Testing Mosquitto Broker and Client on Raspbbery Pi

In this post you're going to test the Mosquitto Broker and Client on a Raspberry Pi by subscribing to an MQTT topic and publishing sample messages.

Recommended resources:

- You need a Raspberry Pi board – read Best Raspberry Pi Starter Kits
- How to Install Mosquitto Broker on Raspberry Pi
- What is MQTT and How It Works
- Getting Started with Node-RED on Raspberry Pi

If you like home automation and you want to build a complete home automation system, I recommend downloading my home automation course.

Testing MQTT Broker Installation

After installing MQTT Broker, I recommend installing an MQTT Cllient to test the Broker installation and publish sample messages.

The next command shows how to install MQTT Mosquitto Client:

You'll have to type **Y** and press **Enter** to confirm the installation.

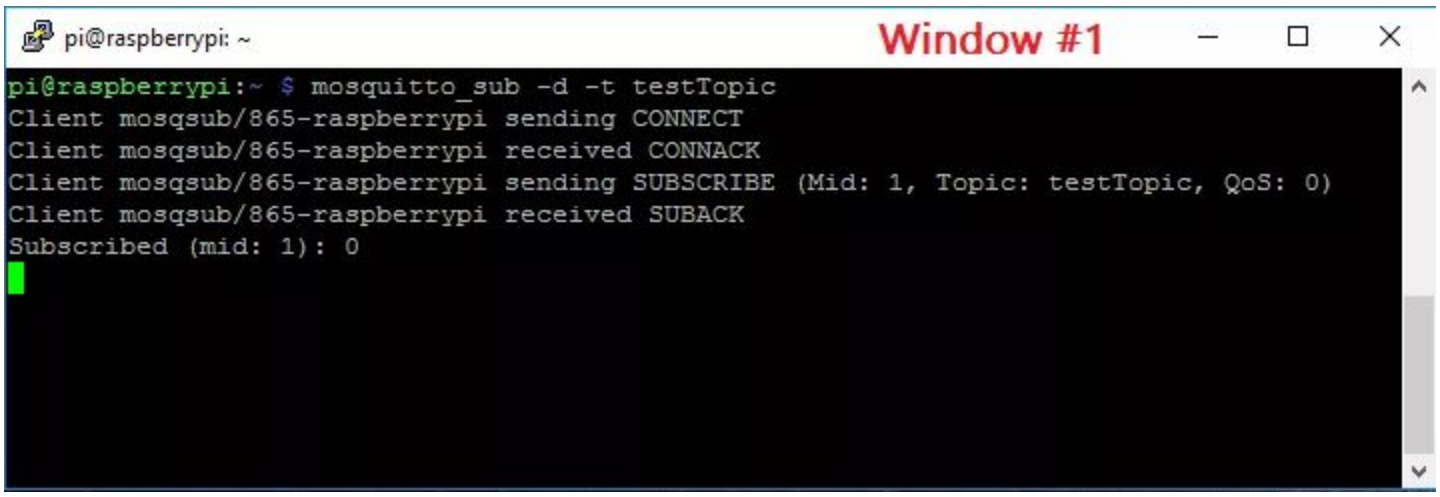
Run Mosquitto on background as a daemon:

```
pi@raspberry:~ $ mosquitto -d
```

Subscribing to testTopic Topic

To subscribe to an MQTT topic with Mosquitto Client open a terminal Window #1 and enter the command:

