

IoT 환경에서의 메시징 시스템의 성능 분석에 관한 연구

이영동

창신대학교 스마트융합공학부 컴퓨터전공

A Study on Performance Analysis of a Messaging System in IoT Environments

Young-Dong Lee

Division of Smart Convergence Engineering, Changshin University

요 약 IoT 기술은 인터넷과 사물이 연결되고, 사물 스스로가 데이터를 분석, 판단하여 실세계와 가상세계를 실시간으로 상호 연결하는 단계로 발전하고 있다. 이러한 IoT 기술은 센서, 액추에이터, 장치, 네트워크로 구성되어 다양한 분야에 적용되고 있으며, IoT 디바이스와 애플리케이션 수가 증가함에 따라 데이터 트래픽도 증가하고 있다. 본 논문에서는 IoT 환경에서의 IoT 디바이스와 브로커 간 성능 분석을 위해 IoT 디바이스에서 센서 데이터를 수집하고 MQTT 프로토콜을 통해 MQTT 브로커에 전달하는 시스템을 구현하였다. MQTT 기반 RTT 측정과 IoT 디바이스와 브로커 간 메시지 전송 시간을 측정하는 실험을 수행한 결과 패킷 손실은 발생하지 않았으며, 전파 지연이 RTT에 영향을 주는 것으로 실험을 통해 알 수 있었다.

• 주제어 : 사물인터넷, MQTT, 서비스 품질, 왕복시간, 전송 지연

Abstract Internet of Things(IoT) technology is developing to a stage where the Internet and objects are connected and objects themselves analyze and judge data to interconnect the real world and the virtual world in real time. This technology consists of sensors, actuators, devices, and networks, and it is being applied in various fields. As the number of IoT devices and applications increases, data traffic also increases. In this paper, a messaging system is designed and implemented in order to analyze the performance between an IoT device and MQTT broker. The experimental was performed to measure MQTT-based round-trip time and message transmission time between the IoT device and the broker. The result shows that there is no packet loss, and propagation delay affects round-trip time.

• Key Words : Internet of Things, MQTT, Quality of Service, round-trip time, transmission delay

Received 17 May 2023, Revised 26 June 2023, Accepted 29 June 2023

* Corresponding Author Young-Dong Lee, Dept. of Computer Software Engineering, Changshin University, 262, Paryong-ro Changwon-si, Korea. E-mail: ydlee@cs.ac.kr

I. 서론

사물인터넷(IoT) 환경은 센서, 게이트웨이, 라우터, 애플리케이션, 플랫폼과 같은 IoT 생태계의 모든 지점 간의 연결을 의미한다. IoT 솔루션은 수백 개에서 수천 개에 이르는 디바이스에 통신 기술을 더해 사물과 사물, 사물과 사람을 연결한다. 수많은 디바이스들이 인터넷에 연결되도록 지원하는 IoT의 유무선 통신 및 네트워크 장치로는 이더넷, WiFi, Bluetooth, Zigbee, Sigfox, LoRaWAN, NB-IoT, LTE-M 등 다양한 유형의 네트워크 인프라 기술을 기반으로 하고 있다. IoT 솔루션 설계 단계에서는 일반적으로 전력 소비, 전송 범위 및 전송률에 따라 필요한 시스템 기능을 고려하고 시스템 요구 사항에 따라 최적의 성능, 비용을 고려하는 것이 중요하다.

현실과 가상세계를 결합하여 스마트한 미래를 제공하는 혁명 기술인 IoT 관련 기술은 무한한 잠재력으로 시장 참여자와 투자를 끌어올리며 확대되고 있으며, 상업용 및 산업용 IoT 기반 기업 간의 경쟁이 치열해짐에 따라 선두 주자가 되기 위해 제공할 수 있는 킬러 시장 차별화 요소를 찾아야 할 필요성이 커지고 있다. IoT 혁명은 산업 및 사회 전반에 걸쳐 효율성, 지속 가능성 및 안전성 측면에서 엄청난 잠재적 가치를 제공하며, 전 세계 IoT 시장 규모가 2021년 3,845억 달러에서 2027년에는 5,664억 달러로, 연간 6.7%로 성장할 것으로 전망하고 있다. 이는 5G 통신 기술의 보급, 클라우드 플랫폼 채택 증가로 인한 데이터 센터 필요성 증가, 무선 스마트 센서 및 네트워크 사용자 증가, IP 주소 증가와 향상된 보안 솔루션 등이 시장 성장을 견인할 것으로 분석되고 있다[1]. Ericsson은 2025년까지 50억 개의 IoT 장치가 셀룰러 3GPP 액세스 기술을 통해 연결될 것으로 예측 발표하였다[2].

