IoT Services Agent Platform: A Case Study in Network Monitoring

Anirach Mingkhwan*, Wipoo Suvunnasan*, and Panya Thongkumdee*

Abstract

IoT Services Agent Platform of Network Monitoring using IoT is a work that was utilizing The Internet of Thing (IoT) devices for gathering information not just from the SNMP equipment's but also has an advantage from the specific functions that perform by the IoT agent itself. By adding an unlimited possibility to the IoT agent monitoring system it has helped to reduce the task of the system by summarizing, inferencing and calculating before sending it to the could monitoring system.

The prototype system platform has shown how to store and retrieve data from cloud storage show information of application dashboard. An efficient way of nodes management such as update scripts and firmware remotely also demonstrate in this paper. The platform presented could also be useful in another area that uses a group of IoT services agent.

Keywords: IoT platform, Network Monitoring, IoT Services Agent.

1. Introduction

IoT a new technology that is welcoming by many areas of application. It changes the way system work by embedded itself into the system to a gathering, control, processing data and many other aspects to the implementor imagination.

In the light of network changing from just router and switch to Software Defined Network (SDN) and more information from many sources like Big Data system flowing through the network and the internet.

An additional intelligence and more robust monitoring are needed to put there inside the network. One that we could reach out to change control config any time.

The main problem of using many IoT nodes in the system is that how to efficiently manage the nodes without having to reach each node one by one physically. How to find the right board for the job, one that can be adapted to any need of the system. In this work, we proposed and built the prototype system that uses cloud services platform to be demonstrated how to archive those need.

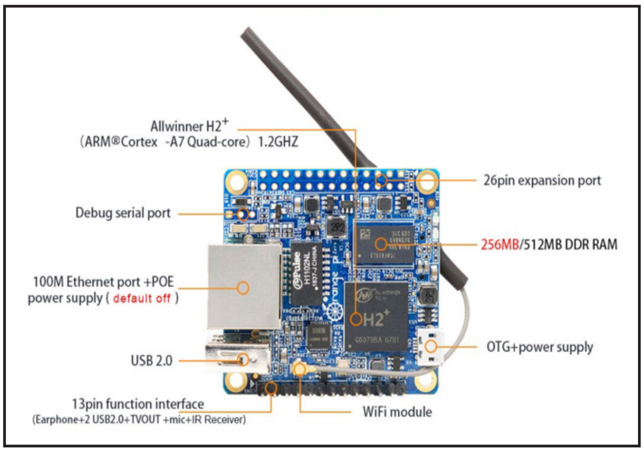

Figure 1. *Orange Pi Zero.*

In our prototype, we use Orange Pi Zero [1] as an IoT services agent that we deploy inside the network infrastructure. With the advantage of small size 46x48 mm, Power over Ethernet, 100M Ethernet port, Wi-Fi, Linux, 512MB and the price of 7$. Figure 1 shows the main component and features of the board.

We use Firebase, GitHub, Node.JS, HEROKU as a cloud service. Language and scripts using python, Cloud Functions

* *Department of Information Technology, Faculty of Industrial Technology and Management, King Mongkut's University of Technology North Bangkok.*

for Firebase, C, BASH. And Finally, alert any critical notification through LINE. We will describe how each of them working together in a later section.

2. Related Work

There are many works related to the area of using IoT services agent as both how to construct the node, sending information, power consumption, collaboration, and management. We show some of our exciting finding in this section.

Karim Saikali [5] proposed that the collaboration scenarios can be considered using the generic model of collaborations in three different levels. The Cloud level, which contains centralized, shareable or intensive computing logic. This level exposes services that used by lower levels and by logic running on other cloud platforms. The Edge level, which contains logic and functions that are consumed by groups of devices residing in the environment "supervised" by the edge node, and also by other edge platforms. And The device level that has application code run on it.

In our work, the collaboration is still on the early level of sharing information and comparing the same task of monitoring. Group collaborates control will be in our future work.

The work of Andrej Kos et al. [6] present applications for network monitoring based on intelligent communication platform that can also be used to support various us-age scenarios related to the future internet of things. Their applications presented include real-time DSL access line monitoring and IPTV monitoring, correlated with lightning reports. Their solution is used in the field of proactive monitoring, enabling the operator's helpdesk and technical teams to locate the cause of service degradations.

Their work concentrated on the usage of the network which is different from us that working on real-time devices performance and status.

Considering about power consumption and extensive area usage of the system we study the work of Xihai Zhang et al. [7]. They propose an LPWAN information monitoring approach based on NB-IoT and LoRa. In their system, the actualize of sub-node based on LoRa communication and the primary node combining NB-IoT and LoRa communication and a solar power supply system. They experiment and tests in real scenarios, the system enables the achievement of wide-area information monitoring. The collection information is accurate, and the system runs stable. The design of the monitoring node realizes low power consumption and provides a reliable guarantee for the long-term operation of the system that is the excellent example of using our IoT services agent outdoor for any other purpose.

