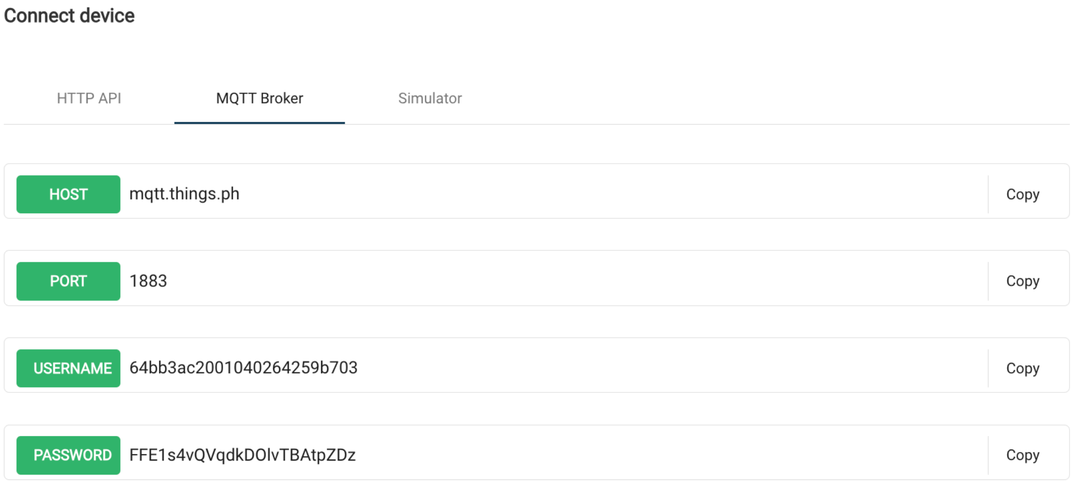
Part II: Testing Broker:

1. *Broker Username, Password, Hostname, and Port number*

a. In things.ph, on the left panel select **Devices**
b. In the top right, select **Connect Device.**
c. Select **MQTT Broker** from the top panel.



Connect device

HOST mqtt.things.ph

PORT 1883

USERNAME 64bb3ac2001040264259b703

PASSWORD FFE1s4vQVqdkDOlvTBAtpZDz

d. Take note of your username and password. This will be different for everyone.

2. *Testing your Broker*

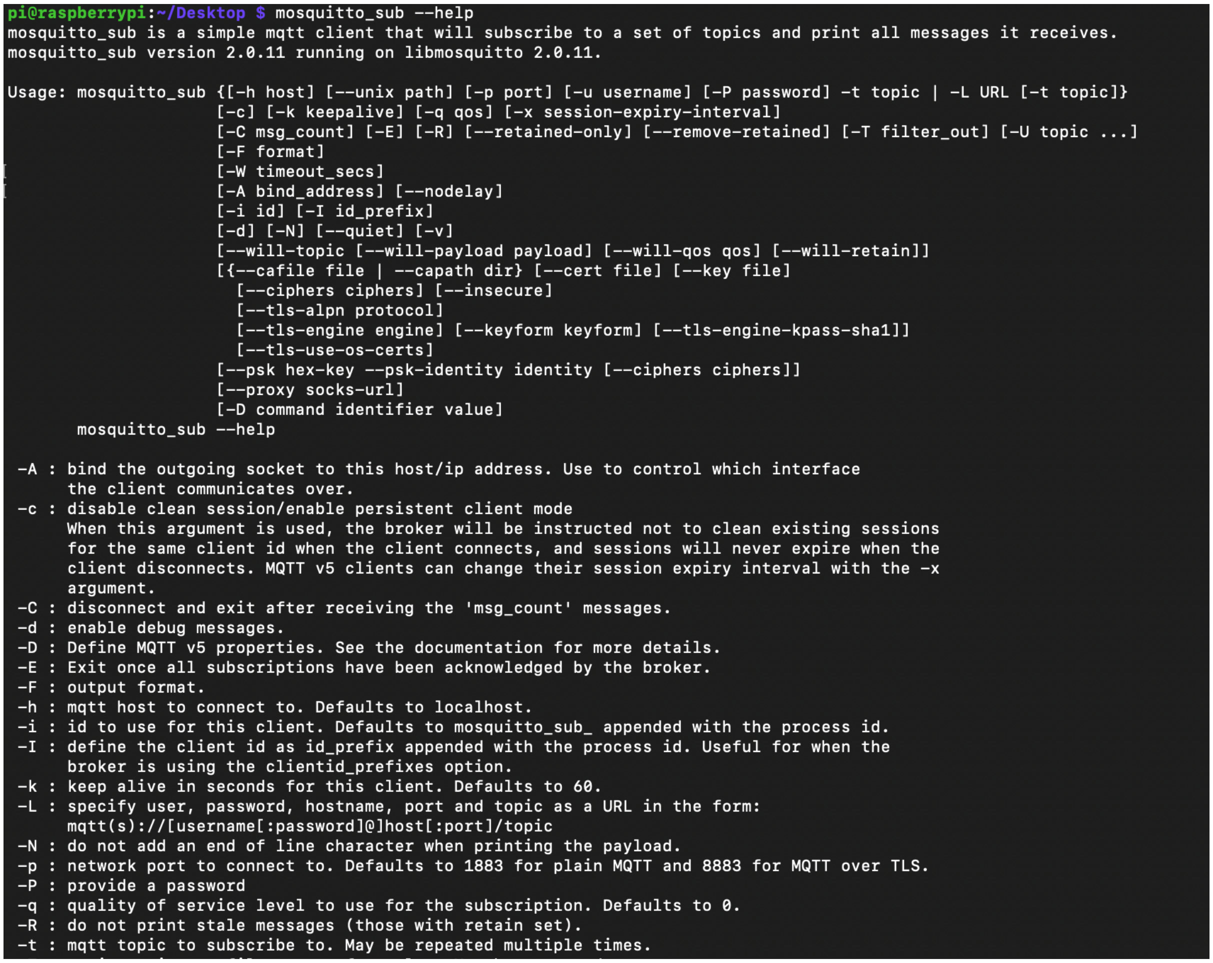
a. Open a new terminal window on your raspberry pi. Enter the following command to install mosquitto.

```
sudo apt install mosquitto mosquitto-clients
```