4차산업혁명시대 핵심 성장동력으로 떠오르고 있는 IoT 기술의 진화는 연결형 IoT, 지능형 IoT, 자율형 IoT 단계로 발전하고 있다. 이러한 발전은 IoT 환경에서는 다양한 애플리케이션에서 센서로부터 다양한 데이터가 실시간으로 발생하게 되고, 중단 간 신뢰성을 갖춘 데이터 송수신 기술 개발이 매우 중요한 요소라고 볼 수 있다. 즉, 디바이스와 애플리케이션 간의 효율적이고 신뢰할 수 있는 데이터 수집을 위한 시스템 구현뿐만 아니라 국제 표준 기반의 메시징 프로토콜 적용이 요구된다. 이에 본 논문에서는 실제 IoT 디바이스에서 센

서 데이터를 수집하고, 국제 표준화된 구독 발행 기반의 Message Queuing Telemetry Transport(MQTT) 프로토콜을 통해 MQTT 브로커에 전달하는 시스템을 구현하였다. 그리고 IoT 환경에서의 메시징 시스템의 성능 분석을 위해 Round-Trip Time(RTT), 시퀀스 번호, latency 측정을 통한 성능 평가를 진행하였다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 publish/subscribe 메시징 시스템에 대해 소개하고, 3장에서는 IoT 메시징 시스템 구현에 관해 기술한다. 4장에서 실험 환경을 토대로 수행한 실험 및 평가에 대해 제시하고, 5장에서 논문의 결론을 맺는다.

II. Publish/subscribe

Publish/subscribe(Pub/Sub) 시스템은 분산 컴퓨팅에서 구독 클라이언트와 발행 클라이언트 간의 통신 모델 역할을 하게 된다. Publisher(발행자)는 데이터를 생성하고 Pub/Sub 시스템에 메시지를 publish(발행) 하며, Subscriber(구독자)는 Pub/Sub 시스템에서 메시지를 subscribe(구독)하고 해당 메시지를 처리한다[3].

MQTT는 Pub/Sub 시스템 중 하나이며, 이 프로토콜은 센서와 같은 장치에서 생성된 데이터를 클라우드 서버 또는 기타 시스템으로 전송하기 위한 경량 메시징 프로토콜이다. MQTT는 네트워크 대역폭이 제한되는 원격 위치와의 연결을 위해 설계되었으며, 센서, 스마트폰, 휴대용 장치 등의 IoT 장치에서도 매우 효율적으로 작동한다. MQTT는 TCP/IP에서 동작하며, 일반적으로 세 가지의 서비스 품질(QoS)의 세 가지 수준을 제공한다[4]. QoS 0은 메시지를 한 번만 전송하고 전송 여부는 확인하지 않아 메시지가 목적지에 도달하는 것을 보장하지 않는다. QoS 1은 메시지의 최소 한 번 전송을 보장하고 브로커가 PUBACK 패킷을 보내어 메시지를 성공적으로 받았음을 클라이언트에게 알리게 되지만 만약 브로커가 PUBACK 패킷을 클라이언트에게 보내기 전 연결이 끊어지면 메시지가 중복으로 전달될 경우가 발생한다. QoS 2는 메시지의 정확히 한 번 전송을 보장하며, PUBREC, PUBREL, PUBCOMP 세 개의 패킷을 사용하여 메시지 전송의 확인 및 재전송을 보장한다[5]. 가장 안전하지만 클라이언트와 브로커 간 4-way handshake를 통해 전달되는 구조라 너무 많은 동작에 따른 트래픽 과부하가 발생하는 단점이 있다.

