ISSN No:-2456-2165

# Comparison of IoT Protocols Performance

Yaseen NADIR
Information Institution
Gazi University
Ankara, Turkey

SUPERVISOR
Dr. Mutlu Tahsin ÜSTÜNDAĞ
Assoc. Prof., Gazi Faculty of Education
Gazi University
Ankara, Turkey

**Abstract:- Internet of Things, it is been more than a decade since this concept was introduced to the society. In this research we aim to compare three IoT application protocols; Advanced Messaging Queuing Protocol (AMQP), Message Queuing Telemetry Transport (MQTT) and Constrained Application Protocol (CoAP). The selected protocols efficiency will be evaluated using indicators related to Throughput and Round-Trip time (RTT). In the considered scenario an IoT device will transfer data to a server and waits for the response. The data will be sent in different sizes of packets. We have proposed a testbed using python programing language's library SciPy and socket programming to evaluate the three protocols. Experimentation tests reveal which protocol is best suited for different scenarios accordingly. Results show that overall MQTT achieves the highest protocols efficiency among other protocols.**

***Keywords:-*** *IoT, IoT Protocols, Performance, MQTT, AMQP, COAP.*

## I. INTRODUCTION

The internet is the most important and transformative technology have ever invented, it is like digital fabric that woven into our lives and rapidly changed the world. Likewise, new technology has emerged and it is poised to change the world again, this technology is not about connecting people or computers, it is about connecting objects "The Internet of Things". The internet of things represents a vision wherein the net extends into the actual world embracing daily objects. Physical objects are no longer disconnected from the digital world, however these objects can be managed remotely and can act as physical access points to Internet services. In the Network of things, objects can start to share experience with other objects, by adding the ability to sense, communicate to control and collaborate with each other, things that are connecting to the internet allowing them to generate, send, receive and exchange data.

IoT communication protocols are a subject of many studies in research community. Bhattacharyya et al. (Soma Bandyopadhyay, Abhijan Bhattacharyya., 2013) have done a comparison between CoAP's request-response model and MQTT publish-subscribe model. Application layer rules for IoT are explored in (Sotirios Kontogiannis, Angelos Chatzimparmpas, George Kokkonis., 2015); here, the authors have presented quantitative examinations of CoAP and MQTT in terms of traffic, packet loss probability, and latency conditions. Lavinia Nastasa in her research concentrates on application layer protocols; CoAP, MQTT, and XMPP from security point of view, described briefly the three protocols and their vulnerabilities, and according to the study, it is none of them are best for any type of solutions in terms of both security and functionality (Nastasa, 2017). Alvin VALERA and Hwee TAN, have proposed a common middleware using common application programming interface, and have tested CoAP and MQTT protocols, the research results shows that; different network conditions can affects the performance of different protocols, MQTT has lower delay for lower packet loss and vice versa, CoAP generates less traffic to ensure reliable delivery when the massage size is small and loss rate is equal or less than 25% (Valera & Tan, 2014). In (Paolo Bellavista, Alessandro Zanni., 2016), the authors proposed an innovative scalable distributed architecture for efficient IoT-cloud integration. The two selected protocols were CoAP and MQTT. The performance benchmark for the time required to complete the transmission of 1000, 10000 and 60000 messages was set and according to the authors, the MQTT protocol was much faster than CoAP.

There are several published survey papers regard IoT application layer protocols but as IoT technology is growing tremendously fast, it was important to test these application protocols with different metrics. This experiment will examine the following selected application services used for IoT data transfer:

### *A. Constrained Application Protocol (CoAP)*

Constrained Application Protocol was created and standardized by the Internet Engineering Task Force (IETF) Constrained Restful Environment working group (CoRE). CoAP is specialized internet application protocol for constrained nodes and networks based on Representational State Transfer (REST) (Shelby, Z; Hartke, K; Bormann, 2014). CoAP is a RESTful application protocol but in order to understand CoAP clearly the REST concept must be introduced first. Representational State Transfer (REST) is a web architectural style. RESTful systems are classified as they are stateless and separate concerns of client and servers. REST architecture is request-response architecture, client sends request to server in order to get to stored resources in it and these each resource has a Uniform Resource Identifier (URI) as an address. In case where server response, a content-type must be included in the header of the response. Unlike REST, CoAP uses datagram-oriented transport such

as UDP layer to keep design simple, with a primary goal of providing communication that works between devices with limited resources or networks with low bandwidth. CoAP supports the use of multicast IP destination addresses, enabling multicast CoAP requests. The CoAP message format is encoded in Binary format (Shelby, Z; Hartke, K; Bormann, 2014). CoAP message format starts with four fixed-size 4 bytes header consisting five fields. As CoAP runs over UDP, it is secured using Datagram Transport Layer Security.

*B. Advance Messaging Queuing Protocol (AMQP)*

The Advance Messaging Queuing Protocol is a binary and an open standard application layer protocol for message oriented middleware. It is been designed to provide peer-to-peer (point-to-point and publish and subscribe) routing, message orienting, queuing, reliability based on underlying on transport layer protocol such as Transmission Control Protocol (TCP), security based on Simple Authentication and Security Layer (SASL) and Transport Layer Security (TLS). The Advanced Message Queuing Protocol (AMQP) was released by various financial institutes and software companies in 2006 and was standardized by OASIS1 in the year 2012 (Turowski, Bosse, Kubela, & Pohl, 2018).

AMQP's main components are clients and servers (brokers). Broker consist of: exchanges, message queue and binding, message routing occur using these three components. Exchange receives messages from publisher and rout it to the appropriate queue. Message Queue stores the messages until the meant consumer client (subscriber) processes it safely. Biding, defines the relationships between exchanges and queues. There are two security approaches provided by AMQP: Transport layer Security (TLS), and Simple Authentication and Security Layer SASL.

*C. Message Queue Telemetry Transport (MQTT)*

MQTT is message-oriented protocol, a client server publish-subscribe message transport protocol. It is an extremely light-weight, open, simple, designed to be easy to implement for high-latency or unreliable network (Konstantinos, et al., 2016).The MQTT protocol run over TCP/IP networks. MQTT was created by IBM then standardized by OASIS. The publish-subscribe model of MQTT is based on the Client-Server model, and the server is more like a broker or gateway, an MQTT Broker is a device that acts as intermediary between client which publish and client which has made subscription. MQTT delivers application messages with three QoS levels:

- QoS 0: "At most once", where the message is delivered but no response is sent by the receiver and no resend message is performed by the sender, lost can occur.
- QoS 1: "At least once", this QoS enures the delivery of message to the receiver at least once, duplicated can occur.
- QoS 2: "Exactly once delivery", this is where the message is assured to arrive at the receiver with no loss or duplication, it is the highest level of QoS.

The important part is that there is no security features in MQTT specifications, the security of each implementation is done according to its design. Since the MQTT protocol is based on TCP/IP and it is implementations responsibility to provide appropriate security and integrity, security features must be implemented on top of MQTT. This can be achieved by using Transport Layer Security TLS/Secure Socket Layer SSL. Authentication is based on certificates and privacy is based on encryption mechanism by the application.

Objectives of the Study;

- Evaluating and comparing the performance of IoT protocols for different network scenarios.
- Is protocol efficiency affected by different conditions of the network?
- Is the Round Trip Time affected by network conditions?

Although there are many studies on the Internet of Things, It is believed that there are still a need for performance analysis studies in different conditions of different networks. Given the significant impact of this situation, more studies are needed. In this study, evaluation of the effectiveness of IOT protocols and comparison will be examined. Information about the selected protocols will be collected based on the previous studies, these protocols will be evaluated and analyzed with the proposed testbed and comparisons will be revealed within the framework of the determined indicators.

The methodology of obtaining, testing and analyzing data is based on several stages:

Firstly: Learning and studying IoT protocols MQTT, AMQP, COAP and these studies will be used for scientific research.

Secondly: Designing the proposed testbed with different scenarios using python programming language.

Thirdly: implementing, simulating, and testing IoT protocols.

Fourthly: After the implementation and testing of the protocol, the data will be analyzed using Anaconda environment to facilitate the qualitative analysis process.

