

## The MI-SDN System to Manage MQTT Data in an Interoperable IoT Wireless Network

Rajae Tamri<sup>a</sup>, Said Rakrak<sup>b</sup>

<sup>a</sup>Laboratory of Computer and Systems Engineering (L2IS), Cadi Ayyad University, Marrakesh, Morocco.  
E-mail: rajae.tamri@gmail.com

<sup>b</sup>Laboratory of Computer and Systems Engineering (L2IS), Cadi Ayyad University, Marrakesh, Morocco.  
E-mail: s.rakrak@uca.ma

**Article History:** Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

**Abstract:** The main challenge for the Internet of Things (IoT) is to ensure interoperability between heterogeneous IoT entities. To support the interaction, intercommunication, and interoperability between these devices several solutions are proposed in the literature. The SDN (Software-defined Network) is one of these solutions to resolve the problem of the heterogeneous network used in IoT. To guarantee network interoperability, the SDN uses a centralized controller, which handles the entire network. The role of end devices in IoT is only forwarding data. The MQTT (Message Queuing Telemetry Transport) protocol is another solution for granting interoperability in IoT. Which is a publish/subscribe based messaging protocol that avoids direct connection between devices by relaying data through a central server called the broker. Combination of these two solutions to manage IoT devices makes it easy to add new devices without touching or changing the existing infrastructure. The new devices only need to communicate with the broker. Moreover, the Controller SDN is responsible for handling networks. Consequently, smart devices added don't need to be compatible with the others. In this paper, we present the design and the implementation of a new IoT architecture, which is a combination of SDN technology and MQTT protocol. That enables heterogeneous IoT devices to be interoperable and interact without any problems. Our system utilizes the lightweight protocol MQTT with a new mechanism using several slave brokers and one master. The slaves manage the group of the end devices in the wireless IoT network, and the master broker installed in the SDN controller supervises the integral network. The SDN controller uses a multicast system to send MQTT data across the external wireless network. As a result, that reduces transmission delay between wireless IoT network compared with the using of a standard MQTT.

**Keywords:** Internet of Things, Interoperability, Network Interoperability, SDN, Wireless Network.

### 1. Introduction

Basically, in the Internet of Things, a huge number of smart connected objects and a massive data amount are produced increases daily. However, with this hyper-connectivity, to handle the different heterogeneous networks and to manage the big variety of data in IoT is becoming more and more difficult. Which creates many challenges to support interoperability (Noura *et al.*, no date; Noura, 2019) with this high degree of heterogeneity. As a solution to ensure the interoperability at the data and network level several solutions are developed in the literature. For network interoperability, the researchers invested SDN (Bekri, Jmal and Chaari, 2020) (Mathews *et al.*, 2018) (Tayyaba and Ahmed, 2017) paradigm as an effective solution to manage IoT network systems. Which is used to handle several heterogeneous networks through separation the data plane of the control plane. The SDN support several network requirements. Which allows a flexible management of the network, with ensuring different functionality such as:

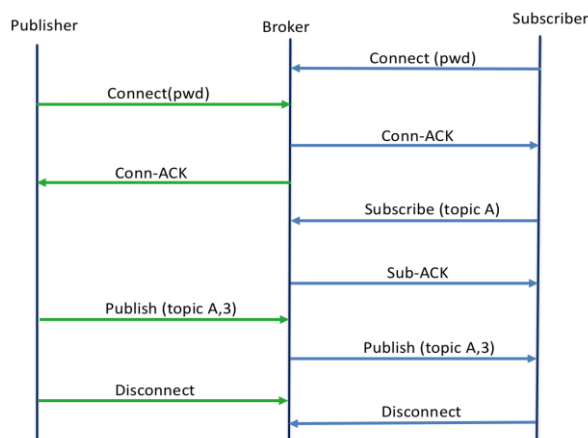


Figure 1. Exchange of Messages in MQTT Protocol

- Decouples and separates the Control plane of the data plane.
- Contains a centralized controller and global view of the network.
- Use the southbound interfaces that is the open interfaces (e.g. using Open flow protocol) for Communication between control plane and the data plane.
- Use the northbound interfaces with various APIs to support the programmability feature of the network.