MQTT의 시스템 구조는 Pub/Sub 방식의 메시지 큐

로 데이터를 송신하고 수신하기 위해 발행자, 구독자, 토픽, 브로커로 구성된다. 일반적으로 발행자가 토픽을 발행하면 브로커가 이를 중개하고, 구독자는 브로커를 받아들일 토픽을 구독하며, 발행자와 구독자가 직접 메시지를 주고받지 않기 때문에 비동기 방식이다. MQTT 프로토콜의 메시지 교환은 발행과 구독 토픽을 기준으로 작동하며, 토픽은 슬래시(/)를 사용하여 계층적 구조를 구분하는데, 주제를 파일 시스템 형식으로 나눌 수 있어서 효율적인 관리가 가능하다[6].

III. IoT 메시징 시스템

IoT 메시징 시스템은 그림 1과 같이 IoT 디바이스, MQTT 브로커, MQTT 구독 역할을 담당하는 Node-RED와 Telegraf, 데이터베이스, 시각화 도구로 구성된다. 먼저 IoT 디바이스는 ESP32-WROOM-32[7] 모듈, 온도, 습도 센서 등을 갖추고 있으며, WiFi 무선 통신을 통해 디바이스-디바이스, 디바이스와 브로커 간 통신이 가능하다. 또한, MQTT 시스템 구조에 따라 IoT 디바이스가 발행자 역할을 하게 되고, 디바이스에 장착된 센서를 통해 수집된 데이터를 토픽으로 발행하여 MQTT 브로커인 Mosquitto가 중개하게 된다. IoT 디바이스와 브로커 구간과 브로커와 MQTT 구독자 구간은 MQTT 프로토콜을 통해 메시지를 교환하게 된다. MQTT 구독은 Node-RED와 Telegraf를 적용하여 목적에 따라 선택하여 사용할 수 있도록 구성하였다. Node-RED와 Telegraf는 센서 데이터가 게시되는 채널을 구독하는 MQTT 브로커에 연결하고 이 정보를 InfluxDB에 저장하는 역할을 한다. InfluxDB는 시계열 데이터 수집 및 처리에 최적화된 데이터베이스로서 온도, 습도 센서 정보를 일정 주기마다 수집하고 Node-RED와 Telegraf에서 구독한 토픽을 저장한다. 마지막으로 시각화 도구는 InfluxDB와 연동된 Grafana를 통해 데이터를 읽고 대시보드를 관리한다.

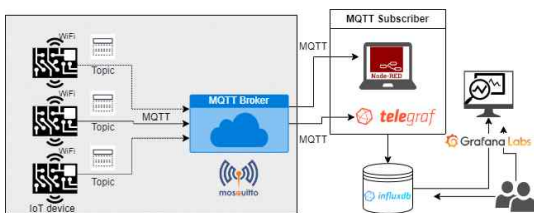


Fig. 1. System architecture

그림 2는 MQTT 발행자 역할을 하는 IoT 디바이스와 MQTT 브로커 간 프로토콜 동작에 따른 메시지 흐름을 나타낸 것이다. MQTT는 TCP/IP 기반의 프로토콜로서, MQTT 메시지는 TCP/IP 패킷으로 캡슐화되어 전송된다. 먼저, MQTT 클라이언트는 MQTT 브로커에 연결되며, TCP 연결을 설정하기 위해 3-way handshake가 수행된다. 3-way handshake는 SYN, SYN-ACK, ACK의 3개의 패킷을 교환하여 연결을 설정하게 된다. 연결이 설정되면, MQTT 클라이언트는 MQTT 브로커에 CONNECT 메시지를 보내며, CONNECT 메시지에는 클라이언트 식별자, 유저네임, 패스워드, 연결 설정 등의 정보가 포함되어 있다. MQTT 브로커는 CONNECT 메시지를 받으면, 클라이언트를 등록하고, 이후에 발행된 메시지를 해당 클라이언트에게 전달할 수 있다. 이후 MQTT 클라이언트는 발행 메시지를 브로커에게 보내고, 브로커는 해당 메시지를 해당 토픽을 구독하고 있는 모든 클라이언트에게 전달한다. MQTT 클라이언트는 구독을 통해 원하는 주제를 구독할 수 있으며, esp32/temperature1, esp32/humidity1, 수신신호강도(RSSI)와 같이 3개의 토픽을 구독하였다. MQTT 브로커는 해당 3가지 주제를 구독하고 있는 모든 클라이언트에게 해당 메시지를 전달하게 된다.

