Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

MQTT-BASED IOTS NETWORK PERFORMANCE ANALYSIS

Shahadan Saad[1]
Adib Farhan Azzemi[2]
Muhammad Syahid Farhan Mohd Jeman[3]
Ahmad Firdaus Ahmad Fadzil[4]
Albin Lemuel Kushan[5]
Azmi Aminordin[6]

[1]Universiti Teknologi MARA. Email: shahadansaad@gmail.com
[2]Universiti Teknologi MARA. Email: adib.farhan97@gmail.com
[3]Universiti Teknologi MARA. Email: fcsyahid@gmail.com
[4]Universiti Teknologi MARA. Email: firdausfadzil@uitm.edu.my
[5]Universiti Teknologi MARA. Email: albin1841@uitm.edu.my
[6]Universiti Teknologi MARA. Email: azmi1107@uitm.edu.my

Article history
Received date : 15-4-2021
Revised date : 16-4-2021
Accepted date : 29-4-2021
Published date : 31-5-2021

To cite this document:
Saad, S., Azzemi, A. F., Mohd Jeman, M. S., Ahmad Fadzil, A. F., Kushan, A. L., & Aminordin, A. (2021). MQTT-Based IoTs Network Performance Analysis *Journal of Islamic, Social, Economics and Development (JISED)*, 6 (37), 37 – 49.

Abstract: *Recent years has seen the rise of popularity of IoT. IoT has become a hot topic discussed in many forums, blogs, and talks. Due to that, big companies, remote area industries, SOHO, and home users are thinking about implementing IoT into their daily operation. However, there are several factors that need to be analyzed before IoT can be implemented. This paper focused on the performance of MQTT-based IoT ecosystem. Packet length and response time are two of the metrics that going to be analyzed in this project using Wireshark. IoT are intended to replace a more traditional and conventional ways of performing tasks. Thus, if the performance of the IoT does not reflect any improvement over the current system, there is no point in implementing IoT into daily operation. From the experiment conducted, it is indicated that MQTT is a better protocol to be used in IoT ecosystem as compared to HTTP. The result gathered proved that MQTT are a better choice to be used by low-power devices. Result obtained in this paper can be useful to companies, industries, SOHO, and home users.*

Keywords: MQTT, Network Performance Analysis, IoT.

Introduction

Nowadays, Internet of Things or IoT popularity has rose due to the rapid development of new low-cost smart equipment and devices. These devices communicate using protocol that define the way data is formatted and transmitted. However, IoT differs from Web as Web uses one standardized protocol which is HTTP. According to Naik (2017) there are several protocols can be used by machines or IoT to communicate with each other. CoAP, XMPP, AMQP and MQTT are common messaging protocol designed to be used in IoT devices. MQTT is one of the

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

messaging protocols designed to be used by machines and IoT devices to communicate with each other (Nuratch, 2019). This project focus on analyzing the performance of MQTT as IoT messaging protocol for IoT ecosystem. The messaging protocol used may affect the communication performance, thus choosing the correct protocol that suits the condition would be crucial.

To deploy IoT into day-to-day business operation, system administrator need to choose the correct messaging protocol. Naik (2017) in his paper has stated that the performance of M2M communication relies on the messaging protocol explicitly designed in the IoT application. Business owner nowadays are looking forward into implementing IoT into their daily operation. Those that are in remote areas where electricity and bandwidth are limited will find this to be troublesome. Toldinas et al. (2019), stated that power consumption of IoT devices in constrained environment is IoT critical point.

This project will provide system administrator with option on choosing the correct protocol before deploying IoT widely into constrained environment. Specification on paper shows that MQTT is a better protocol for the use by IoT ecosystem in constrained environment. This paper focused on analyzing the performance of MQTT in IoT ecosystem. There are several aspects that is analyzed in this paper which are response time and packet length. The result obtained from the experiment conduct will determine, how much better is MQTT as compared to HTTP. It is essential to choose the correct messaging protocol because it can affect the efficiency of business daily operation.

Literature review

Internet of Things

Internet of Things or IoT, refers to a network of interconnected smart devices, sensors, and actuators that can collect, analyses and store data to be used for decision making process. According to Sharma & Gondhi (2018) IoT may be represented as a collection of smart devices linked together over a network to conduct decision making. IoT environment is a decentralized environment as each of the device can operate independently without relying on each other. According to Sethi & Sarangi (2017) IoT is not a standalone system, rather, it is a formation of different technologies operating in unison together. Through the use of smarter, cheaper and affordable devices, industry can save cost and time by implementing IoT as compared to traditional machine without sacrificing daily operation efficiency. According to Muquit et al. (2019) compared to conventional devices, IoT devices bring more advantages by providing a much accurate and real-time data.

