AUTOMATED DATA CENTER AND DISASTER RECOVERY SITES ENVIRONMENTAL MONITORING USING IIOT ENABLED TEMPERATURE SENSOR AND HUMIDITY SENSOR

Atshaya. G[1]

[1] Student, Department of Computer Science, Panimalar Engineering College, Tamil Nadu, India

ABSTRACT

In this era of modernization, lots of system has been introduced by which the human effort has been limited to a certain level. This paper is proposed to automate the monitoring using Kaa IIOT Platform enabled temperature and humidity sensor. The Bluetooth TI Sensor Tag packs contains many sensor but the temperature and the humidity sensors are the important parameters in the operation of Datacenter. TI Sensor tag gathers information from data center by measuring temperature and humidity parameter and transmit the collected data to Kaa endpoint. Kaa endpoint uses Kaa SDK to communicate to Kaa server. In this paper, client application is created where sensor devices transmit data to server. Kaa client process structured data provided by the Kaa server (configuration, notifications, etc.) and to supply data to the server and finally display the data in dashboard. Kaa server uses MongoDB for storing data. So in brief, Kaa-powered gateway simplify data collection from sensor that uses PAN-based protocol such as Bluetooth and sending out the bulk of data to Kaa server. The main aim of this paper is to create cheaper alternative solution in Data center infrastructure management (DCIM) and also provides the ability to remotely monitor the Data center from anywhere within the premises of the company.

Keywords: - *Data Center, Environmental Monitoring, Temperature and humidity sensor, TI Sensor Tag, Kaa IOT.*

---

1. INTRODUCTION

Environmental threats monitoring in the data center is not a new concept. Since the beginning of modern data centers, the temperature of servers and network equipment has been a constant concern as one of the most threatening environmental conditions. Monitoring environmental threats in the data center is very essential in order ensures that services of a data center are delivered without any interruptions or abnormalities. The servers need to be "protected" from environmental threats and operated only under the right conditions. The methods and mechanisms to monitor for physical threats in the data center have also evolved. In this paper, Environmental monitoring system has been explored, one of the newest developments in data center physical threats.

2. ENVIRONMENTAL THREATS IN THE DATA CENTER

Before we get into best practices for monitoring environmental threats, let's first review the physical threats to the data center and impact of these threats. Temperature [1]. The first and most important physical threat to the data center is temperature. There are many physical locations in the data center where temperature is critical: Air temperature in the plenum spaces such as raised floors, especially far away from the CRACs (Computer Room Air Conditioning), Air temperature above the plenum and in front of the racks on cold aisles (i.e., the intake air for the servers in the racks), Air temperature inside the IT racks themselves, especially if the racks are fully enclosed (doors front and rear as well as sides). Humidity [2]. Humidity is like air temperature in that it can vary throughout the data center, although it typically does not vary as much. In data center, humidity readings varies with the outdoor humidity conditions. Most modern CRACs (computer room air conditioning unit) have the ability to control (increase or decrease) the humidity in the data center. The danger of low humidity conditions in the data center is threat of electrostatic discharge (ESD), which can damage electrical components. The result of high humidity in the data center is the possibility of condensation. ASHRAE currently recommends a range of 55%-60% for data center humidity

Threat	Applicability	Impact	Sensor
Temperature	• Data Center • Inside Racks • Plenum Spaces	High temperatures reduces equipment life span	Temperature Sensor
Humidity	• Data Center • Inside Racks • Plenum Spaces	Electrostatic buildup at low RH and condensation at high RH	Humidity Sensor

Table-1 Physical threats Summary

3. PROBLEMS IN THE EXISTING SYSTEM

At BHEL Bharat Heavy Electricals Limited, Data Center (DC) and Disaster Recovery (DR) sites are dedicatedly maintained on 24x7 shift operation. Sites are maintained in the established range using an air conditioning system. With the help of **Kusam-meco temperature meter**, readings are taken. Drawback are [1] Hourly temperature and humidity reading of both DC and DR are manually monitored, [2] Meters are fixed only at one corner of room, [3] Reading are not accurate.