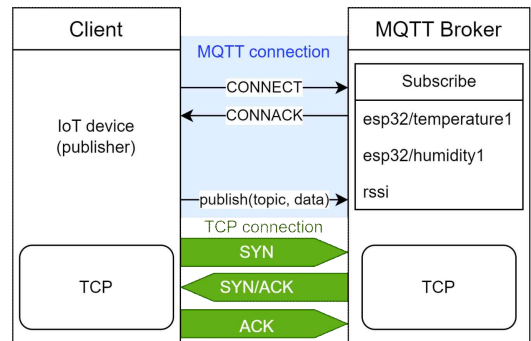


Fig. 2. Broker based MQTT protocol

그림 3은 MQTT 브로커와 InfluxDB간 연동을 위해 Node-RED 흐름 구성을 나타낸 화면이다. IoT 디바이스에 장착된 온도, 습도 센서 그리고 WiFi 수신신호세기를 표시하는 RSSI를 MQTT 포트에 연결하고 각 센서들의 차트, 디버그, influxdb out 노드로 구성하였다. Node-RED를 사용하여 MQTT 브로커에서 InfluxDB로 데이터를 보내게 되며 이때 Node-RED는 브리지 역할

을 하게 된다. influxdb out 노드를 사용하여 InfluxDB의 버킷(bucket)과 value1, value2, ESP32_RSSI와 같이 3개의 _measurement를 설정하여 상호 연동되도록 하였다.

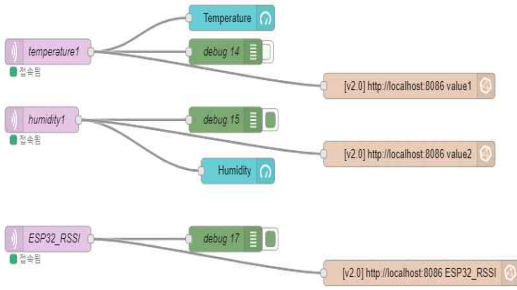


Fig. 3. Node-RED MQTT flow

데이터 수집 및 처리된 데이터는 그림 4와 같이 오픈 소스 IoT 시각화 도구인 Grafana를 사용하여 시계열 데이터를 모니터링할 수 있도록 구성하였다. 온도, 습도 센서 데이터는 수치, 차트와 로그로 나누어 실시간으로 출력되고 RSSI는 수치로 표시되도록 하였다.

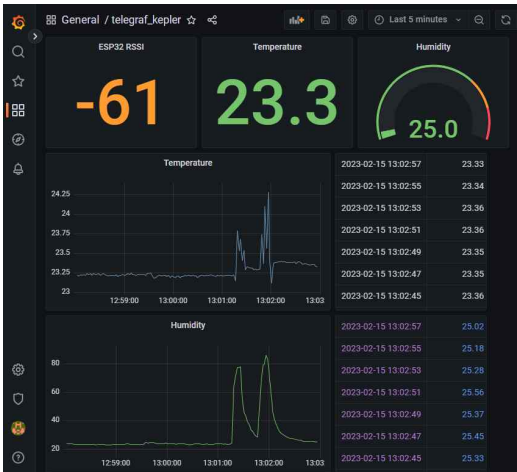


Fig. 4. Dashboard in Grafana via its GUI

IV. 실험 및 평가

이 장에서는 IoT 환경에서의 메시징 시스템의 성능 측정을 위해 구축한 실험 환경을 토대로 수행한 실험 결과 및 분석에 대해 설명한다. 본 논문에서는 3장에서 구현한 시스템을 활용하여 실험을 수행하고 그 성

능을 분석하며, 실험 환경은 표 1과 같다. 실험은 2가지 형태로 나누어 수행하였다.

