TUTORIAL 4: Using movement sensors with Zigbee2Mqtt

Last revision: March 28th, 2022

1 INTRODUCTION & MOTIVATION

Pervasive projects often needs to be able to monitor sensors and control actuators. These include movement sensors of the type PIR (Passive Infrared) which are suitable for detecting human movement. These movement sensors are well-known from burglar alarms and home automation.

In this tutorial, we will look into two popular Zigbee motion sensors: the Xiaomi Aqara RTCGQ11LM and the IKEA E1525/E1745. You will use Zigbee2Mqtt (Z2M) to interface them.

2 WHAT TO READ BEFORE THE TUTORIAL

It is expected that you have completed tutorial 3, i.e. Zigbee2Mqtt (Z2M) is running and you are able to interact with it, either via the Mosquitto clients, MQTT Explorer or your custom Python clients. You should also have a basic knowledge of what a Zigbee network is. More information about joining devices to a Zigbee networks can be found in [1, 2].

3 MATERIALS

You'll need the following materials:

- Raspberry Pi 4
- HDMI to mini-HDMI cable
- Power supply
- Keyboard, mouse and monitor (any keyboard and mouse should work)
- Zigbee controller USB stick
- Xiaomi Aqara RTCGQ11LM
- IKEA E1525/E1745

Figure 1. Xiaomi Aqara RTCGQ11LM (left) and IKEA E1525/E1745 (right).

4 JOINING PIR SENSORS TO THE ZIGBEE NETWORK

The first step for monitoring and/or controlling devices is to join them to a Zigbee network. In this tutorial, the basic steps for joining a device are presented. These should be enough, but if you have problems joining the devices then refer to [3].

When joining new devices to the Zigbee network using Z2M, the process can be monitored through its logs. If Z2M was started manually, then you should see the log messages being output to the shell. If it was configured to run as a service, then you can open the logs located in the directory `/opt/zigbee2mqtt/data/log`[1]. The log messages are also published to the topic `zigbee2mqtt/bridge/logging`, so this process can also be monitored remotely through clients such as MQTT Explorer or Mosquitto clients. A sample log with the joining process of an IKEA's PIR sensor is depicted in appendix A..

In this project the devices will join to a network created by the Zigbee USB adapter, which will act as the network coordinator [2]. The instructions for joining Zigbee devices via Z2M are available in [4], along with troubleshooting, device characteristics and exposed values (these might vary with the device). You should refer to these anytime you need the characteristics of the device. Here, only the instructions for joining the described motion sensors are reproduced:

- **Xiaomi Aqara:** press and hold the reset button on the side of the device for about 5 seconds until the blue light starts blinking;
- **IKEA:** press the pair button 4 times in a row until a red light starts blinking in the front of the device. The pair button is the middle one, with a chain icon.

Once the joining process is started you should see in the logs a message similar to "`Starting interview of '0x14b457fffe7c72fc'`". The value `0x14b457fffe7c72fc` is the device's Zigbee IEEE address and it should be unique for each device – you can think of it as being the IP address of a device on an IP network.

When the device is successfully joined, you should see a message similar to "`Successfully interviewed '0x14b457fffe7c72fc', device has successfully been paired`". After this, Z2M will publish an updated list with all devices that are part of the network. This list can be used by other clients to know if there were any modifications to the Zigbee network (for example, devices added or removed).

Finally, the device will automatically publish its events to its respective topic. The format of the topic is `zigbee2mqtt/<friendly_name>`. For example, the device above will publish events to the topic `zigbee2mqtt/0x14b457fffe7c72fc`, which is the default friendly name attributed by Z2M to the device when, it joins the network[2]. In Figure 2 it is depicted the

---

[1] A new log directory is created with the date Z2M was last started, with a log.txt file inside with the actual log messages. An example directory is 2021-02-04.10-58-05. Note that the logs are always written, either if Z2M is started manually or as a service.