4. PROPOSED WORK

In this proposed work, a new system is designed to monitor the data center environment and send that information to user using Kaa IOT platform. Data Center and Disaster Recovery sites are embedded with Texas Instrument Sensor Tag. This gathers information from data center by measuring temperature and humidity parameter and transmit collected data to Kaa endpoint. Kaa endpoint uses Kaa SDK to communicate to Kaa server. Kaa server uses MongoDB to store the data and finally fetch the data to dashboard. Client application is created where sensor devices transmit data to server and endpoint SDKs is generated for these applications using Kaa Sandbox. Kaa client is connected to Kaa server by using Kaa endpoint SDK. This endpoint SDK is a library that is used to create a Kaa clients. Advantage are [1] it provides accurate reading with low cost, [2] Human effort has been limited to certain level, [3] It will deliver the information to everyone (authorized person), anytime, anywhere within the premises of the company. So this system is economically feasible.

This system is also technically feasible because it reduce the development time from years to months, minimizes the cost and risk of application development, Kaa is a highly flexible open-source platform for building, managing, and integrating applications in the Internet of Things, Kaa manage an unlimited number of connected devices, perform real-time device monitoring, components used in this proposed system are easily replaceable in case of any damage or malfunctions and also The installation process also won't take more days. The components used in this project are affordable price and capital budget needed for this project to implement in the field also low, so this project is financially feasible.