The interesting of cloud-based wire-less sensor network example of S.G. Srinivasa et al. [8]. They proposed that Wireless Sensor Networks (WSNs) has become an integral part of weather monitoring applications in many domains such as environmental monitoring, healthcare, asset monitoring modern warfare scenarios, industrial and production monitoring. They found that available applications provide services to these domains on an hourly basis and are highly accurate. On the exploration and advancement in the field of Internet of Things (IoT), the focal point of their work has shifted towards the interoperability of WSNs and a cloud-based central data repository which collaborates and comprehends a uniquely identifiable internet-like structure. They pro-posed the bottom up an Internet-like structure to pave the way for a Sensor Integrated Cloud-based architecture call PARASENSE.

Wen-Tsai Sung et al. [9] proposed many ZigBee based monitoring systems that built from IoT technology in their work on the perception layer, using temperature/humidity sensors, light sensors, and 3-axis accelerometer modules. They have deployed sensors that wirelessly transmitted to a monitoring center. Their research work has a focus to achieve the practical way of power consumption reduction, improve the health care quality and provide a higher comfort level of a user.

From our finding, we found that there are many areas of research and application on IoT services agent. The prototype and platform that we proposed will have a great benefit for other researcher or developer work to build on.

3. Architecture

In this section, we presented our prototype and platform starting from the overall proposed idea, information flow of the system and finally the management of the IoT services agent code and firmware update solution.

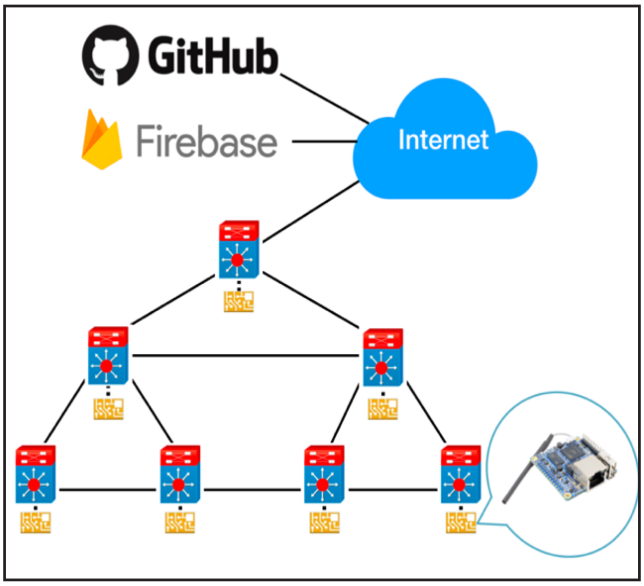

Figure 2. *Nodes deploy.*

3.1 The overall of the system

In our work as shown in figure 3. We deploy our IoT services agent inside the network by plug it to every switch/router that we want to monitor to use SNMP get the MIB value and perform the functions as we specify in the code on the board. The services we tested in our prototype are:

- SNMP data access
- Link Status
- inbound/outbound
- Link Utilization
- Link Quality
- IoT services agent functions
- Upload/Download
- Room condition
- Information summarization

By using the ability of the IoT services agent that can be processing data, we can do two unique things more, and one is the information exchange between agent on collaborating task as we pre-program and the other is the summarization prediction of the data before updating on to the could database or perform the trigger routine.

3.2 Information Flow

The system we designed to have the flow of the information as shown in figure 3.