Table 1. Description of the experimental environment

Component		Specification
IoT Device	MCU	ESP32-WROOM-32
	sensor	Temperature, Humidity
	Wireless	Wi-Fi
MQTT Broker		Mosquitto 5.0
Subscriber		Node-RED Telegraf 1.11.3 windows
DB Server		InfluxDB
Message sent		100 / 1,000 / 10,000
OS		Windows 11

첫 번째 실험은 MQTT기반 RTT[8]를 측정하기 위해 Wireshark 네트워크 패킷 분석 도구를 활용하였으며, IoT 디바이스에서 브로커로 전송한 MQTT 메시지를 포함한 모든 패킷을 캡처하여 분석하였다. MQTT에서 메시지의 재전송은 QoS 1과 QoS 2에서만 발생하며, QoS 0에서는 메시지 전송 후 재전송이 발생되지 않는다. QoS 0에서 latency를 측정하기 위해 MQTT 클라이언트인 IoT 디바이스에서 CONNECT 메시지를 브로커에게 전송한 후, 해당 메시지를 수신하는 브로커가 CONNACK 메시지로 응답을 하는 데 걸리는 시간을 측정하였다. 시퀀스 횟수에 따른 메시지(전송 메시지)를 1,000, 10,000회로 설정하여 실험을 수행하였으며, 시간별 RTT와 시퀀스 번호를 분석하였다. 시간별 RTT는 전송 지연 측정을 위한 것이고, 시간별 시퀀스 번호는 패킷 손실 여부를 분석하기 위한 것이다. 전송 메시지 1,000의 경우, 해당 시간 동안 평균값 범위에 포함되지 않은 200ms 이상의 소요 시간을 보인 지점을 분석하였다. 그림 5의 150초 부근을 살펴봤을 때 다른 구간에 비해 많은 시간이 소요되었으며 시퀀스 번호로 나타난 그림 6에서도 구간 끊어짐 현상이 발생함을 알 수 있었다.

그림 7은 해당 구간을 확대하여 나타난 그래프이며, 140~145초와 150~155초 구간의 시퀀스 번호를 분석해 본 결과 패킷 손실은 발생하지 않았고 일부 지연 현상이 발생한 것으로 분석되었다. 메시지를 10,000회를 전송한 경우도 앞서 살펴본 그림 5에서와 같이 동일하게 실험을 수행하였고, 그 결과는 그림 8과 같다. 앞서 수

행한 실험과 비교하였을 때 더 많은 RTT이 소요되었으며, 지연 시간이 높게 나타나는 횟수가 많이 발생하였다.

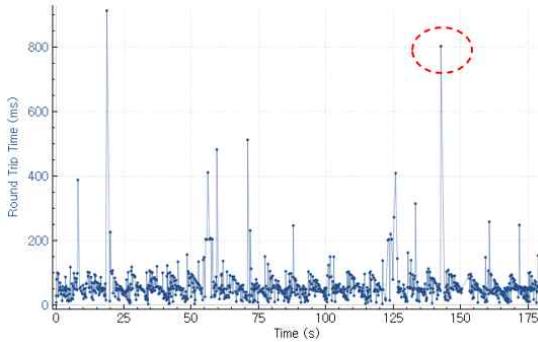


Fig. 5. RTT(ms) vs. Time(s) for 1,000 message

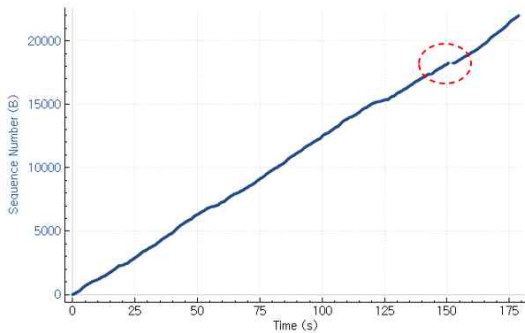


Fig. 6. Sequence number(B) vs. Time(s) for 1,000 message

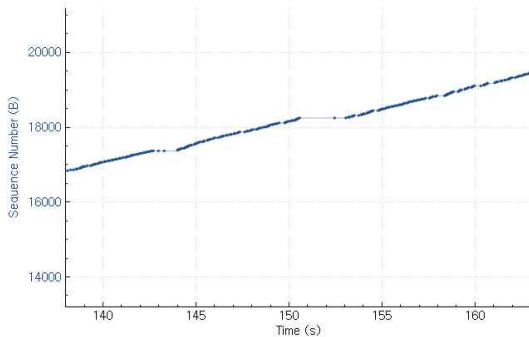


Fig. 7. Sequence number(B) vs. Time(s) for 1,000 message between 140-160 second

특히 1,400~1,500초에 해당하는 구간의 값을 분석하였고 그림 9의 시퀀스 번호로 나타난 그래프에서도 구간 끊어짐이 보였다. 1,430초 지점에서 발생함을 확인하였고 그림 10의 해당 구간을 확대 분석한 결과 1,430~1,445구간에서 지연 현상이 발생하였으나 패킷 손실이 발생하지 않았음을 확인할 수 있다.