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

IoT System

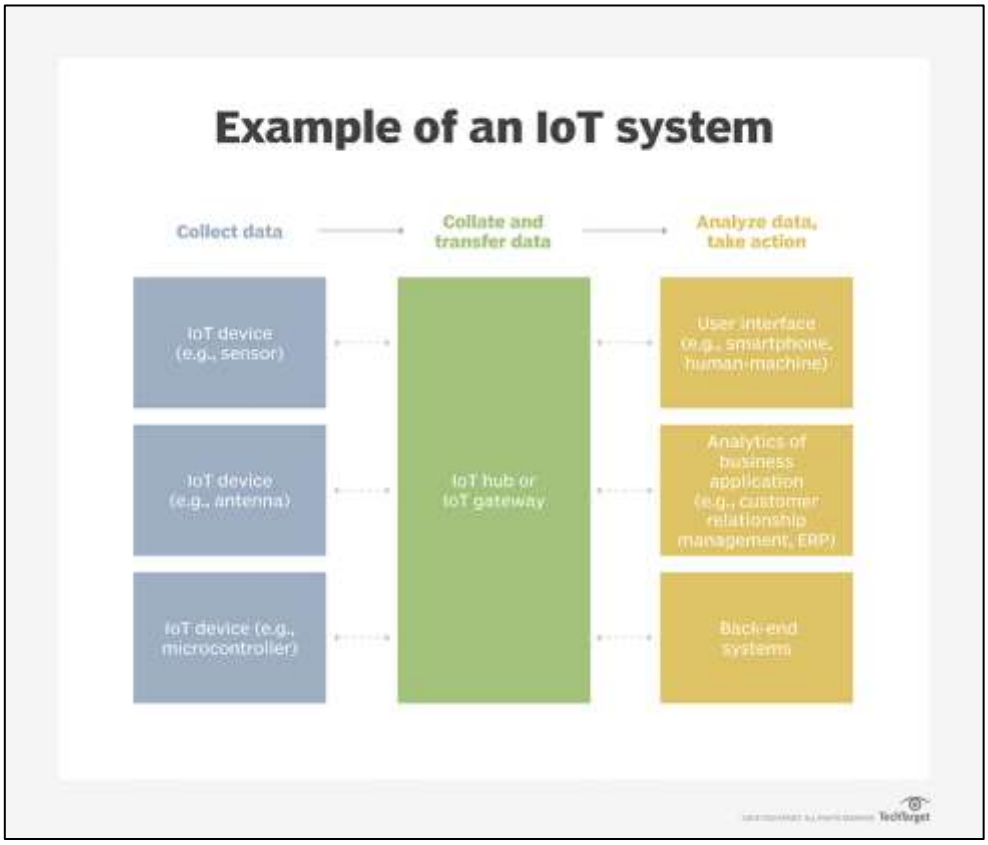


Figure 1 IoT System

(*Source:* https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT)

IoT system consists of hardware such as sensors, network connectivity, software responsible to analyze the data, and user interface to present data in readable format as shown in Figure 1. Sensors such as DHT22 is responsible for gathering and transmitting data that they obtain from their environment to the IoT gateway. These smart devices and sensors connect and share their data to the IoT gateway. IoT gateway such as Raspberry Pi is responsible to send data collected by the sensor to IoT platform either wired or wirelessly. IoT gateway is a device that provide the ability to bring cloud connection services to the IoT end devices (Evanczuk, 2018). Data sent by IoT gateway is stored and analyzed by IoT platform such as ThingsBoard. ThingsBoard can be hosted either using local server or in virtual private server with cloud capabilities. According to Kukkamäki et al. (2018), IoT platform assists in monitoring, managing, reacting to devices communication, and controlling remote devices. Data that has been analyzed are presented in a readable format such as graph or charts. These IoT devices communicate using messaging protocol. There are several messaging protocols that can be used in IoT environment, such as Message Queuing Telemetry Transport (MQTT), Constrained Application Protocol (CoAP), and Hypertext Transfer Protocol (HTTP).

