Remote Monitoring of IoT Device: Cloud Computing & IoT

1st Ms. Hiral Lineswala
Engineering Student
K.J. Somaiya College Of
Enggineering Mumbai,India

2nd Mr. Pratik Swali
Engineering Student
K.J. Somaiya College Of
Enggineering Mumbai,India

***Abstract*-Remote Monitoring Of Iot Devices is an epitome of an IOT and Embedded Applications. It adds on the functionality of any existing system by providing an ease of In Application Update using Remote Programming capability. So, one can program or reprogram his /her device with some clicks from a remote location, when connected to Internet. That includes the capability to provision software to devices directly through OTA (Over the air) update. In addition, it provides provisioning like device tracking, live status update, reporting and monitoring. In the early days of IoT, updating remote de vices often caused intermittent disruption and performance degradation.**

The potential benefits of this are significant: minimizing labour costs, time and energy are saved, reduced need for on-site maintenance, improved device uptime/utilization, longer machine life due to preventive maintenance, longer lifetime of critical components.

Keywords — OTA, Firmware's, MQTT, Linux, IOT, Remote monitoring, Protocol, Status update

I. INTRODUCTION

Remote Monitoring of IoT device is nothing but a way of managing and monitoring on-field devices placed remotely with Internet access.

Internet of Things is the technology that will lead to developments and advancements in leaps and bounds. With the need of establishing a means of communication between dumb devices and machines, to make them smart, is the goal for Internet of Things. Remote Monitoring and OTA Updater works on the same Principles of IoT, allowing us to build Applications that will take Technology to the Next level.

The main aim of this was to implement a way of managing the on-field devices, tracking them and showing their real time parameters over Web Page.The dynamic web based application keeps the details of the device employed updated and hence provides a mean to keep a clean and organized record of them. The device details are stored in a database and are queried at regular time intervals. Hence any change in the details dynamically updates the database.

It also has a provision to monitor and display the in-system parameters of the IoT Devices, and remotely checks the status of the device, i.e. if the device is up and functioning properly as it is supposed to.

Thus, implementation of the Application over Internet of Things, by allowing the devices to communicate to each other, is the solution to the problem. The Internet of Things will help the Applications be smart enough to ease Human efforts to the least possible extent.

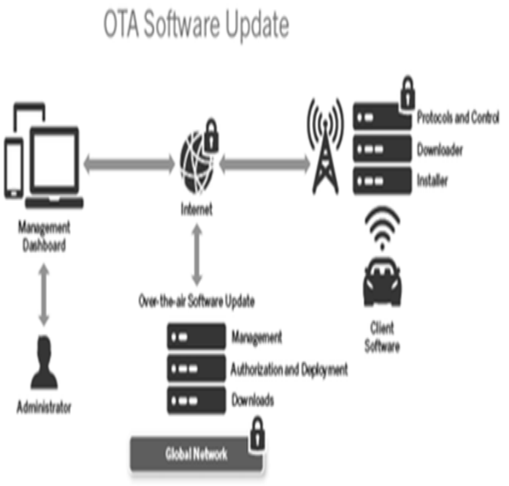
Thus, the tedious process of reprogramming is now simple, efficient and time efficient.

II. OTA OVER THE AIR

An over the air (OTA) update is a mechanism for remotely updating internet-connected hardware with new settings, software, and / or firmware. The OTA update mechanism is a core part of a system's architecture, with the remote hardware device being responsible for identifying and applying updates to itself, and the cloud server responsible for distributing updates to its connected hardware clients.

Why OTA?-Prior to OTA updates, you had to go out and retrieve the device, take it apart, connect it to your computer, reprogram it, put the device back together, and then take the device back. However, this process is overly burdensome and unscalable for companies who have devices out on the field. Hence an approach is made to use the OTA technology.

Thus, any new application can be directly programmed into the controller, without the need to be present at the location where device is installed. Also now we have a way of



keeping a track of deployed and ready devices, all with its live status.

Fig. 1. OTA Software Update

III. MQTT

Message Queuing Telemetry Transport is a machine-to-machine (M2M)/"Internet of Things" connectivity protocol. It was designed as an extremely lightweight publish/subscribe messaging transport. It is useful for connections with remote locations where a small code footprint is required and/or network bandwidth is at a premium. It works on top of the TCP/IP protocol. We'll be making a use of Mosquitto to implement MQTT.

The MQTT protocol is a good choice for wireless networks that experience varying levels of latency due to occasional bandwidth constraints or unreliable connections. Should the connection from a subscribing client to a broker get broken, the broker will buffer messages and push them out to the subscriber when it is back online.

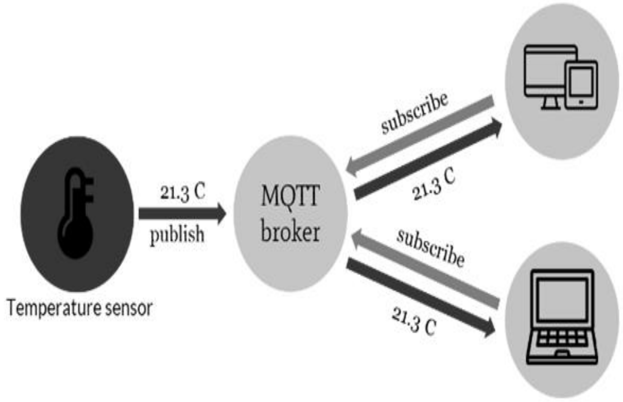


Fig. 2. Schematic data flow from sensor to device.