Moreover for scalability and heterogeneity challenges of the data generated by the sensing things in the IoT system the MQTT(Zabasta *et al.*, 2018) is a solution to solve these challenges and guarantee the interoperability of data(Pramukantoro *et al.*, 2018), that ensure exchanging data between heterogeneous smart object, and avoid a direct communication between them. MQTT is a lightweight publish/subscribe protocol allows many-to-many transmission, with QoS support, Based on TCP/IP. It's designed around a central broker installed on any public server. The connection is established between the broker and clients before communication. A username and a password can be configured for securing the connection. The MQTT clients connected to the broker can publish data or subscribe to any specific topic into the broker. Its main goal is to provide reliable service in a network environment contains constrained resources. Three different roles are involved in an MQTT system(Samer, 2019), namely publisher, broker, and subscriber, as shown in Figure. 1.Publishers are sources of data and information. They send messages to the broker that sends them to subscribed subscribers.

The MQTT provides three levels of Quality of Service (QoS), which is an arrangement between the sender and receiver to guarantee of delivering a message. The three levels of QoS are:

- 0 - at most once: no response is delivered by the receiver and the sender does no retry.
- 1 - at least once: ensures that the receiver receives the message at least once.
- 2 - exactly once: This is the highest quality of service, there's no loss or duplication of the message.

In MQTT, some existing solutions (e.g. Mosquitto, HiveMQ) allow direct connection between brokers using bridge connection, i.e. broker A connected to another broker B as a standard client. That allows broker A to subscribe for all or a subset of the topics published by B. That creates a looping topic and traffic congestion problem [5]. To solve this problem, in our proposal, we will introduce SDN to handle communications between multiple brokers, which avoids any direct connection between brokers. Our work describes the transmission of data between several distributed wireless networks using the MQTT protocol and SDN to manage and manipulate the set of IoT devices, also, which allows solving the problem of data and network inter operability.

This paper is presented as follows. Section 2 describes and analyses the architecture of our system, in this section, we present the materials and parameters used for simulation, also the performance evaluation, moreover, we present our results. Finally, we devote Section 3 for some conclusions and future work.

## 2. Proposal System

### 2.1. Motivation

The authors in [1] presented the advantages of IoT based SDN and its main management functions in different IoT environments. In(Park, Kim and Kim, 2018)SDN controller is used to collect information on the customers and related pub/sub topics from all edge slave brokers through a master broker, that acts as root of the multicast tree. SDN establish multicast per-topic groups to reduce data transfer delay. However, this contribution is adopted in a static wired topology. But in our proposed architecture, we assume a wireless topology in a IoT network, and the SDN controller set a system multicast of MQTT data.

Our proposed system manages all pub/sub wireless nodes in the different wireless networks by master and slave brokers in the distributed networks using SDN. The master broker receives the IP addresses, and pub/sub topics trough slave brokers connected to the wireless nodes from different networks and send them to the SDN controller. The SDN controller applies an MI-SDN (Multicast information-SDN) system by using the information received from the master broker to constitute a multicast table in consideration of the topic to set a MQTT data transmission path. MI-SDN uses a multicast system that minimizes data transfer delays between pub/sub wireless devices in distributed IoT networks, and avoids direct connection between slave brokers.

### 2.2. Architecture Overview

Figure 2 shows the structure of the proposed system The system consists of an SDN controller with a MQTT Master broker, different wireless IoT network managed by MQTT slave broker, which is connected to the

network through access points (AP) with a wired connection. The wireless IoT devices are connected to the AP. To collect data and information from each network the AP uses a slave broker. Network information contains the IP address, and topic.