## II. EXPERIMENTAL DESIGN

In this section, we discuss the implementation of the proposed testbed and evaluate the performance of its proposed algorithm with different routing protocols, e.g. AMQP, MQTT, and COAP.

The proposed software consists of three structures. They are the IoT sensor structure, the gateway module structures, and the cloud application server structures. The first structure is the IoT sensor, where the proposed software simulates a light version of the IoT device, which generates packets in three different packet formats represent the routing protocols, i.e., COAP, MQTT, and AMQP. The second structure is the gateway module, which is up and waiting to sniff different packets coming from a group of IoT sensors and forward them to the cloud application server acting as the man in the middle. The third structure is the cloud application server, which is up and running tending on its listening port to receive connection requests from the

gateway. This server represents the Application Enablement Platform (AEP) in the IoT value chain.

In our simulation, the performance evaluation of the proposed algorithm has been achieved taking into consideration the following evaluation metrics:

• Upstream processing delay: it is the execution time to decapsulate the IoT device packet header, encapsulate the payload into the TCP/UDP header and send the packet to the cloud application server.

• Downstream processing delay: it is the execution time to decapsulate the TCP/UDP header, encapsulate the payload into the IoT device packet header and send the reply back to the corresponding IoT device.

• Average processing delay: it is the round-trip processing time to process the IoT device packet while traveling in both upstream and downstream directions.

• Upstream throughput: it is the total number of packets per second that the gateway can handle in the upstream direction from the IoT device to the cloud application server.

• Downstream throughput: it is the total number of packets per second that the gateway can handle in the downstream direction from the cloud application server to the IoT device.

## III. RESULTS AND DISCUSSION

We have tested the performance by changing the number of packets sent by IoT devices, and check the average processing delay of the upstream, the downstream, and the round-trip processing delay, besides the upstream and downstream throughput.

### *D. Advanced Messaging Queuing Protocol (AMQP)*

Figure 1 shows that in AMQP protocol, the average upstream processing delay is shorter than the average downstream processing delay. It means that the packet decapsulation from AMQP packets and encapsulation into the TCP/UDP header takes lower processing time than decapsulating the payload from the TCP/UDP header and encapsulating it into the AMQP header. Moreover, increasing the total number of packets sent from the IoT devices doesn't have a high effect on the average upstream processing delay, while it has a high effect on the downstream processing delay, accordingly downstream processing delay has a direct effect on the total round-trip processing time.

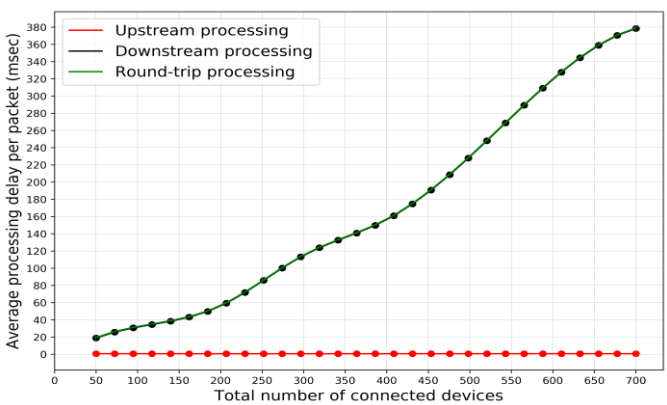


Figure 1: The relation between average processing delay and number of connected devices in AMQP.

Figure 2 depicts the upstream and downstream throughput of the gateway in the case of AMQP routing protocol. It shows that downstream throughput is almost four times higher than upstream throughput. Also, as the total number of packets sent by IoT devices increases, the upstream and downstream throughput of the gateway decreases slightly.