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

Message Queuing Telemetry Transport

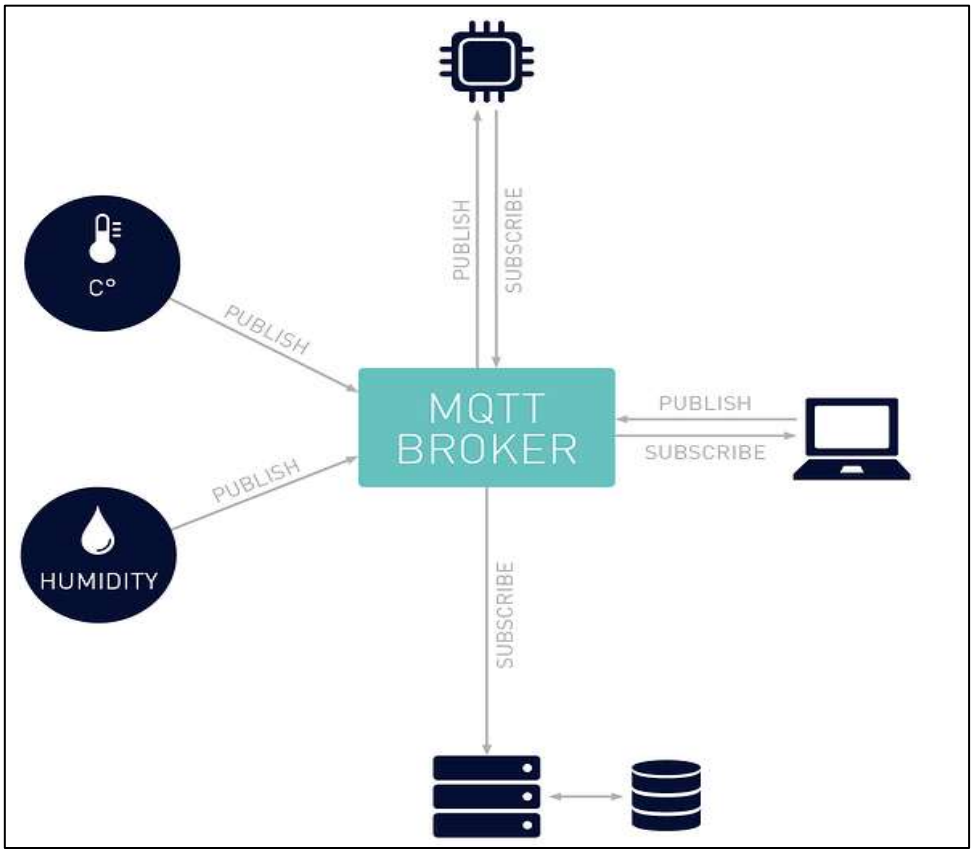


Figure 2 MQTT Architecture

(*Source:* https://blog.paessler.com/why-mqtt-is-everywhere-and-the-security-issues-it-faces)

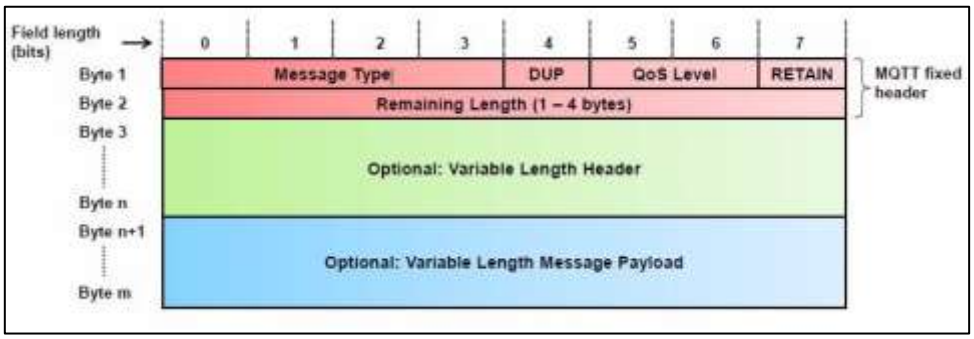


Figure 3 MQTT message packet format

(*Source:* https://www.rfwireless-world.com/Tutorials/MQTT-tutorial.html)

Message Queuing Telemetry Transport is a lightweight publish/subscribe messaging protocol use for IoT and machine-to-machine communications. According to MQTT, (n.d.) MQTT is the protocol that met the standard of OASIS for the use of IoT. It is designed for devices that has limited bandwidth and resources. Tseng et al. (2019) stated that MQTT is suitable to be used in a resource and network constrained environment. MQTT is an omnidirectional

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

messaging protocol which allows MQTT client to subscribe and publish at the same time as shown in Figure 2. The delivery of these messages is handled by MQTT broker, and the destination is identified by topic. Due to it relies on the topic to determine which publisher will get the message, the spelling and topic level is very crucial to guaranty message delivery. MQTT topics are dynamic, as it will remain exist if there are subscribers, or a retained broker, or last messages stored for the topic. Only MQTT broker require a fixed IP addressing, and client does not require IP addressing or port forwarding. Conventional way of using IP addressing for interaction between device would be very difficult with the large number of sensors and actuators. MQTT also works well in a dynamic IP addressing environment due to it being able to reconnects automatically when address changes. This protocol is suitable to be used in an IoT ecosystem that comprise of many low-power devices which pass many small data. Since MQTT is a lightweight protocol, it does not cause protocol overhead like HTTP. Figure 3 shows the MQTT message packet format, which is 2 bytes fixed header size, a variable length header for optional header information, and a variable length message payload up to 256MB.

Hypertext Transfer Protocol

Hypertext Transfer Protocol is the fundamental for transferring files such as text, image, sound, and video on the World Wide Web. It works based on the request-response protocol in the client-server model. HTTP is a request-response based protocol in the client-server model. HTTP contain references to other files which when selected, will evoke additional request for the transfer. According to (Wukkadada et.al, 2018) communication through this protocol will cause severe bandwidth problems for IoT. HTTP does provide reliable communication because it operates over TCP/IP. However, every request requires new connection, and after every response, connections established are released. According to (Yokotani & Sasaki, 2017) applying http in IoT causes huge number of tiny data is transferred and this lead to protocol overhead and performance degradation. After many establishments and released of complete communication, IoT devices experience high consumption of network resources and serious overhead.