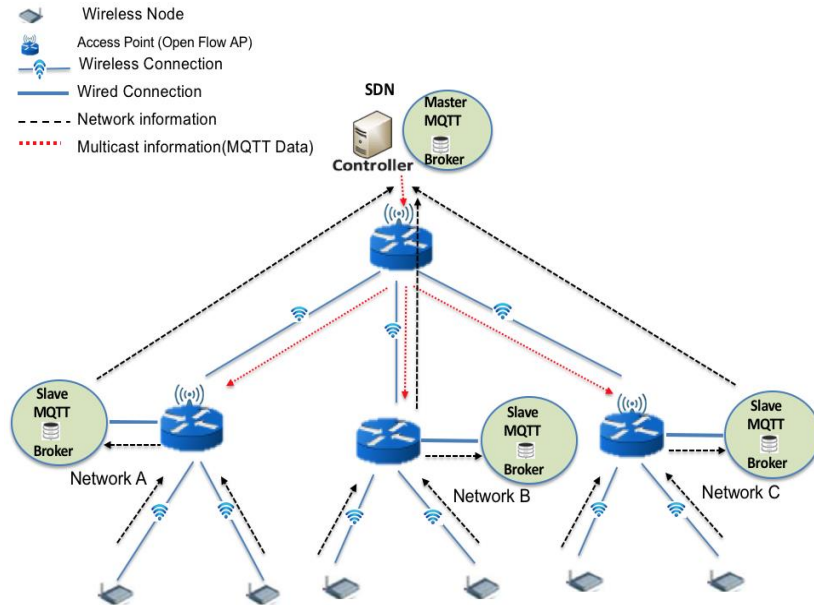


Figure 2. The MI-SDN Architecture

The master broker sends network information from all slave brokers to the SDN through the wireless AP. The master broker does not run the normal MQTT broker feature for connecting and exchanging data with the wireless IoT nodes of the different wireless networks. The SDN controller analyses the information of each network, creates a suitable table to set paths between nodes from different networks and create a MI-SDN system for exchanging MQTT data between networks. When multiple subscribers subscribe to the same topic, the SDN controller configures and generates a multicast path. This path is forwarded to the AP, and the MQTT data of the specific network is transmitted along the specified path to the other network. The SDN controller manages multicast groups and trees quickly, in order to exchange data between IoT devices effectively. The master broker and the SDN are configured on the same server to reduce the data transfer, but there are logically separated. At the result, the SDN-based multicast system is used for Interaction and communication between external networks, and for any internal network data transmission a standard MQTT protocol is applied.

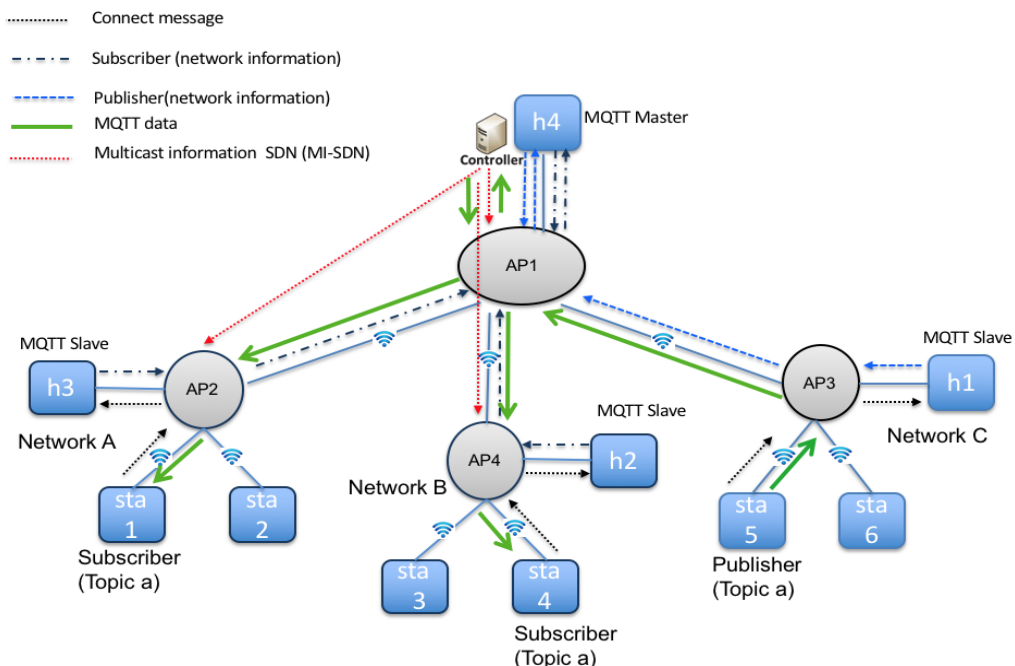


Figure 3. The Process Subscribing to MQTT Data on an External Network

The operation process subscribing to data on an external wireless network is depicted in Figure 3. The following describes the order for the transmission of data to the external network shown in Figure 3:

- The subscriber in sta1 and sta4 forward connection message that subscribes topic (a) to slave brokers h3 and h2 in their network.
- Slave brokers h3 and h2 send network information, including subscription information, to master broker h4.
- The master broker h4 sends subscriber network information to the SDN controller.
- The publisher in sta5 sets topic (a) as a priority message and sends a connection message to their slave broker h1.
- The slave broker h1 sends the network information including the topic (a) publications (MQTT data) to the master broker.
- The master broker sends publisher network information and MQTT data to the SDN controller.
- The SDN controller creates multicast group of topic (a) (MQTT data) and sends it to the different Aps (AP1, AP2 and AP4) on the network.
- Data published from sta5 are sent at the same time to sta4 and sta1 using the MI-SDN of MQTT data (topic (a)).

