:

The traffic signal and ambulance are mounted with fog nodes i.e. wifi-enabled devices. When the ambulance is within the range of the traffic signal's device (fog node), the signal turns green. Once the ambulance crosses the signal and is out of range, the signal starts working normally.

(b) Ambulance and hospital interaction:

On the way to the hospital the details of the patient such as current condition, written formalities, etc. are stored in the ambulance's fog node. The ambulance mounted with the fog node connects to the hospital's local system (fog system) when it enters the hospital premises. The fog node exchanges the data of the patient with the hospital's system and this data is then provided to the doctors or hospital staff directly. The data stored in the hospital's local system is then sent to cloud for permanent storage. The data stored in the device is temporary i.e. for a day and later it is erased once it is stored in the cloud.

(c) Prompt for connecting the traffic signal sensor with the ambulance sensor:
The interface will be provided to the user in the ambulance with a beep sound in the device installed inside the ambulance, when the ambulance will come inside the network range of the device installed at the traffic signal. By pressing the connect button the ambulance can get connected to the system. This connection will be created between the fog nodes of the ambulance and the traffic signal.

(d) Display patient diagnosis details:
This interface will display the patient's diagnosis details which are made available directly with the concerned doctors through the local server working as fog computing agent. Hence, this will not require re-entering of patient details by the doctor in the ambulance and the patient's treatment will not be delayed. The system will work faster as it is making use of network edge technology called fog computing which will improve the efficiency and prevent the latency problem which is a drawback in the existing system.

Most of the existing real time smart systems require an active internet connection on user's device. Thus the users need a system that does not require for them to have active internet connection on their device. The system is fast, responsive and provides appropriate information.

III. IMPLEMENTATION AND RESULTS

A. Overview of Implementation

To implement the proposed method, we are using iFogSim toolkit. It basically consists of a GUI that allows the user to draw physical elements such as fog devices, sensors, actuators, and connecting links. The drawn topologies are saved and reloaded by converting the topology to JSON file format. The amount of CPU and RAM used is calculated. There is less usage of CPU and RAM in the fog topology when compared to the cloud topology.

B. Software and Hardware Requirements

(a) Software details: To implement the proposed and existing method iFogSim toolkit is used. iFogSim models the IoT and Fog environments and measures the impact of resource management techniques in terms of latency, network congestion, energy consumption, and cost. The simulator supports evaluation of resource management policies focusing on their impact on latency.

(b) Hardware details:

All the experiments were conducted on:

a. Operating System: Window 8 and above version.
b. Processor: Intel Core – i5-3337
c. CPU: 1.80 GHz
d. RAM: 4 GB

C. Experimental set up

To perform the experiment, we consider two topologies, first cloud topology and second topology fog topology.

(a) The following Fig 1 represents the cloud topology.

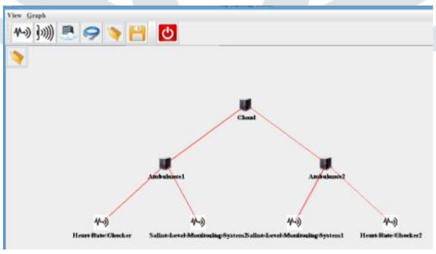

Fig. 1 Cloud Topology

(b) The following Fig. 2 represents the fog topology.

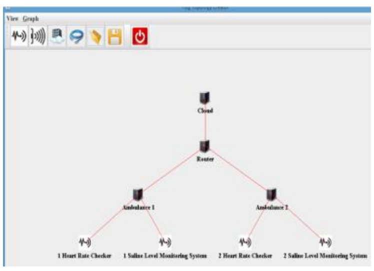

Fig. 2 Fog Topology

D. Results and Discussion

The scalability of the simulator depends upon its resource use (in particular RAM) and the time it takes for the simulation to execute. The heap allocation does not increase considerably with increasing workload and physical topology size. The figures obtained shows that iFogSim scales with minimal memory overhead when the number of sensors and gateways increases.

(a) The amount of heap used by CPU in cloud topology:

The CPU consumes a lot of heap in cloud topology. The heap size is 62,914,560B and the CPU uses 24,620,136B. Initially it comes less heap and later the amount of heap used increases.

