Data Consumption Pattern of MQTT Protocol for IoT Applications

Hansa Lysander Manohar[1] and T. Reuban Gnana Asir[2](✉)

[1] College of Engineering, Guindy, Anna University, Chennai, India
[2] Video Business Unit, Nokia, Chennai, India
gnana_asir.reuban@nokia.com

Abstract. In connecting the networks and people, the HTTP Protocol played a greater role and it is the most widely used protocol for data transfer in variety of applications. In IoT, we got to establish connections between "machines and things". Their communication requirements were different from current needs of Internet and associated data communications. So, HTTP protocol looks quite heavy for Internet of Things (IoT) applications, due to the overheads of HTTP. Hence, we analysed the data consumption pattern of a light weight protocol Message Queue Telemetry Transport (MQTT) supported by literature study and practical validation using Orange-Pi controlled test bed. The test bed comprises PIR Motion sensor coupled with WeMos microcontroller that sends input to the Orange-Pi Gateway over MQTT Protocol. Along with the test results, this paper summarizes the benefits of using MQTT Protocol, in IoT Applications.

Keywords: IoT · Cloud systems · MQTT · HTTP · CoAP

1 Introduction

"We stand on the brink of a technological revolution that will fundamentally alter the way we live, work, and relate to one another", says World Economic Forum, referring the fourth industrial revolution dominated by the recent technologies that include Internet of Things, Artificial Intelligence, etc. [1]. The Internet of Things is poised to dominate the internet technologies in upcoming years due to the enormous evolutions of 'things' which is set to cross beyond 28 billion users with internet connectivity by year 2020 [2], ranging from home appliances, connected cars, wearable's, industrial automation gears, etc. The installed base is poised to grow and exceed 212 billion devices, including the connected devices of 30 billion growth in next 3 years, says the industry analyst firm IDC. It predicts the growth of intelligent systems driven by data collection, data analysis and decision making across both consumer and enterprise applications [3]. Thus the data transfer across the components plays a vital role in the success of IoT applications.

When it comes to the data transfer, HTTP hits the mind due to its successful data transportation in the TCP/IP network over the years. Is the requirement of IoT applications same as the HTTP applications? No. Hence we made a study on the protocols to identify the better suited protocol for the transport layer requirements of IoT

G. P. Venkataramani et al. (Eds.): ICIIT 2017, CCIS 808, pp. 12–22, 2018.
https://doi.org/10.1007/978-981-10-7635-0_2

applications. One of the upcoming demand of IoT applications is to cope-up with the growth in the number of connections, growth in the number of devices, growth in the data transfer needs, wherever they are and whenever they are without choking up. This raises significance on the telemetry which allows things to get measured and monitored from a distance. With the improvement in telemetry technology it becomes possible to interconnect the monitoring and measuring devices from different locations.

Dependency on the smart devices and the ability to interact with other devices has raised the quest of smartness of individuals, corporate and Government organizations in every country. A woman shopping for groceries would like to get a view of what is there and what not in her kitchen. A man flying to Chennai wants to know if the flights going to that city are currently affected by weather or not. A doctor wants to know the patient's blood pressure ahead of his planned flight trip to abroad country so ensure his stability. The information that helps to take wise decisions may come from one or other forms of smart meters and equipments.

The challenge lies in the information transfer from the device to the person and to the application in a timely and effective way with increasing demand. The challenge goes to next stage based on the geographic distribution storage and computing power that shoots the cost as well which is a key factor for developing nations like India [4]. Fortunately, the advancement in the communication protocols and telemetry technologies makes it possible to receive and send the information over the internet reliably, despite network disturbance cases, little processing power of the monitoring devices, etc. using MQTT protocol [5].

This paper starts with an overview of MQTT protocol and its components and then tries to discuss the discriminating factors of MQTT Vs HTTP using the literature study. The main contribution of this paper is the MQTT protocol based application testbed and validation that is supported by the test results, observations and inference on the data consumption patterns.