MQTT and HTTP Comparison

Table 1 shows the differences in the specification of both protocols, and this causes both protocols to work differently from each other and this will affect the performance of device that use these protocols.

Table 1 MQTT and HTTP Comparison

Protocol / Characteristics	MQTT	HTTP
Architecture	**Client and Broker**	**Client and Server**
Abstraction	**Publish and Subscribe**	**Request and Response**
Header Size	**2 Bytes**	**Undefined**
Message Size	**Up to 256MB**	**Vary, depending on technology used**
Transport Protocol	**TCP**	**TCP**
Default Port	**1883/8883**	**80/443**
Encoding	**Binary**	**Text**

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

Methodology

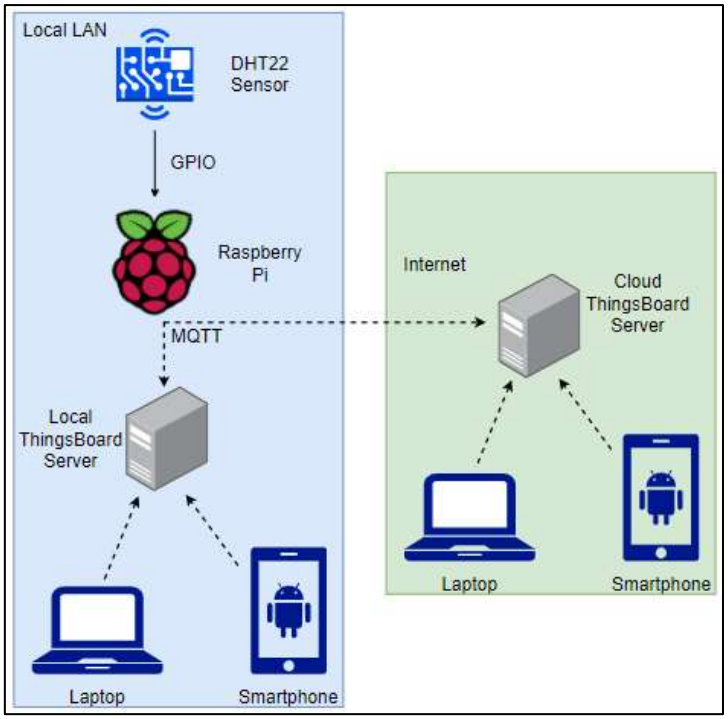


Figure 4 System Architecture

Figure 4 shows the system architecture of MQTT based weather monitoring station. The station consists of 2 devices, which is a DHT22 sensor, and a Raspberry Pi 3 model B+. The DHT22 sensor is connected by wire to Raspberry Pi through the General-Purpose Input Output or the GPIO. Data collected from the sensor is sent to the Raspberry Pi through the GPIO. Raspberry Pi publish the data using MQTT to both cloud and local ThingsBoard IoT platform that act as a MQTT broker. The purpose of local server it for backup data logging for when cloud server is unreachable due to maintenance or lost connection. A python script is used to collect data from the sensor and publish the data using MQTT to both the broker. To make the ease of accessing the server from the internet, instead of finding the server through the IP address, a Domain Name Service is used.

Publish and Subscribe to MQTT Topics

Using Mosquitto MQTT Broker

There are several MQTT broker that can be used to handle MQTT messages and in this project, Mosquitto MQTT broker is used alongside ThingsBoard IoT platform to test and show how data is sent or publish from the publisher and how data is receive or subscribe by the subscriber. Mosquitto is chosen because of it is a lightweight and open-source MQTT broker. One way to publish the message is by using publish function in Mosquitto MQTT through the Windows command line tools. Figure 5 shows the process of sending or publishing message to MQTT broker from MQTT client by using Mosquitto MQTT, while Figure 6 shows the process of subscribing message from MQTT broker by using Mosquitto MQTT.

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

```
C:\Program Files\mosquitto>mosquitto_pub -d -t /home/user -m "hello"
Client mosq-mSTfJKcK9LfGWktFFD sending CONNECT
Client mosq-mSTfJKcK9LfGWktFFD received CONNACK (0)
Client mosq-mSTfJKcK9LfGWktFFD sending PUBLISH (d0, q0, r0, m1, '/home/user', ... (5 bytes))
Client mosq-mSTfJKcK9LfGWktFFD sending DISCONNECT

C:\Program Files\mosquitto>mosquitto_pub -d -t /home/user -m "successfull communication"
Client mosq-I8Liy5IUe3P5PGJlnw sending CONNECT
Client mosq-I8Liy5IUe3P5PGJlnw received CONNACK (0)
Client mosq-I8Liy5IUe3P5PGJlnw sending PUBLISH (d0, q0, r0, m1, '/home/user', ... (25 bytes))
Client mosq-I8Liy5IUe3P5PGJlnw sending DISCONNECT
```