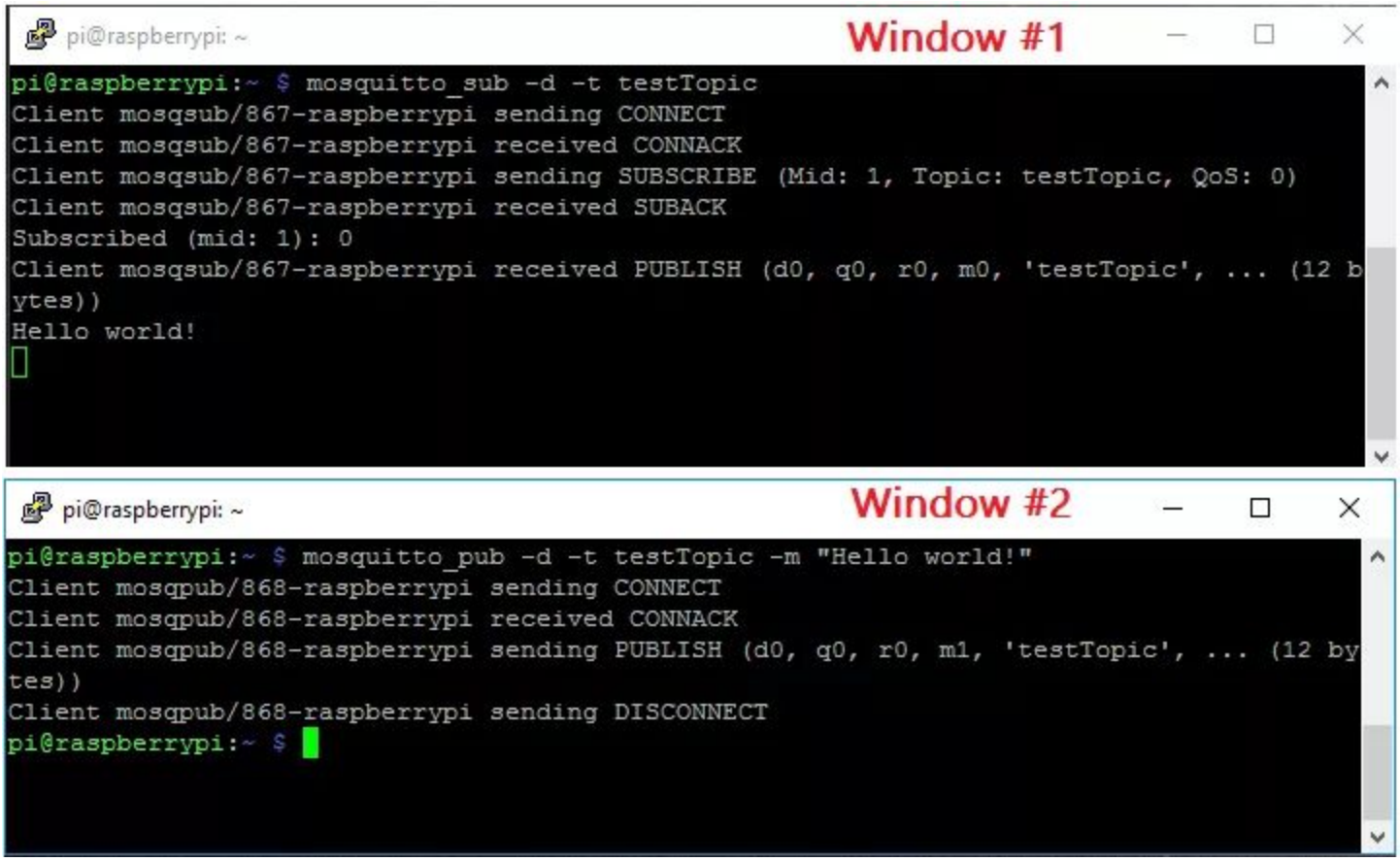
```
pi@raspberry:~ $ mosquitto_sub -d -t testTopic
```



You're now subscribed to a topic called **testTopic.**

To publish a sample message to testTopic, open a terminal Window #2 and run this command:

```
pi@raspberry:~ $ mosquitto_pub -d -t testTopic -m "Hello world!"
```



The message "**Hello World!**" is received in Window #1 as illustrated in the figure above.

Publishing a Message to Multiple Clients

Having Window #1 still subscribed to topic testTopic, open a new terminal Window #3 and run this command to subscribe to **testTopic** topic:

```
pi@raspberry:~ $ mosquitto_sub -d -t testTopic
```

On Window #2 publish the "**Hello World!**" message:

```
pi@raspberry:~ $ mosquitto_pub -d -t testTopic -m "Hello world!"
```

```
Client mosqsub/919-raspberrypi received CONNACK
Client mosqsub/919-raspberrypi sending SUBSCRIBE (Mid: 1, Topic: testTopic, QoS: 0)
Client mosqsub/919-raspberrypi received SUBACK
Subscribed (mid: 1): 0
Client mosqsub/919-raspberrypi received PUBLISH (d0, q0, r0, m0, 'testTopic', ... (12 b
ytes))
Hello world!
Client mosqsub/919-raspberrypi received PUBLISH (d0, q0, r0, m0, 'testTopic', ... (12 b
ytes))
Hello world!
```