presence events for three motion sensors in MQTT Explorer. Note the `occupancy` field in each of the messages.
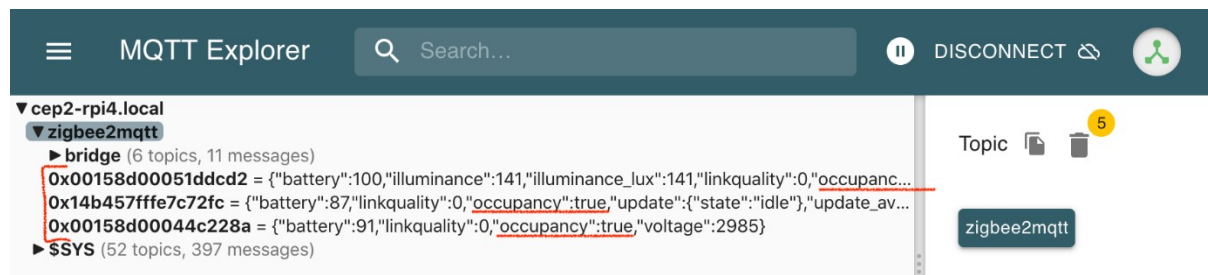

Figure 2. Presence events received in MQTT Explorer.

After this process, the device's address and friendly name are stored in Z2M's `configuration.yaml` file. Follows an example for the devices above:

```
homeassistant: false
permit_join: true
mqtt:
  base_topic: zigbee2mqtt
  server: 'mqtt://localhost'
serial:
  port: /dev/ttyACM0
devices:
  '0x00158d00044c228a':
    friendly_name: '0x00158d00044c228a'
  '0x14b457fffe7c72fc':
    friendly_name: '0x14b457fffe7c72fc'
  '0x00158d00051ddcd2':
    friendly_name: '0x00158d00051ddcd2'
```

In the configuration file it is possible to instruct Z2M to allow new devices to join the network. For that, you can use the `permit_join` option. By default this is set to `true`, i.e. allow new devices to join. Z2M advises to set this option to `false` once the pairing process is done for all devices, so that other Zigbee devices might not inadvertently join the network. This is especially important if more than one Zigbee network is near.

5 NEXT STEPS

5.1 Topics processing

As depicted in appendix A., while new devices are joining the network, Z2M is publishing the log messages to its specific topic, as well as event messages to `zigbee2mqtt/bridge/event`. The last can be used by remote applications to monitor the current state of the pairing process. In the tutorial only joining devices was presented, but devices can also leave the network. Refer to [5] for what type of events are published to this topic.

---

[2] The friendly name is a Z2M feature; it is not part of the Zigbee standard. It can be changed to any string that suits the needs of the user. For example, the device can be named to "IkeaPirSensor" to help the user know which device is publishing events. Beware of white spaces; its use is not advisable in topics.

5.2 Joining process

Now that you are able to add devices, you should consider how this process should be handled in your project: will it be handled through Z2M, as presented, or will it be handled through your application? Who will do it: your team or the final user? What are the advantages/disadvantages of each approach? Bear in mind that although this is not something expected to be made often, it is has to be done at least once.

5.3 Python application

If you already started working in your application, how would you adapt it for receiving devices events? How do you discover the device's topics? Once you receive these events, how would you handle them: parsing JSON messages and actuate?

6 REFERENCES

[1] Joining Zigbee networks:
https://www.digi.com/resources/documentation/Digidocs/90001399-13/references/r-create-join-zigbee-network.htm

[2] Zigbee2Mqtt's Zigbee network:
https://www.zigbee2mqtt.io/information/zigbee_network.html

[3] Zigbee2Mqtt's FAQ "Why does my device not or fail to pair?":
https://www.zigbee2mqtt.io/guide/faq/#why-does-my-device-not-or-fail-to-pair

[4] Zigbee2Mqtt's Supported devices: https://www.zigbee2mqtt.io/supported-devices/

[5] Zigbee2Mqtt's bridge events:
https://www.zigbee2mqtt.io/information/mqtt_topics_and_message_structure.html#zigbee2mqttbridgeevent

A.   APPENDIX

In this appendix it is presented a sample Z2M log for the pairing process of an IKEA motion sensor device.