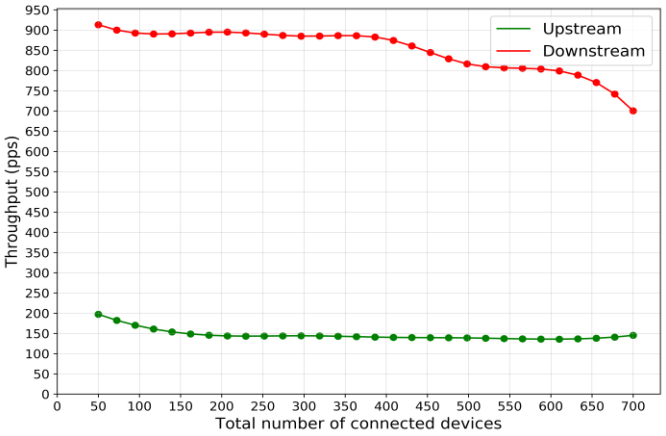


Figure 2: The relation between throughput and number of connected devices in AMQP.

### *E. Message Queuing Telemetry Transport (MQTT)*

Figure 3 shows that in the MQTT protocol, the average upstream processing delay is shorter than the average downstream processing delay. It means that the packet decapsulation from MQTT packets and encapsulation into the TCP/UDP header takes lower processing time than decapsulating the payload from the TCP/UDP header and encapsulating it into the MQTT header.

Besides, as the total number of connected devices increases, the upstream processing delay is almost constant for different numbers of connected devices, while the downstream processing time increases, accordingly the total average round-trip processing delay increases as well. This means the total number of connected devices and the downstream processing time have a direct effect on the scalability performance.

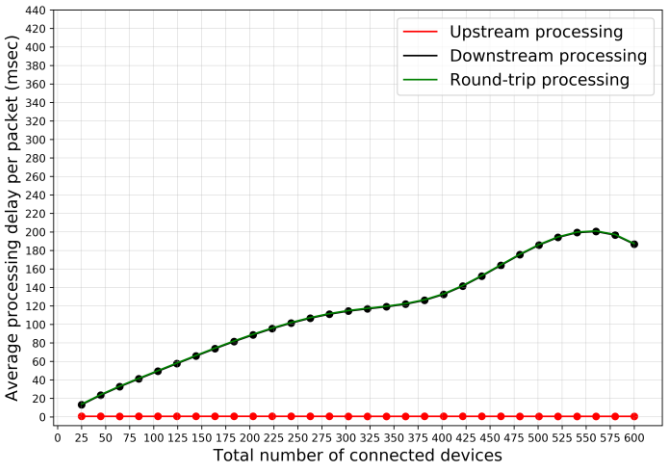


Figure 3: The relation between average processing delay and number of connected devices in MQTT.

ISSN No:-2456-2165

Figure 4 depicts the upstream and downstream throughput of the gateway in the case of the MQTT routing protocol. It shows that downstream throughput is almost seven times higher than upstream throughput. Also, as the total number of packets sent by IoT devices increases, the downstream throughput of the gateway decreases slightly.

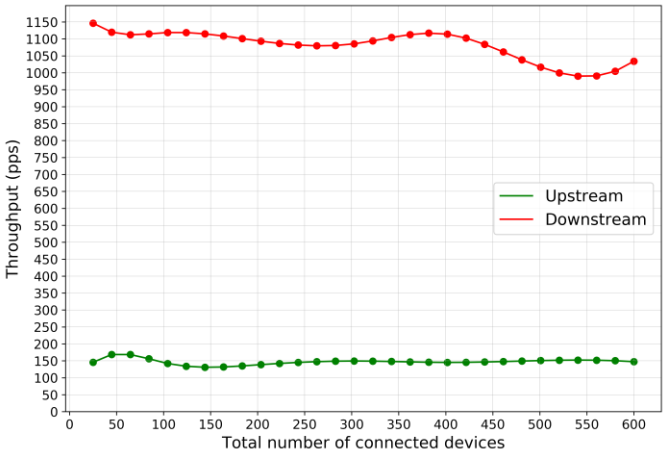


Figure 4: The relation between throughput and number of connected devices in MQTT.

*F. Constrained Application Protocol (COAP)*

Figure 5 shows that in COAP protocol, the average upstream processing delay is shorter than the average downstream processing delay. It means that the packet decapsulation from COAP packets and encapsulation into the TCP/UDP header takes lower processing time than decapsulating the payload from the TCP/UDP header and encapsulating it into COAP header.