2 MQTT Protocol

MQTT (Message Queue Telemetry Transport) Protocol is a light weight and an extremely simple protocol [6]. The publish/subscribe architecture of MQTT is designed with the characteristics of openness and easiness to implement, which can be scaled by single server to support up to thousands of clients accessible from remote. These characteristics of MQTT makes it ideal for usage in constrained environments where there is high latency or low network bandwidth and devices from remote sites that could have limited memory and processing capabilities [7].

The benefits of MQTT protocol includes the following:

(a) It delivers the data relevant to any intelligent and decision-making asset that can utilize it.
(b) It extends the connectivity range exceeding enterprise boundaries to reach to smart devices.
(c) It provides the optimized connectivity options for remote devices and sensors.
(d) It enables enormous scalability of management and deployment to IoT solutions.

2.1 MQTT Highlights

MQTT Protocol claims that, it minimizes the device resource requirements and network bandwidth with attempts to deliver with reliability. This characteristic is the validation goal that is explained in the later parts of this paper.

This approach of minimal resource requirement and reduced network bandwidth makes the MQTT protocol well-positioned for connecting machine to machine (M2M) communications, which is a critical aspect of the IoT.

The other key highlights include [5]:

- Open and royalty-free.
  - MQTT is easy to adopt, open to make and fit for variety of platforms, devices, and operating systems that are used at the network edge.
- Messaging model.
  - The publish/subscribe messaging model facilitates one-to-many distribution. Sender devices or applications need not know anything about the receiving device or applications, not even its address.
- Ideal for constrained networks.
  - MQTT message headers are retained as small as possible and ideal for fragile connections, low bandwidth, data limits, high latency networks. The fixed header is only two bytes, that too on demand, push-style message distribution keeping the network utilization low.
- Multiple service levels.
  - It gives the flexibility in handling various types of messages. For example, developers can design that the messages will be delivered exactly once, at least once, or at most once.
- Design.
  - Its designed to support remote devices with low processing power and minimal memory.
- Ease of use.
  - Usage and implementation is quite easy with simple set of command messages. Various applications of MQTT will be accomplished using CONNECT, DISCONNECT, SUBSCRIBE, UNSUBSCRIBE and PUBLISH methods.
- Built-in support for contact loss.
  - If the connection with client connection breaks abnormally, the information is sent to server facilitating the message either to get preserved for later delivery or to re-send.

3 Findings Over HTTP

In this section, let us list down the key factors of comparison between HTTP Protocol and MQTT Protocol for IoT applications based on the research done in this field by several experts in the past and the next section shares the observation based on validation results.

3.1 Comparison of MQTT vs HTTP

The below Table 1 gives a quick view on the comparison between MQTT and HTTP.

Table 1. Quick view of MQTT vs HTTP

	MQTT	HTTP
Design orientation	Data centric	Document centric
Pattern	Publish/subscribe	Request/response
Complexity	Simple	More complex
Message size	Small, with a compact binary header just two bytes in size	Larger, partly because status detail is text-based
Service levels	Three quality of service settings	All messages get the same level of service
Extra libraries	Libraries for C (30 KB) and Java (100 KB)	Depends on the application (JSON, XML), but typically not small
Data distribution	Supports 1 to zero, 1 to 1, and 1 to *n*	1 to 1 only

- **High Power consumption by HTTP [8]:**
  - In the dynamic data communication scenarios, HTTP is observed to be consuming more power. In the tests done by Hantrakul K et al., the HTTP protocol consumes 10 times higher power than MQTT protocol. They have witnessed MQTT sends 10 times more messages than HTTP in 1 h of operation.
  - Tests done by Upadhyay et al. [9] reveals, power consumption of MQTT Protocol is way lower and 30% faster performance than CoAP [10].
- **High Protocol overheads in HTTP:**
  - IoT applications requires large number of information exchange with tiny packets. Hence the payload is quite less, whereas the overhead caused to transfer the payload is quite high.
  - From the below Figs. 1 and 2 we see the elimination of CONNECT/CONNACK flow for MQTT cases, that reduces the overhead and latency, when compared to HTTP, leading faster data transfer as well [10].