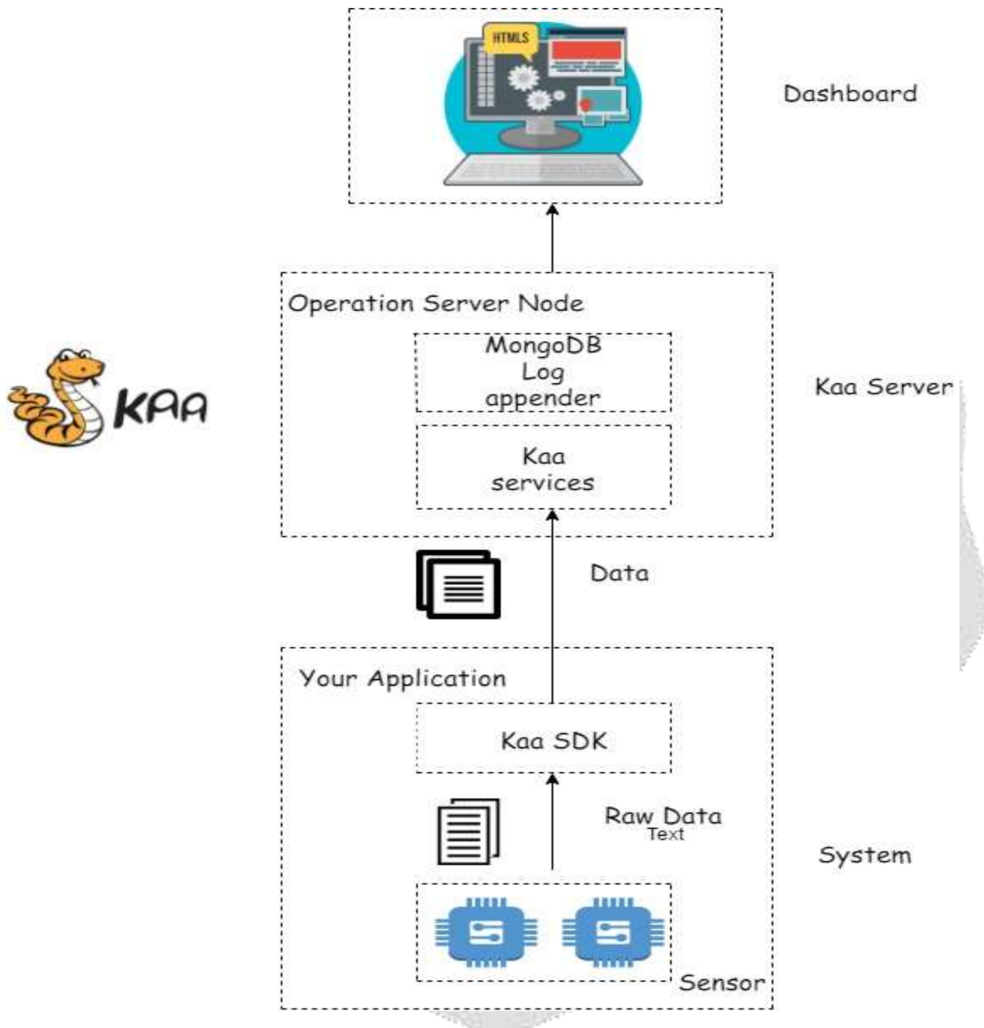

Fig -1 Architecture diagram of proposed system

Texas Instruments (TI) Sensor Tag are embedded in the Data Center site. This TI Sensor tags has built-in ambient temperature and humidity sensors which collects the environment parameters from data center. The built-in Bluetooth BLE beacon of the sensor tag transmits the raw data wirelessly to the KAA endpoint.

At KAA endpoint machine, it scans the datacenter for sensor tags and receives the raw data. The received raw data is first converted and then it is transferred to KAA Server using Java SDK. The Java SDK is generated using Kaa Sandbox. Kaa endpoint SDKs are available in Java, C++, and C, and are designed to be embedded into client application. The end point application is developed using Java code, SDK jar file and slf4j jar file.

Using client application, sensor tag data is collected and transmitted to KAA server. KAA server stores the data in a no-SQL database like MongoDB and MongoDB uses JSON-like documents (JavaScript Object Notation) with schemas. For this project, the KAA server is installed using KAA sandbox client. Sandbox client is a virtual machine on Ubuntu Linux with MongoDB preinstalled. SDK is generated using JSON to implement real-time bi-directional data exchange with the KAA server and end points. KAA client process structured data provided by the KAA server (configuration, notifications, etc.) and supply data to the server and finally display the data in dashboard. A user-friendly dashboard is created using PHP, HTML & CSS to display the collected temperature and humidity data for the end user.