Figure 5 Publish using Mosquitto MQTT broker

```
C:\Program Files\mosquitto>mosquitto_sub -d -t /home/user
Client mosq-MkLtxSedPtYdT8b4Iw sending CONNECT
Client mosq-MkLtxSedPtYdT8b4Iw received CONNACK (0)
Client mosq-MkLtxSedPtYdT8b4Iw sending SUBSCRIBE (Mid: 1, Topic: /home/user, QoS: 0, Options: 0x00)
Client mosq-MkLtxSedPtYdT8b4Iw received SUBACK
Subscribed (mid: 1): 0
Client mosq-MkLtxSedPtYdT8b4Iw received PUBLISH (d0, q0, r0, m0, '/home/user', ... (5 bytes))
hello
Client mosq-MkLtxSedPtYdT8b4Iw sending PINGREQ
Client mosq-MkLtxSedPtYdT8b4Iw received PINGRESP
Client mosq-MkLtxSedPtYdT8b4Iw received PUBLISH (d0, q0, r0, m0, '/home/user', ... (25 bytes))
successfull communication
```

Figure 6 Subscribe using Mosquitto MQTT broker

Using Python Script and Paho-MQTT Library

Python also support MQTT using Paho-MQTT library. Publishing MQTT messages to ThingsBoard IoT platform can be done using several methods. MQTT message can be sent using Mosquitto MQTT broker, as well as using Python script. Using python script, requires the use of Paho-MQTT library that consists of all the prerequisites for connecting to MQTT broker, sending the message, and disconnecting. A python list is used to store the details for both cloud and local ThingsBoard platform and device credential to be used for authentication. Figure 7 shows the syntax for connecting to MQTT broker using python. The required information is stored in a python list and called during connection process. Figure 8 shows the syntax to publish sensor reading data to MQTT broker.

```
client1 = mqtt.Client()
client1.username_pw_set(token1)
client1.connect(host1, 1883, 60)

client2 = mqtt.Client()
client2.username_pw_set(token2)
client2.connect(host2, 1883, 60)

client1.loop_start()
client2.loop_start()
```

Figure 7 Connection to MQTT broker

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

```
client1.publish('v1/devices/me/telemetry', json.dumps(sensor_data), 1)
client2.publish('v1/devices/me/telemetry', json.dumps(sensor_data), 1)
```

Figure 8 Publishing the message

Results and Findings

Message Delivery Testing

The delivery of the message is to be examined to ensure that MQTT and HTTP is used, and message is successfully delivered to the correct broker. Sending message to ThingsBoard IoT platform requires the use of authentication token which can be retrieved in the device panel. The token is sent alongside the sensor data in JSON format to ThingsBoard IoT platform. The sensor data should be displayed in the device telemetry data section. Once the sensor data has been confirmed to be delivered, the response time and total packet length is measured.

```
4530 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=3) [v1/devices/me/telemetry]
4532 0.043  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=3)
4675 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=4) [v1/devices/me/telemetry]
4679 0.043  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=4)
4688 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=5) [v1/devices/me/telemetry]
4689 0.043  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=5)
4697 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=6) [v1/devices/me/telemetry]
4698 0.043  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=6)
4704 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=7) [v1/devices/me/telemetry]
4705 0.043  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=7)
5049 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=8) [v1/devices/me/telemetry]
5085 0.043  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=8)
5481 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=9) [v1/devices/me/telemetry]
5572 0.043  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=9)
6026 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=10) [v1/devices/me/telemetry]
6027 0.046  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=10)
6085 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=11) [v1/devices/me/telemetry]
6086 0.044  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=11)
6125 *REF*  192.168.1.105    188.166.223.199  MQTT   130 Publish Message (id=12) [v1/devices/me/telemetry]
6126 0.043  188.166.223.199  192.168.1.105    MQTT    70 Publish Ack (id=12)
```

Figure 9 Send Message using MQTT

```
 942 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
 954 0.097  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1209 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1214 0.090  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1230 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1235 0.094  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1287 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1292 0.070  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1410 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1414 0.081  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1445 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1448 0.088  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1479 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1482 0.076  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1499 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1505 0.089  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1649 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1655 0.084  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
1715 *REF*  192.168.1.105    188.166.223.199  HTTP   105 POST /api/v1/zbO9t60DgA95JvxpqzXh/telemetry HTTP/1.1
1721 0.096  188.166.223.199  192.168.1.105    HTTP   435 HTTP/1.1 200
```

Figure 10 Send Message using HTTP

Figure 9 and 10 shows the message sent to ThingsBoard IoT platform using both MQTT and HTTP. The packet is inspected using Wireshark to ensure the protocol used, the response time, and the message packet length.

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

Performance Analysis of MQTT

The performance of MQTT and HTTP are evaluated based on two criteria, the packet length, and the response time. Both protocols will send the same message to ensure a fixed and controlled condition. Response time is the time taken between a request and response between client and server. Usually this can be measured by the time a request is sent, and the time a respond is received. Packet length is the total length after the original data is encapsulated and transmitted across a link. Different protocols will result in different information encapsulated together with the original data. This will result in different in the length of the packet transmitted. In this project, the time taken between the message is sent or publish and the time an acknowledgement message is received is recorded to measure the protocol response time. Both the HTTP and MQTT response time as well as packet length is recorded and analyzed.