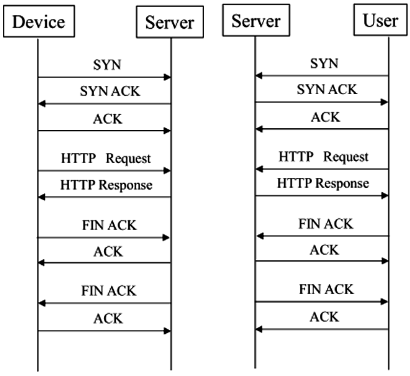

Fig. 1. Communication sequences of HTTP use case.

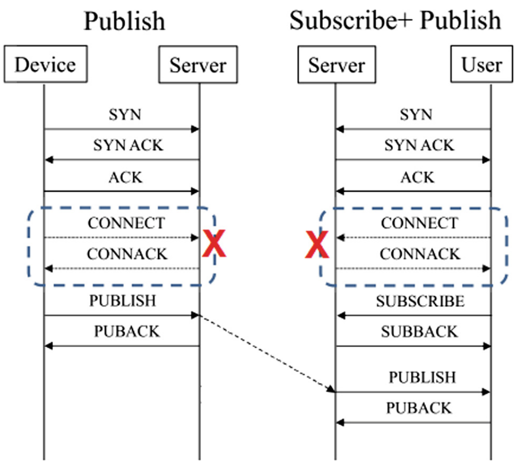

Fig. 2. Communication sequences of MQTT use case where CONNECT/CONNACK is eliminated

- **High Bandwidth consumption in HTTP:**
  - From the research done by Yokotani and Sasaki [11] on the comparison of bandwidth usage between HTTP and MQTT on 2 different cases, with payload and without payload (where only topics exist, that is used to decide on the MQTT broker, which client receive which message).
  - For MQTT topics cases, where zero payload exists and only the transmission bytes exists reveals, HTTP consumes 300% higher bandwidth as in Fig. 3.

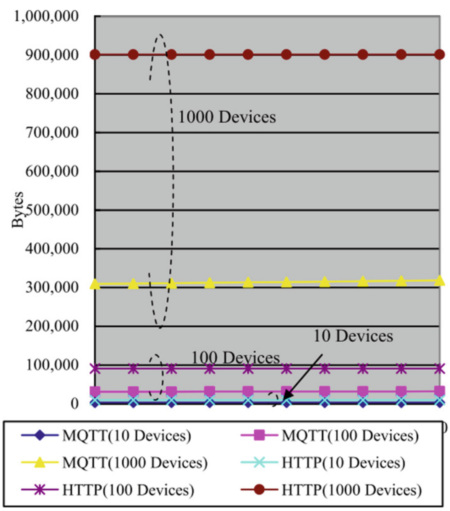

Fig. 3. Characteristics with zero payload

– For MQTT message sharing cases, where pay load and transmission bytes exists, HTTP consumes 250% higher bandwidth as in Fig. 4.

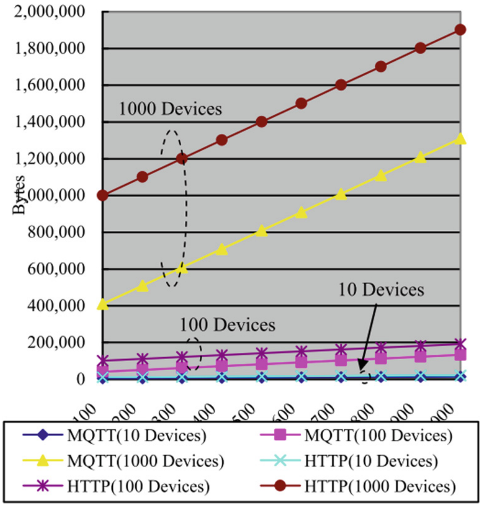

Fig. 4. Characteristics with payload and overhead.