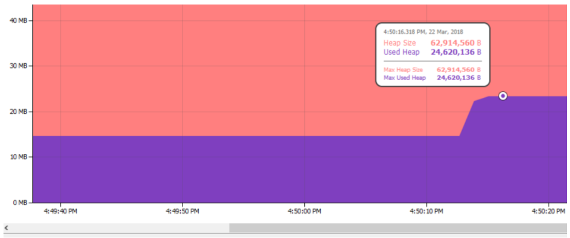

Fig. 3 Heap used in cloud topology by CPU.

(b) The amount of heap used by CPU in fog topology:

The CPU in fog topology uses 23,763,848B out of 62,914,560B. The amount of heap used is comparatively less than cloud topology.

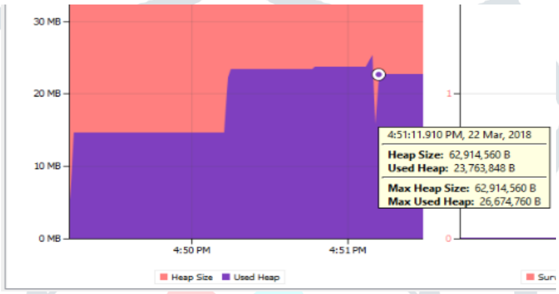

Fig. 4 Heap used in fog topology by CPU.

E. Testing

This project implementation includes 2 modules: Ambulance and traffic signal interaction along with ambulance and hospital interaction. The traffic signal, ambulance and hospital are mounted with fog nodes. The system will work faster as it is making use of network edge technology called fog computing which will improve the efficiency and prevent the latency which is a drawback in the existing system. This system assists the delay caused due to traffic congestion.

(a) Module 1: Ambulance and traffic signal

→Before the Ambulance enters the range of signal:

The traffic signal will work normally before the ambulance enters the range of the traffic signal.
GREEN → YELLOW → RED.

→After ambulance enters the range of the traffic signal:

Once the ambulance enters the range of the traffic signal, the hotspot created by the ambulance is captured by the wifi enabled device at the traffic signal which turns the traffic light to green until and unless the connection is destroyed.

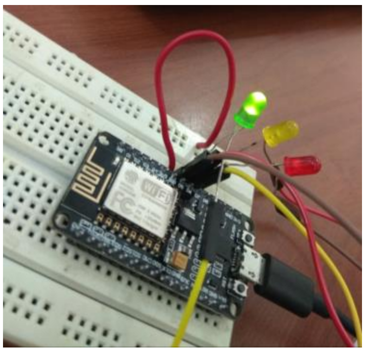

Fig. 5 Green LED ON

(b) Module 2: Ambulance and hospital

Here we have used the ambulance as a client and hence hospital becomes the server enabling data to be sent locally to the cloud. The project uses FTP server along with bash commands to execute and store data on the fog node.

IV. CONCLUSION

We successfully tested the network edge technology called fog computing to store data efficiently at the hospital and to control traffic signal. It reduces latency which is one of the drawbacks of cloud computing.

References

[1] Elias Tragos "A Fog Elias Tragos "A Fog Enabled IoT Platform for efficient management and data collection System", Journal of Advanced Computer Science and Technology Research, Vol.3 No.1, March 2017, 60-70.

[2] Ashkan Yousefpour and Mena Saeed "Fog Computing :Towards minimizing delay in Internet Of Things", International Journal of Advanced Computer Science and Applications, Vol. 4, No. 6, 2017.

[3] Lin, W.-H. and J. Zeng. "Fog Computing for Internet of mobile things: Issues and Challenges". In Transportation Research Record: Journal of the Transportation Research Board, TRB, National Research Council, Washington, D.C., 2016,

[4] Kidwell,B."Fog Computing for minimizing delay for IoT." Science Laboratory, Bridgewater State College, Bridgewater, Mass., 2016.

[5] Arwa Alrawais; Nelson, A.Y.; Sowah, R.A.; Bremang, A., "Challenges and software architecturte for Fog", Adaptive Science & Technology (ICAST), IEEE 4th International Conference on, vol., no., pp.1,6, 25-27 Oct. 2014

[6] Walid Saad and C.Prakesh (2015), "An online Secretary framework using fog", World academy of science.