The packet length using both MQTT and HTTP is gathered using Wireshark and recorded. Message is sent using Mosquitto and CUrl on Windows command line tools. Table 2 shows the result when Raspberry Pi sent the data once to the ThingsBoard server and the length of the packet is recorded. The length of HTTP packet for sending message, and the whole process from connecting, sending and acknowledging, is bigger than MQTT packet. Thus, the time taken to complete the process for HTTP is longer than that of MQTT.

Table 2 Packet Length

Protocol	Packet Length (message)	Whole Process Packet Length	Time Taken for Whole Process
MQTT	**114 bytes**	**343bytes**	**90ms**
HTTP	**260 bytes**	**635bytes**	**108ms**

Second test is done by sending 10 messages containing readings data from sensor. The test is done by executing a python script in Raspberry Pi that sent 10 messages for both protocols. There are some differences in the length of the packet for both MQTT and HTTP message if sent from terminal and from python script. The packet length for both MQTT and HTTP differs due to several reason. As for HTTP, it relies on the technology used to send the message, thus the differences in the packet length. AS for MQTT, using Mosquitto to send single message, it requires to establish connection and to request for disconnection on every message. However, using Python script, connection and disconnection request are only needed to be issued once. Thus, resulting in a more simplified and lightweight process. Table 3 shows the packet size for one message and packet size for 10 messages, while the average response time for MQTT and HTTP is recorded and analyzed as shown in Table 4.

Table 3 Total Packet Length for 10 Messages

Protocol	MQTT	HTTP
Total Packet Length	**200 Bytes**	**540 Bytes**
Total Packet Length For 10 Messages	**2 Kilobytes**	**5.4 Kilobytes**

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

Table 4 Average Time Taken Between Transmission and Acknowledgement

Protocol / Messages	MQTT	HTTP
1st message	**43ms**	**97ms**
2nd message	**43ms**	**90ms**
3rd message	**43ms**	**94ms**
4th message	**43ms**	**70ms**
5th message	**43ms**	**81ms**
6th message	**43ms**	**88ms**
7th message	**43ms**	**76ms**
8th message	**46ms**	**89ms**
9th message	**44ms**	**84ms**
10th message	**43ms**	**96ms**
Average response time	**43.4ms**	**86.5ms**