IV. MEAN STACK

The term MEAN stack refers to a collection of JavaScript based technologies used to develop web applications. MEAN is an acronym for MongoDB,Express.JS,Angular.JS and Node JS from client to server to database, MEAN is full stack JavaScript.

- Node.js- It is a server side JavaScript execution environment. It's a platform built on Google Chrome's V8 JavaScript runtime. It helps in building highly scalable and concurrent applications rapidly.
- Express.js- It is lightweight framework used to build web applications in Node. It provides a number of robust features for building single and multi-page web application..
- Mongo DB - It is a schema less MySQL database system. MongoDB saves data in binary JSON format which makes it easier to pass data between client and server.
- Angular.JS - It is a JavaScript framework developed by Google. It provides some awesome features like the two-way data binding. It's a complete solution for rapid and awesome front end development.

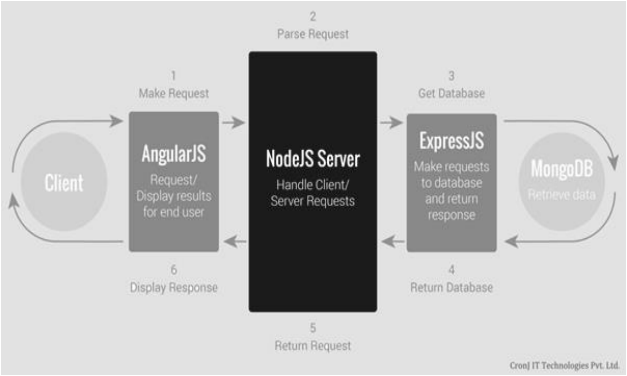


Fig. 3. MEAN Stack

V. FURTHER ENHANCEMENTS

The expansion of IoT applications allows more remote devices to wirelessly collect, store, and transmit information across vast networks and distances to multiple applications. This advancement now demands that remote IoT solutions be designed to have individualized device security, well thought out IoT hardware and with consideration of risk aversion because hackers now have a larger playing field with even more targets.

Remote IoT connected devices can be accessed from both wired and wireless networks, which leave them vulnerable to these basic types of attacks to consider:

Access/Authentication of IoT Devices – Hackers can cause mistrust by misleading remote network devices by altering the manufacturer code. Up-to-date security systems – Hackers can attack systems that have fallen behind on updates or lack support to patch issues in large numbers of scattered IoT devices. Encryption Network Security – Hackers can easily access and find encryption keys to decrypt IoT data. Hardware Port access Protection-Hackers can physically attack remote IoT devices and gain access through the JTAG port, network ports, or an Ethernet port.

The IoT solution to help prevent these cyber-attacks is to design and implement a futuristic IoT security framework. The security solution will be tailored to a specific IoT solution and will provide advance features like device authentication, using a remote system that will monitor and update devices. Remote services will also help store IoT data and validate that data as originating from the proper device. It will include a hardened coprocessor that add other layers of IoT security by enabling security functions separate from the main processor in a hardened security environment.

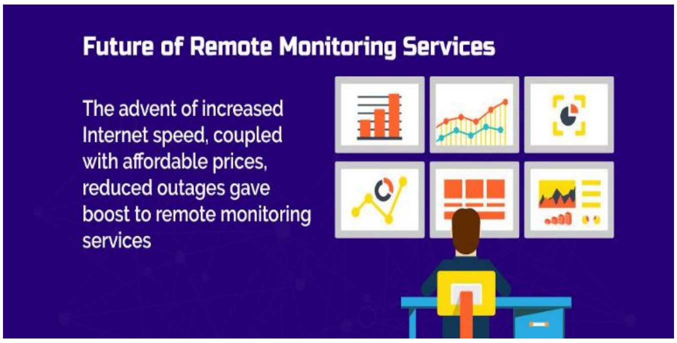

Future of Remote Monitoring Services
The advent of increased Internet speed, coupled with affordable prices, reduced outages gave boost to remote monitoring services

Fig. 4. Future of remote monitoring services.

CONCLUSION

There is a theoretical side and a practical side to knowledge and both are valuable. The true masters of any craft or discipline understand both ends of the spectrum. They put in the hours to acquire the practical techniques while also putting in the time to understand how those techniques fit into a larger context and tradition and why they work.

IT and control systems manufacturers are seizing the opportunity of having new novel hardware devices as the “Internet of Things” begins to scale up. As the number of devices continues to increase, more automation will be required for both the consumer (e.g.on-field devices) and industrial environments. As automation increases in IoT control systems, software and hardware vulnerabilities will also increase. In the near term, data from IoT hardware sensors and devices will be handled by proxy network servers since current end devices and wearables have little or no built-in security.

The IoT client device which is installed on the field, sends its in-system parameters like current core temperature, system uptime, host name and MAC address through MQTT. The same is received by the MQTT Broker, Mosquitto in this case and is passed to the server which in turn updates the database and displays the data dynamically on the webpage.

Thus, we have practically tested the remote monitoring of on-field devices and have been successful in doing the same. It is an ongoing project with the proposed plan for remote OTA (over-the-air) update.

"Just keep the Device connected, We will do the rest".

REFERENCE

Books:

[1] Data Communication by Behrouz Forouzan.

[2] Getting MEAN with – MongoDb, Express, angular and Node.js by Simon Holmes.

Links:

[1] MQTT Related:
https://www.hivemq.com/blog/mqtt-essentials-part2-publish-subscribe

[2] MEAN STACK:
https://dzone.com/articles/mean-stack-introduction-part-1