

# MQTT Broker의 트래픽 측정을 위한 시스템에 관한 연구

김성진\* · 오창현\*

\*한국기술교육대학교

## A Study on System for Traffic Measurement of MQTT Broker

Sung-Jin Kim\* · Chang-Heon Oh\*

\*Korea University of Technology and Education(KOREATECH)

E-mail : tjdwls3527@koreatech.ac.kr

### 요 약

MQTT broker는 트래픽에 따라 네트워크 성능이 저하되어 패킷 손실, 지연 등의 문제가 발생한다. 하지만 MQTT broker는 트래픽 측정을 위한 별도의 인터페이스를 지원하지 않아 네트워크 성능 저하에 대처할 수 없다. 본 논문에서는 MQTT broker의 트래픽 측정을 위한 시스템을 제안한다. 제안하는 시스템은 jnetpcap 라이브러리를 사용하여 MQTT broker로 송수신되는 모든 트래픽을 측정한다.

### ABSTRACT

The MQTT broker has problems such as packet loss and delay due to degraded network performance according to traffic. However, the MQTT broker does not support a separate interface for traffic measurement, so it can not cope with network degradation. In this paper, we propose a system for traffic measurement of MQTT broker. The proposed system uses the jnetpcap library to measure all traffic to and from the MQTT broker.

### 키워드

MQTT, Traffic Measurement Module, Jnetpcap, Network Traffic, Broker, Publish/Subscribe

## I. 서 론

최근 IoT(Internet of Things) 기술의 발전으로 다양한 사물들이 인터넷에 연결되어 소통하고 있다. IDC(International Data Corporation) 보고서에 따르면 2020년에 약 220억 개의 IoT 사물들이 인터넷에 연결될 것으로 예측하고 있다. IoT 사물은 제한적인 네트워크 환경에서 동작하기 때문에 MQTT(Message Queue Telemetry Transport)와 같은 경량화 프로토콜이 필요하다 [1]. 하지만 MQTT broker로 송수신되는 트래픽에 따라 네트워크 성능이 저하되어 패킷 손실, 지연 등의 문제가 발생한다 [2]. 이러한 문제는 트래픽을 지속적으로 측정하여 네트워크 환경에 대한 분석 및 관리를 통해 해결할 수 있다 [3]. 하지만 MQTT broker는 트래픽을 측정하는 별도의 인터페이스를 제공하지 않기 때문에 네트워크 변화에 따른 관리 기능을 지원할 수 없다 [4]. 따라서 본 논문에서는 MQTT broker의 트래픽 측정 시스템을 제

안한다. 제안하는 시스템은 jnetpcap 라이브러리를 사용하여 MQTT broker로 송수신되는 모든 트래픽을 측정한다.

## II. MQTT

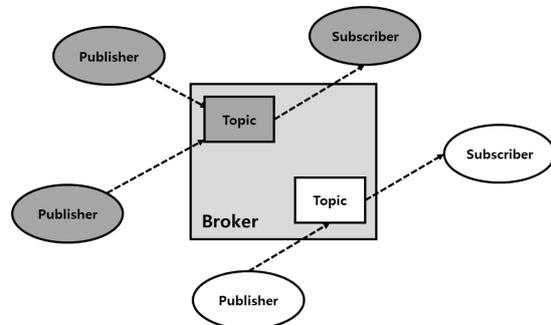


그림 1. MQTT publish/subscribe 구조

MQTT는 제한된 네트워크 환경에서 효율적인 메시지 전달을 위해 1999년 IBM에서 개발한 경량화 프로토콜이다. MQTT의 특징은 TCP/IP 기반으로 동작하며, 2byte의 고정 헤더로 오버헤드를 최소화한다. 또한 그림 1과 같이 publish/subscribe 구조를 가지며 publisher, subscriber, broker가 있다. Broker는 publisher와 subscriber 간의 메시지를 전달하는 중간 매개체 역할이다. Publisher는 broker로 특정 토픽에 대한 메시지를 전달한다. 해당 토픽을 구독한 subscriber는 broker를 통해 메시지를 수신한다.

네트워크의 상황은 데이터베이스에 저장된 패킷을 통해 확인할 수 있다. 그림 3은 데이터베이스의 테이블에 저장된 패킷에 대한 정보이다. 테이블은 number, time, source, destination, size로 구성된다. Number와 time 속성은 트래픽이 측정된 순서, 시간을 의미한다. Source와 destination 속성은 출발지와 목적지에 대한 IP와 port 정보이다. Size 속성은 패킷의 프레임 크기를 의미한다. 이러한 정보를 통해 broker로 송수신되는 publisher와 subscriber의 트래픽을 확인할 수 있다.