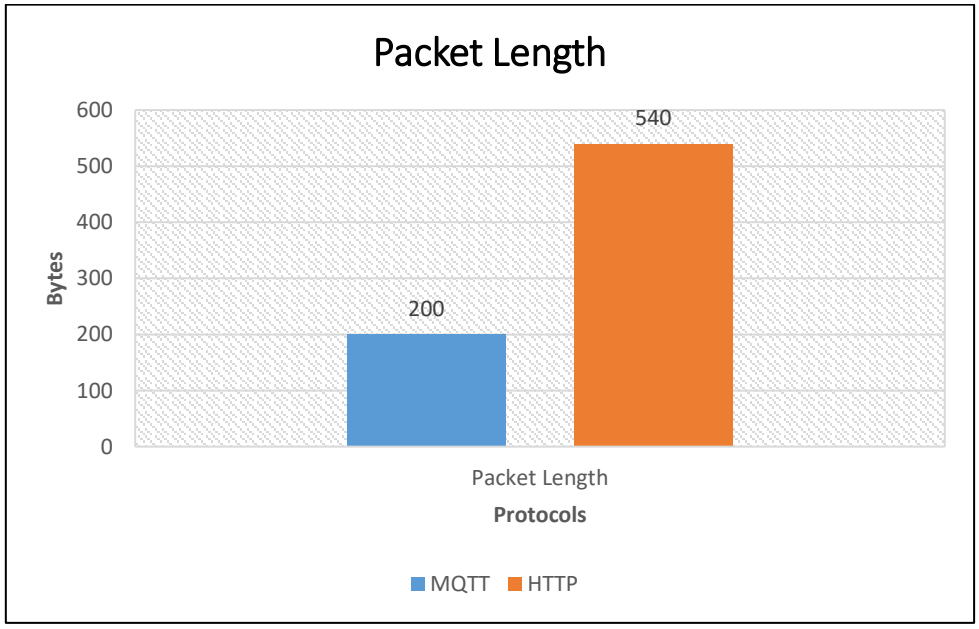


Figure 11 Bar Graph for Message Packet Length

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

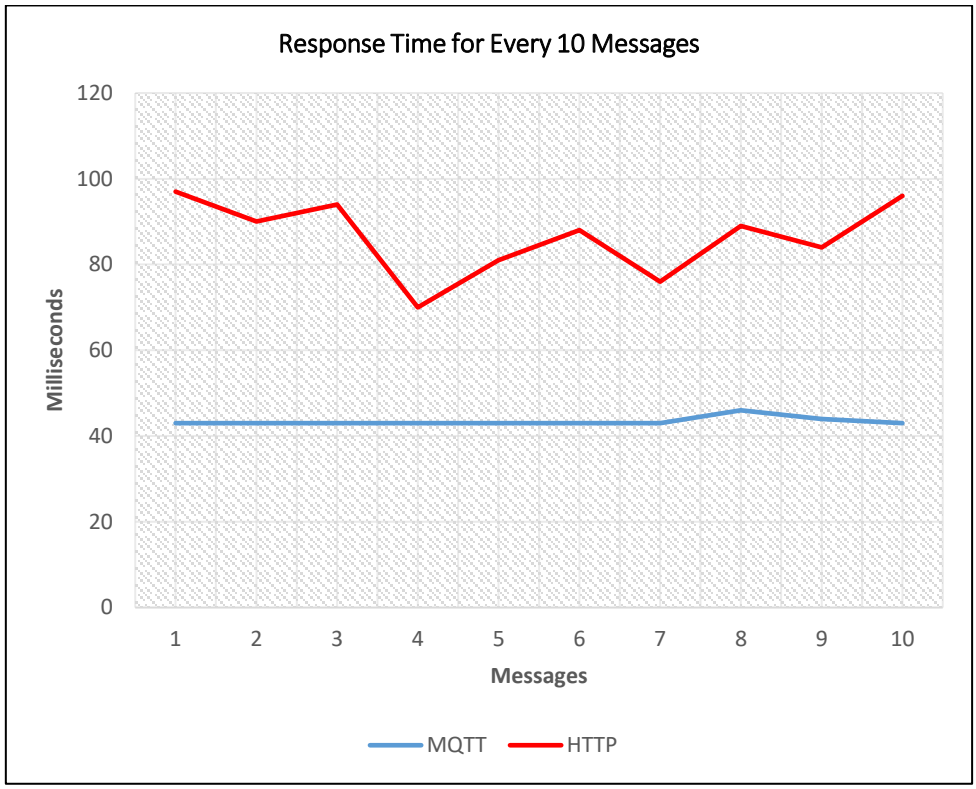


Figure 12 Line Graph of Response Time for Every 10 Messages

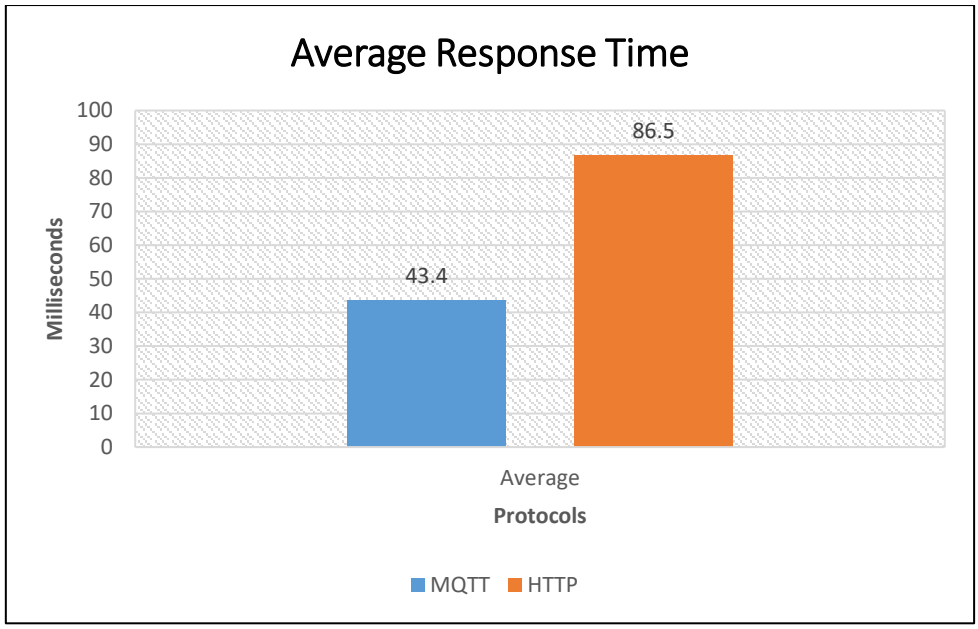


Figure 13 Bar Graph for Average Time Taken Between Transmission and Acknowledgement

Based on figure 11, the total packet length for 10 MQTT messages and acknowledgment is smaller than the total packet length for 10 HTTP messages and acknowledgement by 91.9%. Figure 12 shows that for every 10 messages sent using MQTT, the response time is lower as compared to messages sent using HTTP. Figure 13 shows that the average response time would be 43.4ms for MQTT and 86.5ms for HTTP. Thus, based on this test the response time of

Volume:6 Issues: 37 [June, 2021] pp. 37 - 49]
Journal of Islamic, Social, Economics and Development (JISED)
eISSN: 0128-1755
Journal website: www.jised.com

MQTT is faster than HTTP by 66.3587%, and it can be said that MQTT is lighter and faster than HTTP which make it suitable to be used by low-power devices in IoT ecosystem.