pi@raspberrypi: ~ Window #2

```
pi@raspberrypi:~ $ mosquitto_pub -d -t testTopic -m "Hello world!"
Client mosqpub/920-raspberrypi sending CONNECT
Client mosqpub/920-raspberrypi received CONNACK
Client mosqpub/920-raspberrypi sending PUBLISH (d0, q0, r0, m1, 'testTopic', ... (12 by
tes))
Client mosqpub/920-raspberrypi sending DISCONNECT
pi@raspberrypi:~ $ mosquitto_pub -d -t testTopic -m "Hello world!"
Client mosqpub/922-raspberrypi sending CONNECT
Client mosqpub/922-raspberrypi received CONNACK
Client mosqpub/922-raspberrypi sending PUBLISH (d0, q0, r0, m1, 'testTopic', ... (12 by
tes))
Client mosqpub/922-raspberrypi sending DISCONNECT
pi@raspberrypi:~ $
```

pi@raspberrypi: ~ Window #3

```
pi@raspberrypi:~ $ mosquitto_sub -d -t testTopic
Client mosqsub/921-raspberrypi sending CONNECT
Client mosqsub/921-raspberrypi received CONNACK
Client mosqsub/921-raspberrypi sending SUBSCRIBE (Mid: 1, Topic: testTopic, QoS: 0)
Client mosqsub/921-raspberrypi received SUBACK
Subscribed (mid: 1): 0
Client mosqsub/921-raspberrypi received PUBLISH (d0, q0, r0, m0, 'testTopic', ... (12 b
ytes))
Hello world!
```

Since two clients are subscribed to **testTopic** topic, they will both receive **“Hello world!”** message.

This simple example shows how MQTT works and how your devices (for example: ESP8266) could be subscribed to the same topic to receive messages or a device could publish messages to multiple devices. We'll explore this concept further in future blog posts.

Wrapping up

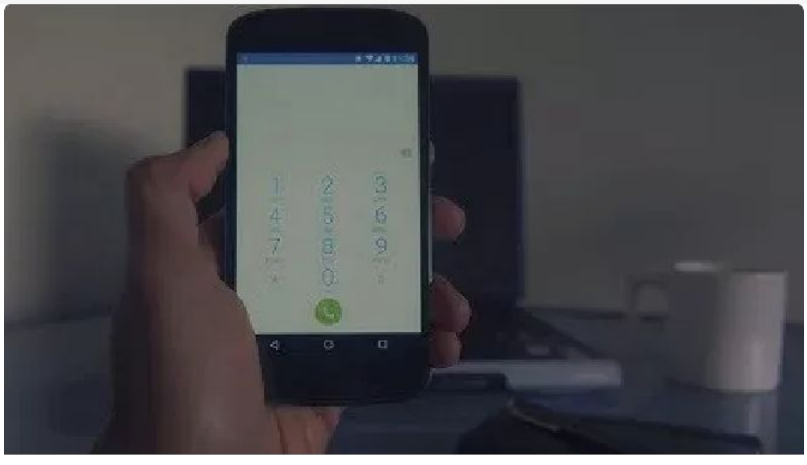
In the next blog post, we will experiment with MQTT and ESP8266 to see how everything works with practical examples.

Like home automation? Learn more about Node-RED, Raspberry Pi, ESP8266 and Arduino with my course: Build a Home Automation System for $100.

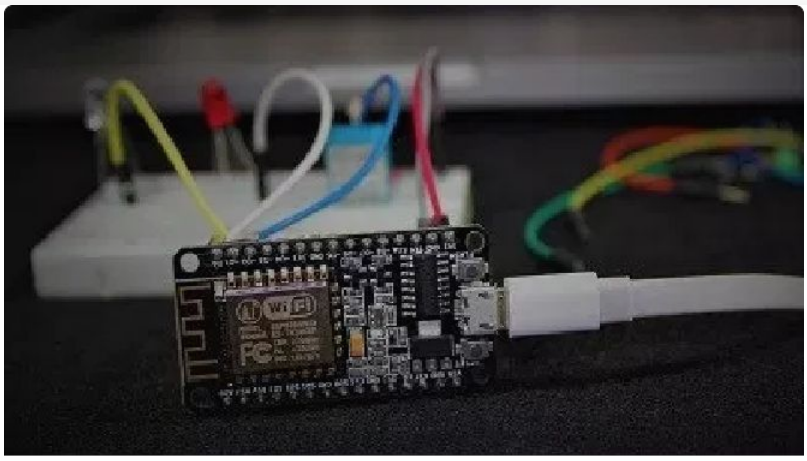
Do you have any questions? Leave a comment down below!