When these studies reveals that MQTT is better choice that HTTP for IOT applications, we wished to get into it to detail to understand any additional behavior of MQTT and got an interesting observation, that will be explained in next chapter.

4 Experiment and Test Results

4.1 Test Environment

The test environment mainly comprises of following components as in Fig. 5.

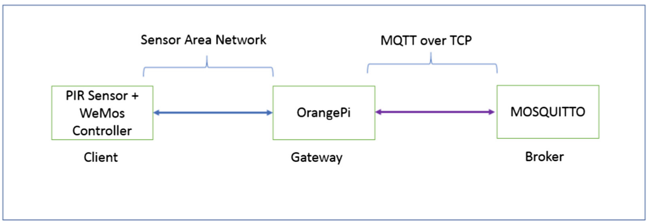

Fig. 5. Test bed overview

- PIR Sensor HC-SR501
- WeMos D1 mini ESP8266
- Orange Pi Zero Processor as MQTT Gateway
- Mosquitto MQTT Broker v3.1

PIR Sensor HC-SR 501 + WeMoS ESP8266 Controller

The short distance communication is realized by the Wireless Sensor Networks (WSN) among the objects nearby. In this experiment, we've picked PIR motion sensor that contains low-cost Wi-Fi chip with full TCP/IP stack. The PIR motion sensor detects the presence of anyone coming closer or moving away and will send the signal. However, it's difficult to connect each other with the mobile communication networks, the Internet and WSN because there is not much standardization exists with respect to communication protocols and sensing technologies. The other restriction is from the data transmission from WSN in long distance due to the limitation of WSN's transmission protocols. Therefore, we house WeMos D1 mini controller as in Fig. 6, whose aim is to balance the heterogeneity between mobile communication, sensor network and Internet that strengthens the management of the terminal nodes, WSN and bridge [12].

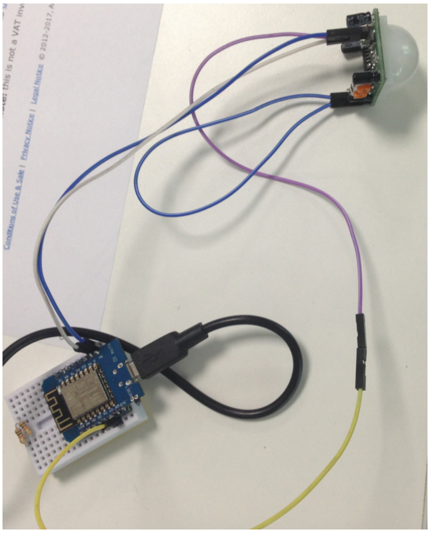

Fig. 6. PIR sensor and WeMos controller

OrangePi Zero

Its an open source single board computer that can run on Operating Systems, that includes Android, Ubunto, Debian, Armbian. This acts as IoT Gateway that converts the input from sensory network and passes to MQTT broker over TCP. In our tests, we tried to hook into these conversations using TCP dump and observed the pattern of MQTT communications as in Fig. 7.

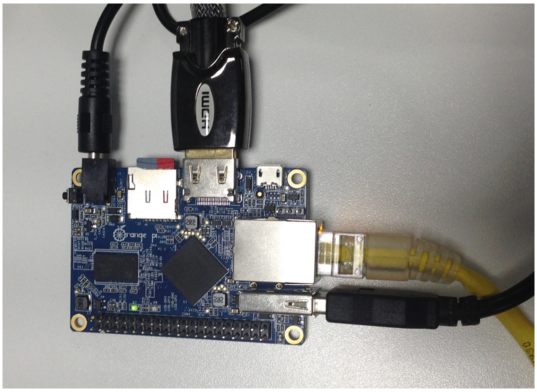

Fig. 7. MQTT Gateway with Orange-Pi

Mosquitto Broker

This is an open source message broker that implements MQTT protocol versions 3.1 and 3.1.1. It carries out the Publish/Subscribe model that makes it suitable for messaging with low power sensors, mobile devices, embedded computers and Arduino micro controllers. To write the control logic, we used the Node-RED as in Fig. 8, that facilitates the flow based programming method with ease of use.