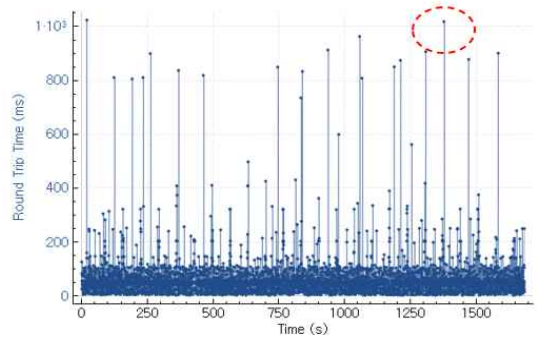


Fig. 8. RTT(ms) vs. Time(s) for 10,000 message

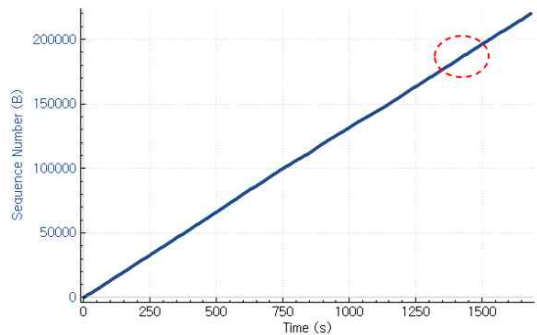


Fig. 9. Sequence number(B) vs. Time(s) for 10,000 message

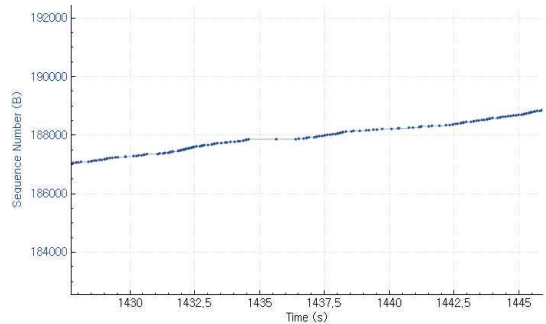


Fig. 10. Sequence number(B) vs. Time(s) for 10,000 message between 1,430-1,445 second

두 번째 실험은 IoT 디바이스(클라이언트)와 브로커 간에 메시지 전송에 따른 소요 시간을 측정하였다. 이를 위해 클라이언트에서 메시지를 보내기 전에 타임스탬프를 찍어서 브로커가 메시지를 받은 시간과 비교하여 전송 시간을 계산하였다[9]. 즉, 클라이언트에서 메시지가 전송된 시점의 타임스탬프와 브로커에서 해당 메시지를 수신한 시점의 타임스탬프 값의 차이로 성능을 측정하게 된다. 시퀀스 횟수에 따른 메시지를 100, 1,000, 10,000으로 구분하여 각 5번의 실험 후 측정된

평균 소요 시간에 관한 결과는 표 2와 같다. 100회의 메시지를 전송한 경우는 평균 106.23ms로 나타났으며, 1,000회와 10,000회에서는 각각 131.105ms, 121.7373ms의 평균 전송 지연을 보였다. 또한, 100, 1,000, 10,000회의 메시지를 전송한 경우 최솟값은 각각 6ms, 50ms, 45ms의 지연 시간을 보였다. 최댓값을 비교해 보면 100회에서 602ms 소요된 것보다 1,000, 10,000에서 각각 2,535ms, 2,566ms의 시간이 걸린 것으로 나타났다. 결과적으로 첫 번째 수행한 전송 메시지 1,000, 10,000 실험에서는 데이터를 전송하고 ack 신호를 수신하기 위한 전파 지연이 RTT에 영향을 줄 수 있으나 패킷 손실이 발생하지는 않는 것으로 분석되었다. 두 번째 실험에서는 일반적으로 전송 메시지가 많아질수록 트래픽 양이 증가하여 전송지연이 높게 나타날 것으로 가정하고 실험을 수행하였으나, 표 2와 같이 평균, 최소, 최댓값 등을 측정된 결과 전송 메시지가 많아질수록 전송 지연이 증가하는 것과 관계없이 전파 지연이 RTT에 영향을 주는 것으로 확인되었다.