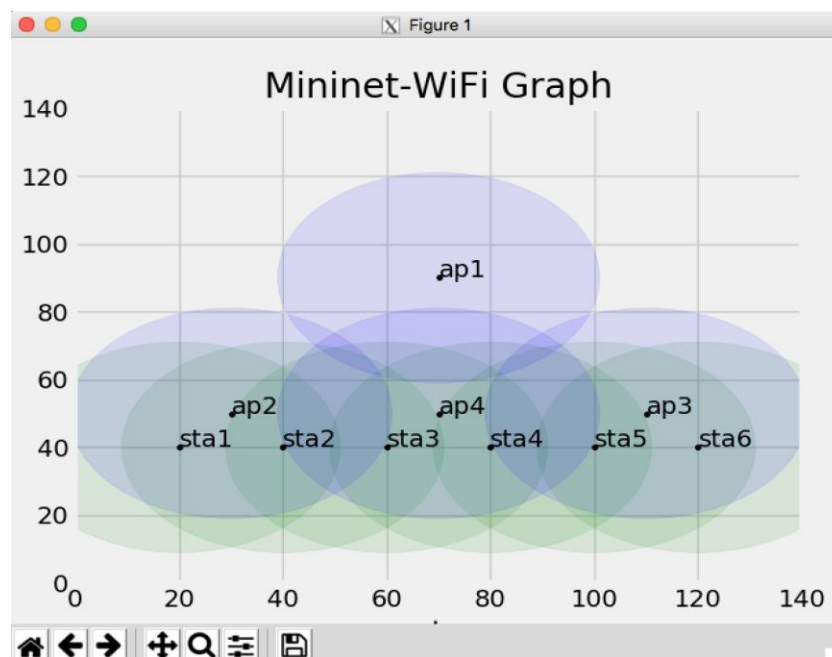
**Table 1.** Simulation Parameters

Number of Nodes (stations)	6
Communication	IEEE 802.11
Distance between APs	40m
Estimated AP range	30m

### 2.3. Materials and Parameters

The Mininet-Wifi emulator (Fontes *et al.*, 2015) is used for deploying WiFi stations and access points. Mininet-WiFi is a tool for emulating wireless OpenFlow / SDN scenarios that enable high-fidelity experiments to replicate real networking environments. The mininet-wifi is used to simulate the process for subscribing to data on an external network. The SDN controller uses the OpenDaylight and the Mosquitto broker is used for the master broker and the slave broker. Figure 3 illustrates the topology and the scenario used for simulation in mininet-wifi is illustrated in Figure 4. The parameters used in the simulation are presented in Table 1.

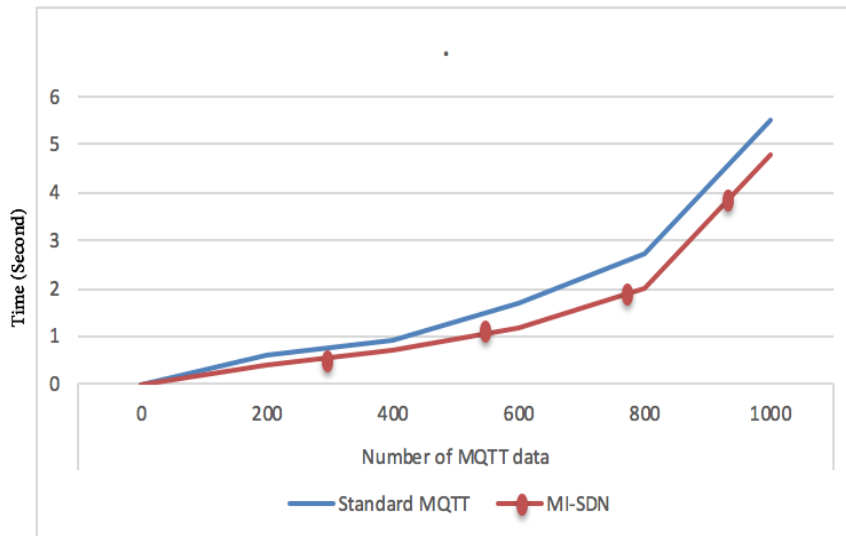
### 2.4. Performance Evaluation and Results