Conclusion

It can be concluded that based on several tests, the performance of MQTT in IoT ecosystem is better than that of HTTP. This is due to the nature of MQTT which is lightweight that made it suitable to be used by IoT devices. MQTT is built to be used by devices in constrained environment by minimizing the resource usage, thus why it is smaller and lighter that HTTP. Furthermore, due to the smaller and lighter size of MQTT packet length, the response time of also faster as compared to HTTP. The total size of MQTT message is smaller than half the size of HTTP message, thus the reason why MQTT is faster. Lastly, due to the size and performance of MQTT, low-power devices are able to operate without great impact in device resources as compared to HTTP.

References

Bharati Wukkadada ; Kirti Wankhede ; Ramith Nambiar ; Amala Nair. (2018). Comparison with HTTP and MQTT In Internet of Things (IoT) - IEEE Conference Publication. *2018 International Conference on Inventive Research in Computing Applications (ICIRCA 2018)*, *Icirca*, 249–253. Retrieved from https://ieeexplore-ieee-org.e.bibl.liu.se/document/8597401

Evanczuk, S. (2018, May 22). *Creating a Raspberry Pi IoT Gateway | DigiKey*. Retrieved from https://www.digikey.my/en/articles/use-a-raspberry-pi-and-mozilla-open-framework-to-rapidly-develop-an-iot-gateway

Kukkamäki, J., Costa, R., Heck Júnior, V., & Bitencourt, E. N. (2018). *IoT Centralization and Management Applying ThingsBoard Platform*. *August*. Retrieved from https://www.hamk.fi/wp-content/uploads/2019/03/Project-Report_bitencourt_anjos.pdf

MQTT. (n.d.). *MQTT - The Standard for IoT Messaging*. (n.d.). Retrieved January 24, 2021, from https://mqtt.org/

Muquit, S. P., Yadav, D., Bhaskar, L., & Ahmed, W. F. (2019). IoT based smart trash bin for waste management system with data analytics. *Proceedings of the 2018 International Conference On Communication, Computing and Internet of Things, IC3IoT 2018*, 137–142. Retrieved from https://doi.org/10.1109/IC3IoT.2018.8668113

Naik, N. (2017). Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP. *2017 IEEE International Symposium on Systems Engineering, ISSE 2017 - Proceedings*. Retrieved from https://doi.org/10.1109/SysEng.2017.8088251

Nuratch, S. (2019). Applying the MQTT protocol on embedded system for smart sensors/actuators and IoT applications. *ECTI-CON 2018 - 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, 628–631. Retrieved from https://doi.org/10.1109/ECTICon.2018.08619981

Paessler. (n.d.). Why MQTT Is Everywhere, and the Security Issues It Faces. Retrieved April 19, 2020, from https://blog.paessler.com/why-mqtt-is-everywhere-and-thesecurity-issues-it-faces

Sethi, P., & Sarangi, S. R. (2017). Internet of Things: Architectures, Protocols, and Applications. *Journal of Electrical and Computer Engineering*, *2017*. Retrieved from https://doi.org/10.1155/2017/9324035

Sharma, C., & Gondhi, N. K. (2018). Communication Protocol Stack for Constrained IoT Systems. *Proceedings - 2018 3rd International Conference On Internet of Things: Smart Innovation and Usages, IoT-SIU 2018*, 1–6. Retrieved from https://doi.org/10.1109/IoT-

SIU.2018.8519904

Syafarinda, Y., Akhadin, F., Fitri, Z. E., Yogiswara, Widiawanl, B., & Rosdiana, E. (2018). The Precision Agriculture Based on Wireless Sensor Network with MQTT Protocol. *IOP Conference Series: Earth and Environmental Science*, *207*(1). Retrieved from https://doi.org/10.1088/1755-1315/207/1/012059

Tech Target. (n.d.). What is IoT (Internet of Things) and How Does it Work?. Retrieved October 23, 2020, from https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT

Toldinas, J., Lozinskis, B., Baranauskas, E., & Dobrovolskis, A. (2019). MQTT Quality of Service versus Energy Consumption. *Proceedings of the 23rd International Conference Electronics 2019, ELECTRONICS 2019*, 4–7. Retrieved from https://doi.org/10.1109/ELECTRONICS.2019.8765692

Tseng, C. L., Cheng, C. S., Hsu, Y. H., & Yang, B. H. (2019). An IoT-Based Home Automation System Using Wi-Fi Wireless Sensor Networks. *Proceedings - 2018 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2018*, 2430–2435. Retrieved from https://doi.org/10.1109/SMC.2018.00417

Yokotani, T., & Sasaki, Y. (2017). Comparison with HTTP and MQTT on required network resources for IoT. *ICCEREC 2016 - International Conference on Control, Electronics, Renewable Energy, and Communications 2016, Conference Proceedings*, 1–6. Retrieved from https://doi.org/10.1109/ICCEREC.2016.7814989