Table 2. Result of MQTT latency measurement

Messages	Mean (ms)	Median (ms)	Min (ms)	Max (ms)	S.Dev (ms)
100	106.23	69	50	602	92.46794
1000	131.105	91	6	2535	135.23101
10000	121.7373	93	45	2566	114.34804

V. 결론

본 논문에서는 IoT 환경에서의 IoT 디바이스와 MQTT 브로커 간의 메시징 송수신 과정에서의 성능 분석을 수행하였다. MQTT 기반 RTT 측정과 클라이언트와 브로커 간 메시지 전송 시간 측정으로 구분하여 실험을 수행하였으며, 이를 위해 IoT 디바이스, MQTT 브로커, MQTT 구독자, 데이터베이스, 시각화 도구로 구성된 IoT 메시징 시스템을 구현하였다. IoT 디바이스에서 메시지 1,000, 10,000회를 전송하였을 때 일부 지연 현상이 발생하였으나, 패킷 손실은 발생하지 않았다. 첫 번째 수행한 전송 메시지 1,000, 10,000 실험에서는 데이터를 전송하고 ack 신호를 수신하기 위한 전파 지

연이 RTT에 영향을 줄 수 있으나 패킷 손실이 발생하지는 않는 것으로 분석되었다. 두 번째 실험에서는 전파 지연이 RTT에 영향을 미치며 전송 메시지가 많아질수록 트래픽 양이 증가하여 전송 지연이 높게 나타나는 것은 관계가 없음을 실험을 통해 알 수 있었다. 본 연구에서 얻은 결과를 토대로 다양한 산업현장에 국제 표준 기반의 메시징 프로토콜을 적용시켜 디바이스와 애플리케이션 간의 효율적이고 신뢰할 수 있는 데이터 수집 시스템 구현이 가능할 것으로 사료된다.

REFERENCES

- [1] IoT Technology Market-Analysis and Forecast to 2027, MarketsandMarkets, Oct. 2021.
- [2] GSMA Whitepaper, Mobile IoT in the 5G Future - NB-IoT and LTE-M in the Context of 5G, April 2018.
- [3] M. K. Hasan, "CoAP and MQTT Measurements over LoRaWAN," M.S. thesis, Faculty of Information Technology and Electrical Eng., Univ. of Oulu, Finland, 2020.
- [4] H. F. Coban, "An experimental comparison of messaging protocols MQTT and CoAP," M.S. thesis, Dept. Information Systems, Middle East Technical Univ., Turkey, 2017.
- [5] R. W. Tareq, et al, "Implementation of MQTT Protocol in Health Care Based on IoT Systems:A Study," International Journal of Electrical and Computer Engineering Systems, vol. 12, no. 4, pp. 215-223, 2021.
- [6] P. Julio, "MQTT performance analysis with OMNET++," M.S. thesis, IBM Zurich Research Lab., Switzerland, 2005.
- [7] ESP32-WROOM-32 datasheet, https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf
- [8] B. Bendele, et al, "A Study of IoT MQTT Control Packet Behavior and its Effect on Communication Delays," Electronic Imaging, pp. 120-129, 2017.
- [9] M. S. Rocha, "On the performance of OPC UA and MQTT for data exchange between industrial plants and cloud servers," Acta IMEKO, vol. 8, no. 2, pp. 80-87, 2019.

저자소개

이 영 동 (Young-Dong Lee)



2004년 2월 동서대학교
정보통신공학과(공학사)
2006년 2월 동서대학교
컴퓨터네트워크학과(공학석사)
2009년 2월 동서대학교
유비쿼터스IT학과(공학박사)
2010년 12월 핀란드 University of
Oulu 전기정보공학과(공학박사)
2012년 3월~현재 창신대학교
스마트융합공학부 컴퓨터전공
부교수
관심분야 : IoT, 무선센서네트워크,
임베디드시스템