**Figure 4.** Simulation Scenario with Several Distributed Wireless Network Using MI-SDN

In this paper, we verify the performance of MI-SDN by evaluating data transmission delay between several external wireless networks. The experimental results are compared with MI-SDN and the standard MQTT

mechanism. The experiment evaluates the delay of transmission MQTT data while increasing the data for transmission. The number of transmitted data is increased by 200 per experiment for the publisher. At the end of the experiment, the publisher increases the amount of data sent to 1000 data per second.



**Figure 5.** Evaluations of the Data Transfer Delay Depending on the Number of MQTT Data

The result of the evaluation of the MQTT data transmission delay between different external networks is shown in Figure 5. The MI-SDN uses a multicast system to transmit MQTT data at the same time, which reduces the data transmission delay between external wireless networks compared to the standard MQTT which transmits the data to each subscriber, which creates more delay.

### 3. Conclusions and Future Work

IoT is an evolutionary technology, allows interconnecting and interaction between smart things using various technologies. That creates a high degree of heterogeneity, which brings many challenges to ensure interoperability. To solve the problem of interoperability, in our system we combine SDN technology and MQTT protocol to ensuring two levels of interoperability such as network and exchanging data. Consequently, It's easy to add new devices without touching the existing infrastructure of the network. The SDN handles the entire network, and the new devices only need to communicate with the broker, they don't need to be compatible with the other devices. However, in the MQTT protocol, a problem of looping topics and traffic congestion can be created. To solve that we propose a new system multicast-based SDN (MI-SDN) to exchange MQTT data between several external wireless networks, which avoids direct connection between devices. As a result, managing MQTT data between external wireless networks using the MI-SDN system reduce data transfer delays compared to standard MQTT.

As future work, we aim to apply and evaluate our system in a wireless network with mobile nodes taking into consideration the QoS of MQTT.

### References

1. Bekri, W., Jmal, R., & Chaari Fourati, L. (2020). Internet of things management based on software defined networking: a survey. *International Journal of Wireless Information Networks*, 27, 385-410.
2. Fontes, R.R., Afzal, S., Brito, S.H., Santos, M.A., & Rothenberg, C.E. (2015). Mininet-WiFi: Emulating software-defined wireless networks. In *11th International Conference on Network and Service Management (CNSM)*, 384-389.
3. Modieginyane, K.M., Letswamotse, B.B., Malekian, R., & Abu-Mahfouz, A.M. (2018). Software defined wireless sensor networks application opportunities for efficient network management: A survey. *Computers & Electrical Engineering*, 66, 274-287.
4. Noura, M., Atiquzzaman, M., & Gaedke, M. (2019). Interoperability in internet of things: Taxonomies and open challenges. *Mobile Networks and Applications*, 24(3), 796-809.
5. Noura, M., Atiquzzaman, M., & Gaedke, M. (2017). Interoperability in internet of things infrastructure: Classification, challenges, and future work. In *International Conference on Internet of Things as a*

- Service, Springer, Cham, 11-18.
6. Park, J.H., Kim, H S., & Kim, W.T. (2018). Dm-mqtt: An efficient mqtt based on sdn multicast for massive iot communications. *Sensors*, 18(9), 3071.
  7. Pramukantoro, E.S., Bakhtiar, F.A., Aji, B., & Pratama, R. (2018). Middleware for network interoperability in IoT. In *5th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 499-502
  8. Jaloudi, S. (2019). MQTT for IoT-based applications in smart cities. *Palestinian Journal of Technology and Applied Sciences (PJTAS)*, (2).
  9. Tayyaba, S.K., Shah, M.A., Khan, O.A., & Ahmed, A.W. (2017). Software defined network (sdn) based internet of things (iot) a road ahead. In *Proceedings of the International Conference on Future Networks and Distributed Systems*, 1-8.
  10. Zabasta, A., Kunicina, N., Kondratjevs, K., Patlins, A., Ribickis, L., & Delsing, J. (2018). MQTT service broker for enabling the interoperability of smart city systems. In *Energy and Sustainability for Small Developing Economies (ES2DE)*, 1-6