### III. MQTT broker의 트래픽 측정 시스템

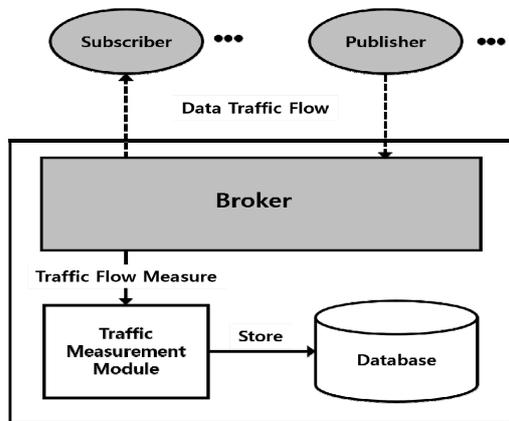


그림 2. MQTT broker의 트래픽 측정을 위한 시스템 구성도

그림 2는 MQTT broker의 트래픽 측정 시스템의 구성도이다. Publisher와 subscriber간의 통신에 대한 모든 트래픽 흐름은 broker를 측정하여 확인할 수 있다. Traffic measurement module은 jnetpcap 라이브러리를 사용하여 지속적으로 broker로 통과하는 모든 트래픽을 캡처하고, 서버의 데이터베이스에 저장한다.

number	time	source	destination	size
1	2017-04-29 14:23:23.974	220.68.65.121:65527	220.68.65.128:1883	1072
2	2017-04-29 14:23:24.11	220.68.65.121:65527	220.68.65.128:1883	2088
3	2017-04-29 14:23:24.115	220.68.65.128:1883	220.68.65.121:65527	56
4	2017-04-29 14:23:24.119	220.68.65.128:1883	220.68.65.121:65526	1516
5	2017-04-29 14:23:24.124	220.68.65.128:1883	220.68.65.121:65526	1516
6	2017-04-29 14:23:24.128	220.68.65.121:65526	220.68.65.128:1883	62
7	2017-04-29 14:23:24.132	220.68.65.128:1883	220.68.65.121:65526	184
8	2017-04-29 14:23:24.137	220.68.65.121:65526	220.68.65.128:1883	62
9	2017-04-29 14:23:24.142	220.68.65.121:65527	220.68.65.128:1883	1072
10	2017-04-29 14:23:24.146	220.68.65.128:1883	220.68.65.121:65526	1072
11	2017-04-29 14:23:24.151	220.68.65.121:65527	220.68.65.128:1883	2088
12	2017-04-29 14:23:24.155	220.68.65.128:1883	220.68.65.121:65527	56
13	2017-04-29 14:23:24.164	220.68.65.128:1883	220.68.65.121:65526	1516
14	2017-04-29 14:23:24.17	220.68.65.121:65526	220.68.65.128:1883	62
15	2017-04-29 14:23:24.176	220.68.65.128:1883	220.68.65.121:65526	628
16	2017-04-29 14:23:24.182	220.68.65.121:65527	220.68.65.128:1883	1072
17	2017-04-29 14:23:24.194	220.68.65.121:65527	220.68.65.128:1883	2088
18	2017-04-29 14:23:24.203	220.68.65.128:1883	220.68.65.121:65527	56
19	2017-04-29 14:23:24.211	220.68.65.128:1883	220.68.65.121:65526	1516
20	2017-04-29 14:23:24.218	220.68.65.128:1883	220.68.65.121:65526	1516
21	2017-04-29 14:23:24.224	220.68.65.121:65526	220.68.65.128:1883	62
22	2017-04-29 14:23:24.232	220.68.65.128:1883	220.68.65.121:65526	184
23	2017-04-29 14:23:24.239	220.68.65.121:65526	220.68.65.128:1883	62
24	2017-04-29 14:23:24.246	220.68.65.121:65527	220.68.65.128:1883	1072
25	2017-04-29 14:23:24.269	220.68.65.128:1883	220.68.65.121:65526	1072

그림 3. 측정된 트래픽 정보

### IV. 결론

본 논문에서는 MQTT broker의 트래픽 측정하는 시스템을 제안하였다. 제안하는 시스템은 jnetpcap 라이브러리를 사용하여 broker로 송수신되는 publisher와 subscriber의 트래픽을 측정하고, 네트워크 상황을 확인할 수 있다. 향후 시스템을 적용하였을 경우 MQTT broker에서 네트워크 상황에 따라 다양한 서비스 품질을 위한 서비스를 제공할 것으로 기대된다.

### 참고문헌

- [1] A. A. Fuaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things : a survey on enabling technologies, protocols, and applications," *IEEE COMMUNICATION SURVEYS & TUTORIALS*, vol. 17, no. 4, pp. 2347-2376, Jun. 2015.
- [2] S. H. Lee, H. W. Kim, D. K. Hong, and H. T. Ju, "Correlation Analysis of MQTT Loss and Delay According to QoS Level," *Information Networking (ICOIN), 2013 International Conference on*, pp. 714-717, 2013.
- [3] S. Habib, M. A. Qadeer, and A. Y. Javaid, "Authentication Based QoS Using Bandwidth Limitation," *International Symposium on Computer Science and its Applications*, pp. 7-12, Oct. 2008.
- [4] A. A. Fuqaha, A. Khreishah, M. Guizani, A. Rayes, and M. Mohammadi, "Toward better horizontal integration among IoT services," *IEEE Communications Magazine*, vol. 53, no. 9, pp. 72-79, Sep. 2015.