python mqtt script on raspberry pi to send and receive messages

stackoverflow.com/questions/37006863/python-mqtt-script-on-raspberry-pi-to-send-and-receive-messages

5

MQTT question:

Hi, I’m trying to set up a MQTT network between multiple Raspberry Pis (starting with two). I have one raspberry pi (RPi-A), MQTT client, with a thermistor sensor attached and one raspberry (RPi-B), MQTT broker/client, acting as a hub for my network. Through python scripting I’d like the temperature to be sent every 30mins from RPi-A via MQTT to topic sensor/data and received by RPi-B. When RPi-B receives a message from RPi-A via topic sensor/data, I want it to respond with an instruction via MQTT topic sensor/instructions to RPi-A. Below is my script, so far RPi-A can send messages and RPi-B receive them but I cannot work out how RPi-B can respond.

Basically, what I’m trying to understand is, is it possible for a MQTT device to act as both broker and client at the same time? And, can a client both send and receive messages and if so how to implement all the above via python? I’ve read a lot of blogs, official MQTT articles and the paho module documentation (which for me is very hard to fathom) but still cannot figure this out. Your help would be most useful/ appreciated.

Code RPi-A (with thermistor sensor):

```
from sense_hat import SenseHat
import time
import paho.mqtt.client as mqtt
import paho.mqtt.publish as publish
sense = SenseHat()

Broker = "192.168.1.252"

sub_topic = "sensor/instructions"    # receive messages on this topic

pub_topic = "sensor/data"       # send messages to this topic


############### sensehat inputs ##################

def read_temp():
    t = sense.get_temperature()
    t = round(t)
    return t

def read_humidity():
    h = sense.get_humidity()
    h = round(h)
    return h

def read_pressure():
    p = sense.get_pressure()
    p = round(p)
    return p

def display_sensehat(message):
    sense.show_message(message)
    time.sleep(10)

############### MQTT section ##################

# when connecting to mqtt do this;

def on_connect(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))
    client.subscribe(sub_topic)

# when receiving a mqtt message do this;

def on_message(client, userdata, msg):
    message = str(msg.payload)
    print(msg.topic+" "+message)
    display_sensehat(message)

def publish_mqtt(sensor_data):
    mqttc = mqtt.Client("python_pub")
    mqttc.connect(Broker, 1883)
    mqttc.publish(pub_topic, sensor_data)
    #mqttc.loop(2) //timeout = 2s

def on_publish(mosq, obj, mid):
    print("mid: " + str(mid))
```

```
client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message
client.connect(Broker, 1883, 60)


while True:
    sensor_data = [read_temp(), read_humidity(), read_pressure()]
    publish.single("monto/solar/sensors", str(sensor_data), hostname = Broker)
    time.sleep(1*60)
```

Code RPi-B (network hub):

```
import time
import paho.mqtt.client as mqtt
import paho.mqtt.publish as publish

Broker = "192.168.1.252"

sub_topic = "sensor/data"    # receive messages on this topic

pub_topic = "sensor/instructions"               # send messages to this topic


# mqtt section

# when connecting to mqtt do this;

def on_connect(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))
    client.subscribe(sub_topic)

# when receiving a mqtt message do this;

def on_message(client, userdata, msg):
    message = str(msg.payload)
    print(msg.topic+" "+message)
    publish_mqtt(‘got your message’)

# to send a message

def publish_mqtt(sensor_data):
    mqttc = mqtt.Client("monto_hub")
    mqttc.connect(Broker, 1883)
    mqttc.publish(pub_topic, "this is the master speaking")
    #mqttc.loop(2) //timeout = 2s

def on_publish(mosq, obj, mid):
    print("mid: " + str(mid))


client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message
client.connect(Broker, 1883, 60)
client.loop_forever()
```

python raspberry-pi mqtt paho

edited May 3 '16 at 14:33

asked May 3 '16 at 14:24

Tullio_IRL
8911 gold badge22 silver badges99 bronze badges

- What actual error are you seeing? The code looks about right (RPI-A doesn't need any of the MQTT client code before the loop as you are using `publish.single`) and the RPI-B code looks ok at a first glance. – hardillb May 3 '16 at 14:48

- @hardillb thanks for responding, I want RPi-A to receive mqtt messages from RPi-B as well as send them so I thought I'd need the client-mqtt code. I don't get an error per se but RPi-B does not seem to send messages in response to RPi-A. – Tullio_IRL May 3 '16 at 15:10

- Sorry missed that bit. Answer provided – hardillb May 3 '16 at 15:19

add a comment |

1 Answer

Active Oldest Votes
8

✓

The simplest way is to start the network loop on a separate thread using the `client.loop_start()` function, then use the normal `client.publish` method

```
from sense_hat import SenseHat
import time
import paho.mqtt.client as mqtt
import paho.mqtt.publish as publish
sense = SenseHat()

Broker = "192.168.1.252"

sub_topic = "sensor/instructions"    # receive messages on this topic

pub_topic = "sensor/data"       # send messages to this topic


############### sensehat inputs ##################

def read_temp():
    t = sense.get_temperature()
    t = round(t)
    return t

def read_humidity():
    h = sense.get_humidity()
    h = round(h)
    return h

def read_pressure():
    p = sense.get_pressure()
    p = round(p)
    return p

def display_sensehat(message):
    sense.show_message(message)
    time.sleep(10)

############### MQTT section ##################

# when connecting to mqtt do this;

def on_connect(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))
    client.subscribe(sub_topic)

# when receiving a mqtt message do this;

def on_message(client, userdata, msg):
    message = str(msg.payload)
    print(msg.topic+" "+message)
    display_sensehat(message)

def on_publish(mosq, obj, mid):
    print("mid: " + str(mid))


client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message
client.connect(Broker, 1883, 60)
client.loop_start()
```

```
while True:
    sensor_data = [read_temp(), read_humidity(), read_pressure()]
    client.publish("monto/solar/sensors", str(sensor_data))
    time.sleep(1*60)
```

answered May 3 '16 at 15:20

hardillb
38.5k99 gold badges4747 silver badges7373 bronze badges

- Super stuff, that worked a treat :) thank you very much.<br/> Should I even be using the client.loop_forever() on RPi-B or should I replace it with client.loop_start() ? – Tullio_IRL May 3 '16 at 15:46 ✎
- Thank you so much for this ans. This worked for me as well – Tia May 27 '18 at 12:51

add a comment |

Your Answer

Sign up or log in

Sign up using Google
Sign up using Facebook
Sign up using Email and Password

Post as a guest

Required, but never shown

By clicking “Post Your Answer”, you agree to our terms of service, privacy policy and cookie policy

Not the answer you're looking for? Browse other questions tagged python raspberry-pi mqtt paho or ask your own question.