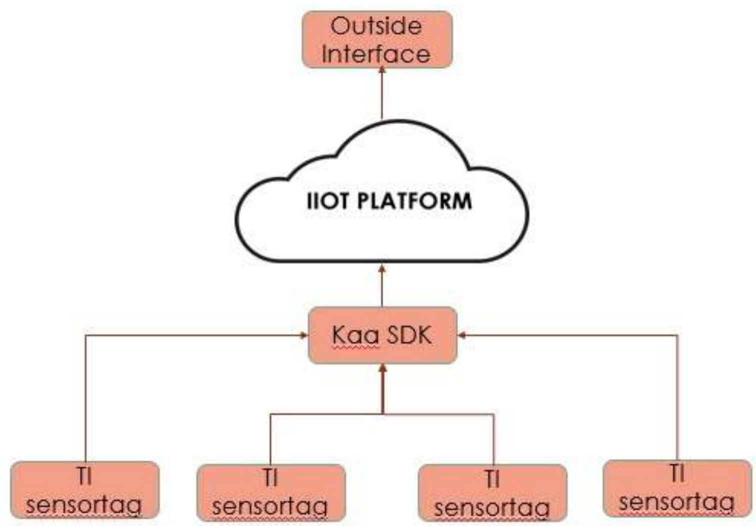

Fig -2 Control Flow diagram of proposed system

5.1. MODULES

5.1.1. FETCHING RAW DATA FROM SENSOR TO ENDPOINT

The Sensor Tag includes a CC2650 Bluetooth Low Energy Radio System on Chip with 256KB flash and 8KB RAM. The devices are preprogrammed with Bluetooth Low Energy software stack firmware for configuring the sensors and communicating with Bluetooth 4.0 enabled window system. The Sensor Tag is configured as a BT4.0 Peripheral device and can thus be connected to one Central device. The Sensor Tag uses I2C to interface to the sensors. Built-in ambient temperature and humidity sensors collects the environment parameters from data center. The built-in Bluetooth BLE beacon of the sensor tag transmits the data wirelessly to the KAA endpoint. KAA endpoint scans the data center for sensor tag. The connected sensor tag transfers the ambient temperature and humidity to the KAA endpoint.

5.1.2. CONNECTING KAA ENDPOINT AND KAA SERVER

KAA endpoint is a java application which connects sensor devices and KAA server. The data transfer is real-time bi-directional data exchange with the KAA server using SDK generated on JSON from KAA server. SDK is an executable jar file generated from the KAA server. The Data collection function will be used to transmit data from endpoint to KAA server at a configured sample period. The Configuration function will be used to send the sampling period values from KAA server to the temperature sensors. Thus SDK enables bi-directional data flow between KAA endpoints & KAA server. The received data is send to KAA server to be populated into database. In this project, MongoDB is used for storing data. So in brief, KAA-powered gateway simplifies data collection from sensor that uses PAN-based protocol such as Bluetooth and sending out the bulk of data to KAA server.

5.1.3. INTERFACING THE GATEWAY WITH DASHBOARD

The MongoDB is a no-SQL database. MongoDB uses JSON-like documents with schemas. Mongo DB stores all information for a given object in a single instance in the database, and every stored object can be different from every other. This makes mapping objects into the database a simple task, normally eliminating anything similar to an object-relational mapping. The stored data is retrieved using a user-friendly dashboard. The dashboard is created on PHP, HTML & CSS to display the collected temperature and humidity data for the end user.

6. CONCLUSION

The system automates the process of environment monitoring of data center. This system overcomes the drawbacks of manual monitoring like inaccurate reading, human errors etc., Also provides the ability to remotely monitor the Data center from anywhere within the premises of the company. This project provides a cheaper alternative solution in Data center infrastructure management (DCIM).

7. FUTURE WORKS

The system can be further enhanced to provide dynamic notifications via SMS or email whenever the temperature and humidity parameters cross the threshold values. Data analytics on the captured data to assess the trend on thermal load on the data center depending on the utilization factor of the servers.

8. REFERENCES

Books:

[1] Understanding wireless connectivity in Industrial IOT by Olivier Monnier, Eran Zigman, and Amit Hammer.

[2] Industrial IoT Technologies and Applications International Conference, Industrial IoT 2016 by Wan Jiafu, Humar, Iztok, Zhang, Daqiang.

[3] Data center fundamentals by Mauricio Arregoces, Maurizio Portolani.

E-learning sites:

[1] https://kaaproject.github.io/kaa/docs/v0.10.0/Programming-guide/

[2] https://www.kaaproject.org/industrial-automation/

[3] https://github.com/kaaproject/kaa

[4] http://www.cybervisiontech.com/kaa-platform/

[5] http://processors.wiki.ti.com/index.php/SensorTag_User_Guide

[6] http://www.oracle.com/technetwork/java/javase/downloads/index.html

[7]https://www.microsoft.com/en-gb/developers/articles/week04jun15/using-the-windows-bluetooth-generic-attribute-api-with-sensortag/

[8] https://www.kaaproject.org/wp-content/uploads/2016/03/Introducing-Kaa-0.8-Webinar.pdf

[9] http://www.ti.com/ww/en/wireless_connectivity/sensortag2015/tearDown.html

[10] http://processors.wiki.ti.com/index.php/SensorTag_Firmware

[11] http://processors.wiki.ti.com/index.php/Category:BluetoothLE#TI_BLE_Module_Partners

[12] http://blog.makezine.com/2013/04/18/teardown-of-the-ti-sensortag