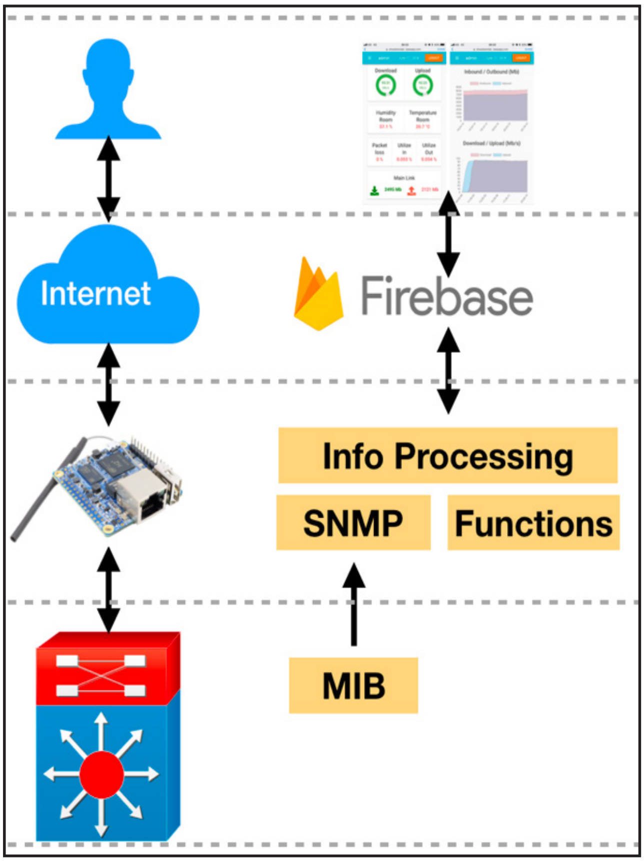

Figure 3. *Information Flow.*

The information flow of the SNMP data access starts with the IoT services agent using SNMP protocol to get the information of the network device from its Management Information Base (MIB) that collected by SNMP Agent. Data then processed to obtain the information needed and sent to store on the Firebase database on could. The user then accesses that information through their mobile device.

In the case of IoT services agent functions. The services function is activated from the application to perform the routine the get information such as upload/download or get the room condition value from the sensor then store it on Firebase and display on user application.

All of those information processing and sending are build using the software platform that explains in next section.

3.3 Software use in this work

Figure 4. *Software and Tools.*

As shown in figure 4. this implemented work a base on a platform that uses the combination of free software that is available for developer. From cloud base till the low-level IoT interface of C language as we describe below.

3.3.1 Cloud Level

At the cloud level, we use Node.js as the device back-end to run functions or services that we use for collecting data. Node.js also use to send Line-alert to inform user using Heroku when there is a trigger of an incident that has been a monitor such as a switch or a router down.

GitHub is where we store and manage our code project which provides the version controller. We also use GitHub to help us on OTA update to online update firmware and script that running on our IoT devices.

We use Firebase as our No SQL and real-time database to receive the collecting data and store it for retrieving by the application.

3.3.2 Software

In this work to operate the Linux operating system of the Orange Pi Zero, we use BASH to write a script. And by using BASH combining with Git-watch which is the services we put in the IoT board to work with Webhook to make the system can be remotely updated script and firmware as in diagram that shows in figure 5 and 6.

Vue.js Framework is used as our Responsive Web Application to show the information results that retrieve from Firebase.

The speed test function that implements in our work use Python calling Ookla speed test library. Node.js activated the speed test function.

C language is using as a low-level interface with the sensor that connects with IoT board. For the environment detection of the switch and router that we monitor we use DHT22 connect with the GPIO of Orange Pi Zero. The function of getting environment also called by Node.js script.

To manage the database space on Firebase that is limited. We use the new feature of Firebase that is called Cloud function to clear the data that we store on it every hour to prevent it from out of space.

3.4 Code Management

As we all know that there is not an easy way to manage many IoT nodes to keep the code and firmware up to date. We have the design to handle these tasks using the GitHub function call GitWatch to do the job as we show in figure 5.

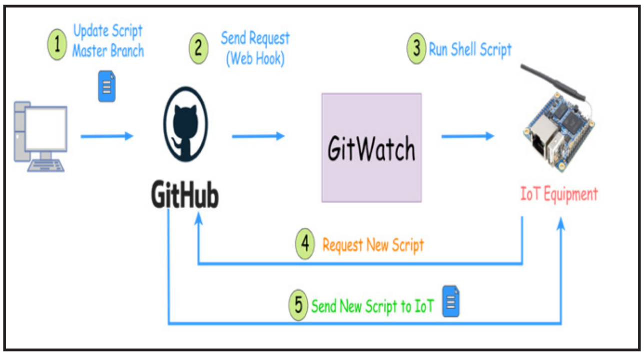

Figure 5. *Function Update.*

By updating the script in Master Branch of GitHub, the request will send by Webhook to GitWatch to run the script on IoT services agent which will request the new script from GitHub to send to an IoT services agent. We also use the similar idea to update the firmware of IoT services agent as shown in figure 6.

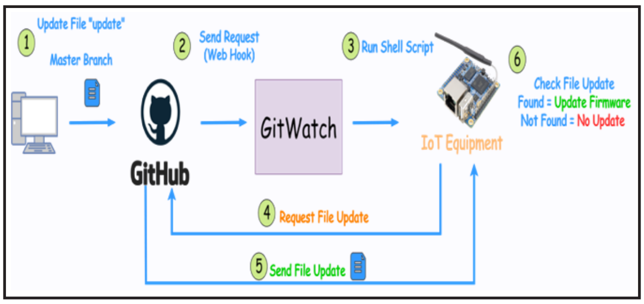

Figure 6. *Function Update.*

4. Results and Discussion

From the gathering data that we collected the system can show the information of each switch as in figure 7. This dashboard web application is display on any mobile device using web responsive technology which is convenient for the user to monitor their network.