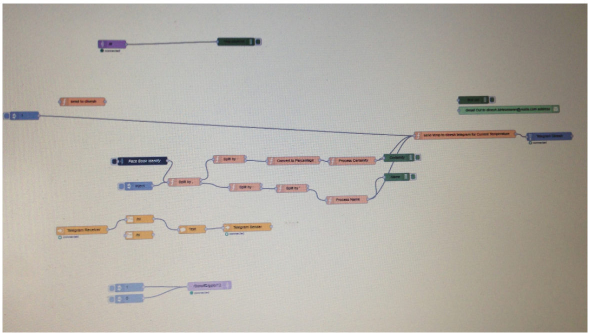

Fig. 8. Node red flow using Mosquitto MQTT Broker

4.2 Test Results

From the tests performed we tried to collect the TCP dump for the conversations between IoT Gateway and MQTT Broker as in Fig. 9.

The interesting observations are:

1. The sensor could sense either an object coming near or moving away and it sends the signal, which we see in the Fig. 10 as ON and OFF. Whenever the sensor senses a signal, it transmits to MQTT broker with TCP message of length 28 bytes.
2. When there is no signal change read by sensor, the TCP message with 0 bytes is transmitted, which is presumably the CONNACK.

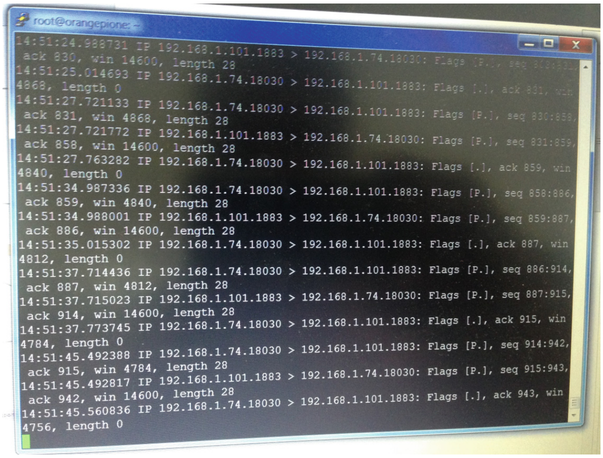

Fig. 9. TCP dump collected from MQTT Gateway

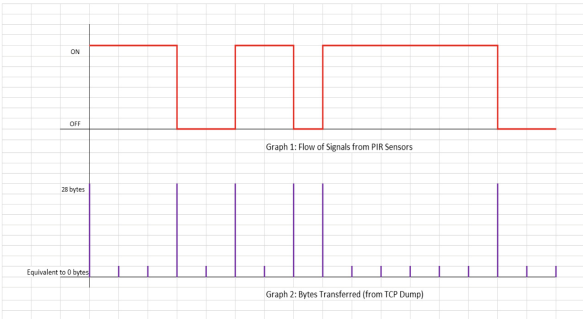

Fig. 10. Graphical representation of TCP message length

This pattern is graphically represented in Fig. 10 and we realize MQTT consumes 28 bytes data, only when there is signal ON/OFF to be transmitted. On all other cases, it remains very minimal with byte length as zero. This pattern is another feather on MQTT's effective data consumption trend.

5 Future Work

In the coming days, we wish to extend the test bed integrated with actuators like SONOFF switches and observe the data consumption pattern between the MQTT broker and the actuators. With additional efforts, this test bed will get integrated with multiple types of IoT applications, other protocols used in IoT applications like CoAP and make a study on the data consumption patterns in both simple cases and under traffic situations.