Also, as the total number of connected devices increases, the upstream processing delay is almost constant for different numbers of connected devices, while the downstream processing time increases, accordingly the total average round-trip processing delay increases as well. This means the total number of connected devices and the downstream processing time have a direct effect on the average round-trip processing delay.

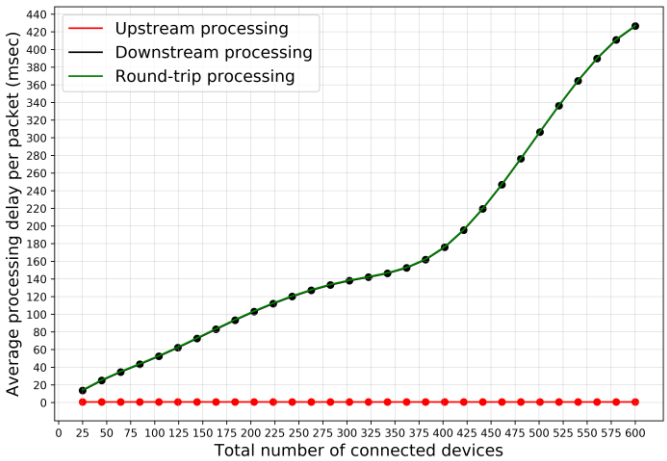


Figure 5: The relation between average processing delay and number of connected devices in COAP.

Figure 6 shows the upstream and downstream throughput of the gateway in the case of COAP routing protocol. It shows that downstream throughput is almost six times higher than upstream throughput. Also, as the total number of packets sent by IoT devices increases, the downstream throughput of the gateway decreases.

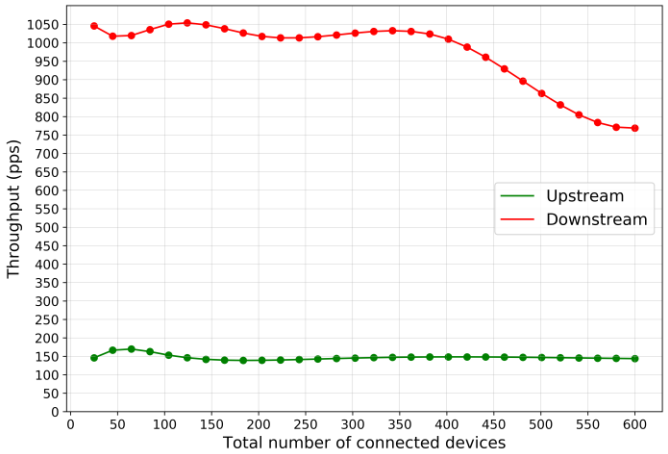


Figure 6: The relation between throughput and number of connected devices in COAP.

Figure 7 shows the relation between the average processing delay and the number of connected IoT devices for AMQP, MQTT, and COAP routing protocols. It is obvious that for small traffic volume, e.g., 150 connected devices, AMQP routing protocol has the highest average round-trip processing delay among other routing protocols. COAP routing protocol comes second, and then MQTT has the lowest average round-trip processing delay. While for the high traffic volume, e.g., 550 connected devices and beyond, AMQP routing protocol provides lower round-trip processing delay than COAP, however MQTT still the lowest round-trip processing delay among other routing protocols.

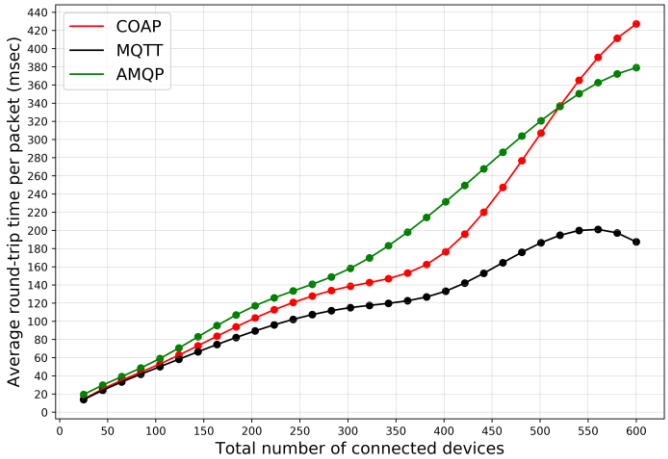


Figure 7: The relation between average processing delay and number of connected devices for all routing