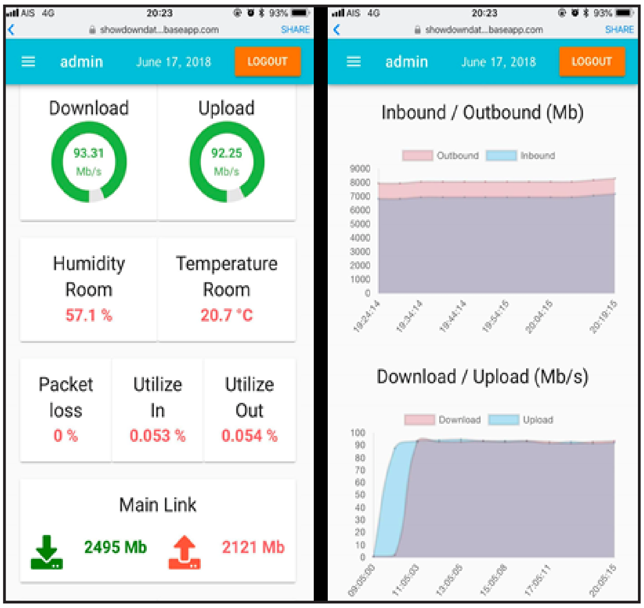

Figure 7. *The devices dashboard.*

In figure 7 the first part of the dashboard shows real-time download/upload performance of the network from this switch/router device which is performed by the function on the IoT board. The second part show information of room temperature and humidity that gets from the DHT22 sensor that we attach with the GPIO of our prototype Orange Pi Zero.

The next two parts that show on the dashboard are the data that we collected using SNMP to get the information from MIB on the switch/router that our IoT services agent connected with consists of packet loss and link utilization. Link utilization in/out is the calculation form the MIB link interface value. We also monitor the Main Link traffic to show inbound/outbound volume.

The second screenshot of figure 7 shows the graph comparing the value of inbound/outbound and upload/download to let the user know the performance over a period.

And this closely watch also can make the condition to alert the user through Line application notification. The urgent warnings that test in our prototype are high temperature, link down, unreachable network device and suspected inbound/outbound traffic volume.

Put together data form every node that we monitor the system can process the series of information to determine the flow and performance of the network.

5. Conclusion

The prototype was showing results that we can gather the information for real-time monitoring and alter the code to update functions of the IoT devices.

By collecting all the types of information of the network form, every IoT nodes into the database means its open up to all the possibilities that needed for the next generation network.

This work also is an excellent demonstration platform for IoT sensors of how to collect and store data. It also demonstrates the efficient way of managing network of sensors code and firmware.

6. References

[1] *Orange Pi Zero*. Available Online at http://www.orangepi.org/orangepizero/index.html , accessed on 14 June 2018.

[2] *SNMP*. Available Online at http://tools.ietf.org/html/rfc1157, accessed on 14 June 2018.

[3] GitHub. *How developers work*. Available Online at https://github.com/features, accessed on 15 June 2018.

[4] Firebase. *Firebase helps mobile app teams succeed*. Available Online at https://firebase.google.com/?gclid=EAIaIQobChMItsm1o4ra2wIVyTUrCh11uQ5FEAAYASAAEgKZ2PD_BwE, accessed on 15 June 2018.

[5] K. Saikali. *Collaborative IoT Agents*. Available Online at https://www.linkedin.com/pulse/collaborative-iot-agents-karim-saikali, accessed on 16 June 2018.

[6] A. Kos, U. Sedlar, and J. Sterie. "Network monitoring applications based on IoT system." *Proceedings of the 2013 18th European Conference on Network and Optical Communications & 2013 8th Conference on*

Optical Cabling and Infrastructure (NOC-OC&I). Available Online at https://ieeexplore.ieee.org/document/6582870/, accessed on 16 June 2018.

[7] X. Zhang, M. Zhang, F. Meng, Y. Qiao, S. Xu, and S. H. Hour. "A Low-Power Wide-Area Network Information Monitoring System by Combining NB-IoT and LoRa." *IEEE Internet of things Journal*. Available Online at https://ieeexplore.ieee.org/document/8386654/, accessed on 16 June 2018.

[8] S.G. Srinivasa, N. Siddiqui, and A. Kumar. "ParaSense-A Sensor Integrated Cloud Based Internet of Things Progotype for Real Time Monitoring Applications." *IEEE Region 10 Symposium*. Available Online at https://ieeexplore.ieee.org/document/7166236/, accessed on 16 June 2018.

[9] W. T. sung, J. H. Chen, and M. H. Tsai. "Applications of wireless sensor network for monitoring system based on IoT." *IEEE International Conference on Systems, Man and Cybernetics (SMC)*. Available Online at https://ieeexplore.ieee.org/document/7844308/, accessed on 16 June 2018.