6 Conclusion

With the findings of the MQTT Data consumption pattern, it is evident that MQTT Protocol is essential for effective unitization of network bandwidth, to reach out to devices at remote locations, low power enabled devices. Usage of HTTP as the transport layer for IoT Applications will help for initial stages only. With the increasing trend of number of IoT applications it is a value add to switch to MQTT to match various usecases that pops-up like IaaS [13]. The REST API support of MQTT to Push or Pull the data as and when required, is getting leveraged by industry champions in Cloud Platforms include Aercloud, IBM IoT [14].

Abbreviations and Acronyms

CoAP - Constrained Application Protocol
HTTP - Hyper Text Transfer Protocol
IaaS - IoT As A Service
IoT - Internet of Things
MQTT - Message Queue Telemetry Transport
TCP - Transmission Control Protocol
WSN - Wireless Sensory Networks

Acknowledgment. We would like to thank Nokia, College of Engineering Guindy, for giving us such an opportunity to carry out this research work and for providing us the requisite resources and infrastructure for carrying out the research. Special thanks to Mr. Dinesh Birlasekaran, Mr. Wilson Anandaraj from Nokia for being a mentor and continuous inspiration towards the research on IoT.

References

1. https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/
2. The Internet of Things: Making sense of the next mega-trend. Goldman Sachs Global Investment Research, IoT Primer, September 2014
3. IDC, Worldwide Internet of Things (IoT) 2013–2020 Forecast: Billions of Things, Trillions of Dollars, Doc # 243661, October 2013
4. Reuban Gnana Asir, T., Anandaraj, W., Naga Sivaranjani, K.: Internet of things and India's readiness. In: International Conference on Computing Paradigms (ICCP2015), pp. 274–279, July 2015
5. Lampkin, V., Leong, W.T., et al.: Building smarter planet solutions with MQTT and IBM WebSphere MQ telemetry, September 2012
6. https://en.wikipedia.org/wiki/MQTT
7. Barata, D., Louzada, G., Carreiro, A., Damasceno, A.: System of acquisition, transmission, storage and visualization of pulse oximeter and ECG data using android and MQTT. Procedia Technol. J. **9**, 1265–1272 (2013)

8. Hantrakul, K., Sitti, S., Tantitharanukul, N.: Parking lot guidance software based on MQTT protocol. In: 2017 International Conference on Digital Arts, Media and Technology (ICDAMT) (2017)
9. Upadhyay, Y., Borole, A., Dileepan, D.: MQTT based secured home automation system. In: 2016 IEEE Symposium on Colossal Data Analysis and Networking, CDAN 2016 (2016)
10. Amaran, M.H., Noh, N.A.M., Rohmad, M.S., Hashim, H.: A comparison of lightweight communication protocols in robotic applications. Procedia Comput. Sci. **76**, 400–405 (2015)
11. Yokotani, T., Sasaki, Y.: Comparison with HTTP and MQTT on required network resources for IoT. In: International Conference on Control, Electronics, Renewable Energy, and Communications 2016, Conference Proceedings, CCEREC 2016 (2016)
12. Zhu, Q., Wang, R., et al.: IOT gateway: bridging wireless sensor networks into internet of things. In: IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, pp. 347–352 (2010)
13. Reuban Gnana Asir, T., Manohar, H.L., Anandaraj, W., Naga Sivaranjani, K.: "IoT as a Service" Internet of things and India's Readiness. In: International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS 2016), pp. 1093–1096 (2016)
14. Ray, P.P.: A survey of IoT cloud platforms. Future Comput. Inf. J. **1**(1–2), 35–46 (2017)

http://www.springer.com/978-981-10-7634-3

Smart Secure Systems – IoT and Analytics Perspective
Second International Conference on Intelligent
Information Technologies. ICIIT 2017, Chennai, India,
December 20-22, 2017, Proceedings
Venkataramani, G.P.; Sankaranarayanan, K.;
Mukherjee, S.; Arputharaj, K.; Sankara Narayanan, S.
(Eds.)
2018, XV, 337 p. 161 illus., Softcover
ISBN: 978-981-10-7634-3