**ADVANCING CONFIDENTIALITY IN IOT HEALTH ASSISTANCES UTILIZING FOG COMPUTING**

**ABSTRACT**

Internet of Things (IoT) is the interconnection of physical objects or devices that can transmit and receive data through the internet without human involvement. With the advancement in IoT devices particularly in healthcare sector, huge amount of data is collected from different sensors and all this data are transferred and stored in cloud. It becomes difficult to handle such huge amount of data in cloud specially the healthcare data where it requires real time data computation and storage. Security of the data is also major challenge in cloud. Fog computing is the answer to overcome the challenges. Fog nodes works at the edge side and enhances data security, accuracy, consistency and reduces the latency rate which is an important factor for application like medical data. Implementation work is also described in the paper where a digital human temperature sensor device is built using DS18B20 temperature sensor. The data collected from it is being encrypted in fog node using Advance Encryption Standard (AES) algorithm and it is send to cloud. Therefore, the security of the health care data is enhanced using Fog computing.

**INTRODUCTION**

In the world of technology where we are living in, almost all the devices are connected to internet. The number of IoT devices are increasing in an exponential rate and all these devices are relying on cloud computing system for data computation and storage. It becomes a bottle neck problem when it comes to real time data operation which is the major drawback in the existing IoT healthcare system . In order to overcome the problem Fog Computing concept has been introduced. Fog Computing in an archetype that extends the cloud computing platform.

Fog acts as a middle layer between the cloud server and the end devices. It is not the complete replacement of cloud, rather it Fig1: Fog Computing Architecture complements the functionality of cloud. Fog works closer to the edge devices and provides computing resources to these devices. Fog computing overcomes the scalability and reliability issues which is there in the traditional IoT-cloud architecture. Since Fog nodes works at the edge side and more geographically distributed as in (Fig:1), it enhances data security, accuracy, consistency and reduces the latency rate which is an important factor for application like medical data. As well as the overall bandwidth to cloud is saved, thus achieving better quality of service (QoS).

The term fog computing shifts capabilities of the cloud near to the end user, and provides storage, computation, and communication to edge devices, which facilitate and enhance mobility, privacy, security, low latency, and network bandwidth so that fog computing can perfectly match latency-sensitive or real-time applications. On the one hand, fog computing infrastructure consists of plenty of fog nodes, edge device networks, and even virtualized data Centers or IoT devices that are connected to these nodes. These are connected to the cloud for the purpose of implementing large storage and rich computing.

The distribution of functions between the cloud and the fog nodes is considered a crucial factor. Millisecond to sub-second latency offered by fog, even faster than real-time interaction, supports multitenancy and performs better in low-latency applications. The concept of fog computing has been designed to satisfy the applications that require low latency with a real-time response such as healthcare IoT systems.

**SYSTEM ANALYSIS**

**EXISTING SYSTEM**

physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effor

, physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effort

, physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effort

N , physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effort

, physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effort

, physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effort

, physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effort

, physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effort

, physics-based, machine learning, and

statistical models have all been used to model average fuel

consumption. The EPA and the European Commission devel-

oped physics-based, full vehicle simulation models for heavy

duty vehicles [1], [2]. These models are capable of predicting

average fuel consumption with an accuracy of ±3% compared

to real measurements obtained from a ﬂowmeter [2]. This level

of accuracy comes at the cost of a substantial development

effort

Currently fog uses the Decoy system as a security service from malicious attacker. Like in Cloud, Fog uses this method to trick the attacker by providing fake data when they try to extract the data. In the decoy system the user has to sign up then login, while logging in the system will ask security questions related to information given while signing up. So when an attacker tries to login her/she will be trapped with the question and the system will give back spurious file which is very such similar to the original file and when the attacker tries to download it will turn out to be a fake data. But there is chance that the attacker might guess the questions right. Therefore, this system is not a very good way of securing data

**DISADVANTAGES OF THE EXISTING SYSTEM**

❖ In the decoy system the user has to sign up then login, while logging in the system will ask security questions related to information given while signing up

❖ It becomes a bottle neck problem when it comes to real time data operation which is the major drawback in the existing IoT healthcare system

**PROPOSED SYSTEM**

In the proposed system a three tier architecture model is considered as in fig2. First layer will be the Edge devices which will collect the data and this data will be transferred to the middle layer. The middle layer will be the fog layer; encryption process of the collected data will be performed in this layer. The encrypted data from the middle layer will then be send to the third layer which is the cloud layer. In the cloud the final encrypted data will be permanently stored.

**ADVANTAGES OF PROPOSED SYSTEM**

(1) In order to overcome the problem Fog Computing concept has been introduced. Fog Computing in an archetype that extends the cloud computing platform. Fog acts as a middle layer between the cloud server and the end devices.

(2) it enhances data security [4][5], accuracy, consistency and reduces the latency rate which is an important factor for application like medical data. As well as the overall bandwidth to cloud is saved, thus achieving better quality of service(QoS)

**MODULES**

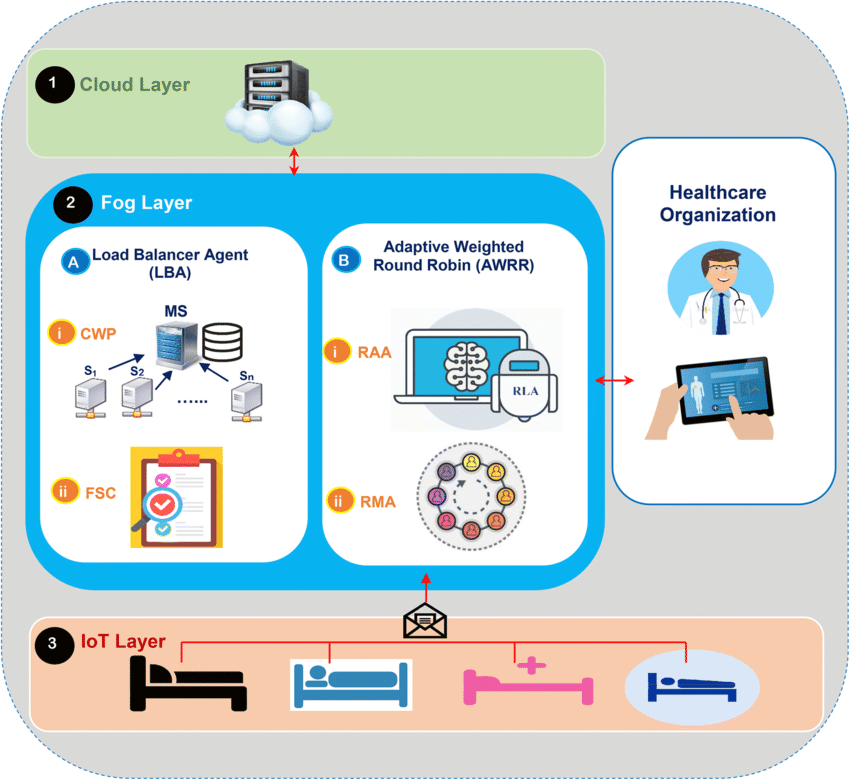
1. User

2. Owner

3. Fog Node

4. Cloud

**SYSTEM ARCHITECTURE**

****

**SYSTEM REQUIREMENTS**

**HARDWARE REQUIREMENTS**

•**PROCESSOR :**I3

**• Hard Disk** : 40 GB.

• **Ram :** 2GB

**SOFTWARE REQUIREMENTS**

**Operating system :** Windows

**Coding language :** JAVA/J2EE

**Data Base Server :** MySQL

**IDE :** Netbeans8.1