On the other hand, Figure 8 depicts the relation between the upstream and downstream throughput with the total number of connected IoT devices for AMQP, MQTT, and COAP routing protocols. It is obvious that there is no

high difference in the upstream throughput for all routing protocols, as they are almost closed to each other. While in the downstream throughput, the results show that the MQTT routing protocol has the highest throughput, and COAP is the second highest, then AMQP is the lowest throughput among all routing protocols.

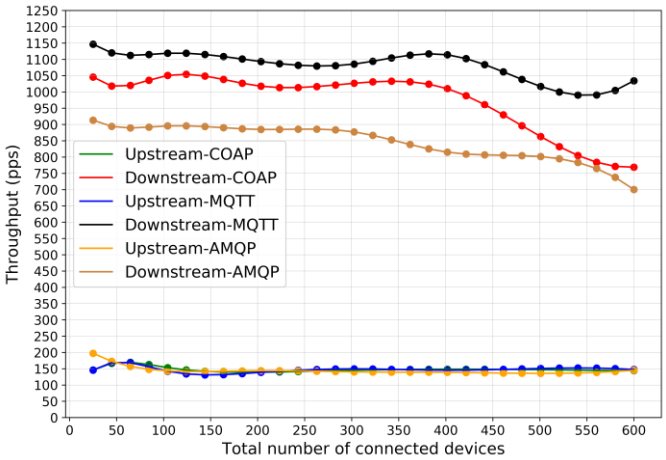


Figure 8: The relation between throughput and number of connected devices for all routing protocols.

## IV. CONCLUSION

In this research the selected application layer protocols of Internet of things are been simulated using the proposed testbed software and number of tests with different volume traffics were run on the three protocols performance.

The results show that for low volume round-trip time, every protocol has almost the same round-trip processing delay. While the number of packets increases the difference between the protocols increases too, as a result for high volume round-trip time, MQTT has the lowest average round-trip processing delay among other protocols. Likewise, the upstream throughput results clearly show there is no high difference between all protocols, as every protocol almost close to each other. While according to the results of the tests, MQTT protocol became the highest throughput among the protocols in terms of downstream throughput.

Experimental results showed that the performance of different protocols are dependent on different network conditions. Also, one of the most important points that is been observed that the header size of each protocol plays a big role in the efficiency of the routing protocol, as a fixed size payload is been used for all tests.

## REFERENCES

[1]. Konstantinos, Fysarakis, Ioannis Askoxylakis, Othonas Soultatos, Ioannis Papaefstathiou, Charalampos Manifavas, and Vasilios Katos. 2016. "Which IoT Protocol? Comparing standardized approaches over a common M2M application."

[2]. Nastasa, Lavinia. 2017. "Security in the Interner of Things: A survey on Application Layer Protocol."

[3]. Paolo Bellavista, Alessandro Zanni. 2016. "Towards Better Scalability for IoT-Cloud Interactions." *IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI).*

[4]. Shelby, Z; Hartke, K; Bormann. 2014. "The Constrained Application Protocol (CoAP)." *RFC 7252.*

[5]. Soma Bandyopadhyay, Abhijan Bhattacharyya. 2013. "Lightweight Internet Protocols for Web Enablement of Sensors using Constrained." *International Conference on Computing, Networking and Communications, Workshops Cyber Physical System.*

[6]. Sotirios Kontogiannis, Angelos Chatzimparmpas, George Kokkonis. 2015. "Middleware IoT protocols performance evaluation for carrying out clustered data."

[7]. Turowski, Klaus, Sascha Bosse, Janick Kubela, and Matthias Pohl. 2018. "Performance Evaluation of Application Layer."

[8]. Valera, Alvin Cerdeva, and Hwee Xian Tan. 2014. "Performance Evaluation of MQTT and CoAP.