Thanks for reading. If you like this post probably you might like my next ones, so please support me by subscribing my blog.

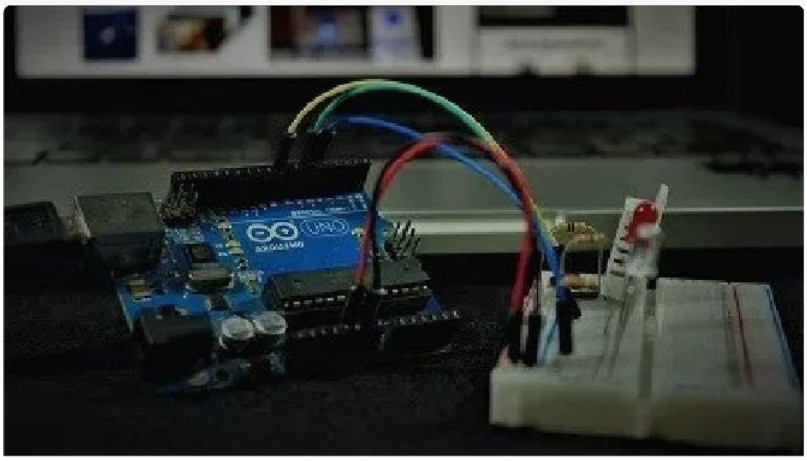
Recommended Resources



Build a Home Automation System from Scratch » With Raspberry Pi, ESP8266,



Home Automation using ESP8266 eBook and video course » Build IoT and home automation projects.



Arduino Step-by-Step Projects » Build 25 Arduino projects with our course, even with no prior experience!

What to Read Next...

ESP32 Built-In Hall Effect Sensor with Arduino IDE and MicroPython

ESP32 Servo Motor Web Server with Arduino IDE

Getting Started with Thonny MicroPython (Python) IDE for ESP32 and ESP8266

ESP32 with BMP180 Barometric Sensor – Guide

ESP32 Web Server with BME280 – Advanced Weather Station

Installing the ESP32 Board in Arduino IDE (Windows, Mac OS X, Linux)

3 thoughts on “Testing Mosquitto Broker and Client on Raspbbery Pi”

John

January 30, 2018 at 4:19 am | Reply

Hi Rui, Odd things happening for me on this one. I am using a Rpi2b, Raspian Pixel, Mosquitto ver 1.4.14. When I type in the second terminal window “mosquitto_pub -d -t testTopic -m “Hello World” the following occurs:
1 I receive an Error: Unknown option ‘world’
2 I noticed when I typed the command again on the first depression of the quote (shift + ‘) the quote did not show. (it does show in text editor).
AND if I type “mosquitto_pub -d -t testTopic -m Hello World (without the quotes) I get the same error, but if I type mosquitto_pub -d -t testTopic -m Hello, the message “Hello” is transmitted. Any thoughts on this?

November 5, 2018 at 7:51 am | Reply

I have tried using my PC with a virtual machine and linux installed and it works, Thanks!

David

April 24, 2019 at 4:29 pm | Reply

awesome tutorial, thanks. So many questions answered in my head, going through it and seeing it work really helps the understanding.

Leave a Comment

Name *

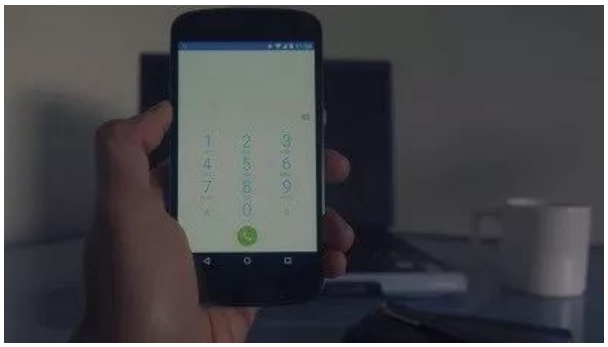
Email *

Website

Notify me of follow-up comments by email.

Post Comment

Home Automation using ESP8266 eBook and video course » Build IoT and home automation projects.



Build a Home Automation System from Scratch » With Raspberry Pi, ESP8266, Arduino, and Node-RED.