b. mosquitto_sub is used to subscribe to a topic as a client.

To see a full list of commands, enter:

```
mosquitto_sub -- help
```



```
pi@raspberrypi:~/Desktop $ mosquitto_sub --help
mosquitto_sub is a simple mqtt client that will subscribe to a set of topics and print all messages it receives.
mosquitto_sub version 2.0.11 running on libmosquitto 2.0.11.

Usage: mosquitto_sub {[-h host] [--unix path] [-p port] [-u username] [-P password] -t topic | -L URL [-t topic]}
                     [-c] [-k keepalive] [-q qos] [-x session-expiry-interval]
                     [-C msg_count] [-E] [-R] [--retained-only] [--remove-retained] [-T filter_out] [-U topic ...]
                     [-F format]
                     [-W timeout_secs]
                     [-A bind_address] [--nodelay]
                     [-i id] [-I id_prefix]
                     [-d] [-N] [--quiet] [-v]
                     [--will-topic [--will-payload payload] [--will-qos qos] [--will-retain]]
                     [{--cafile file | --capath dir} [--cert file] [--key file]
                       [--ciphers ciphers] [--insecure]
                       [--tls-alpn protocol]
                       [--tls-engine engine] [--keyform keyform] [--tls-engine-kpass-sha1]]
                       [--tls-use-os-certs]
                     [--psk hex-key --psk-identity identity [--ciphers ciphers]]
                     [--proxy socks-url]
                     [-D command identifier value]
       mosquitto_sub --help

 -A : bind the outgoing socket to this host/ip address. Use to control which interface
      the client communicates over.
 -c : disable clean session/enable persistent client mode
      When this argument is used, the broker will be instructed not to clean existing sessions
      for the same client id when the client connects, and sessions will never expire when the
      client disconnects. MQTT v5 clients can change their session expiry interval with the -x
      argument.
 -C : disconnect and exit after receiving the 'msg_count' messages.
 -d : enable debug messages.
 -D : Define MQTT v5 properties. See the documentation for more details.
 -E : Exit once all subscriptions have been acknowledged by the broker.
 -F : output format.
 -h : mqtt host to connect to. Defaults to localhost.
 -i : id to use for this client. Defaults to mosquitto_sub_ appended with the process id.
 -I : define the client id as id_prefix appended with the process id. Useful for when the
      broker is using the clientid_prefixes option.
 -k : keep alive in seconds for this client. Defaults to 60.
 -L : specify user, password, hostname, port and topic as a URL in the form:
      mqtt(s)://[username[:password]@]host[:port]/topic
 -N : do not add an end of line character when printing the payload.
 -p : network port to connect to. Defaults to 1883 for plain MQTT and 8883 for MQTT over TLS.
 -P : provide a password
 -q : quality of service level to use for the subscription. Defaults to 0.
 -R : do not print stale messages (those with retain set).
 -t : mqtt topic to subscribe to. May be repeated multiple times.
```

We will be using **-t -u -p -P -h.**

Using your username, password, hostname, and port number obtained from Part I, enter the following command into the terminal window.

mosquitto_sub -t RaspiMQTT -u "**YOUR USERNAME**" -p "1883" -P "**YOUR PASSWORD**" -h "mqtt.things.ph"

c. mosquitto_pub is used to publish to a topic as a client.

   Open a second terminal window.

   To see a full list of commands, enter:

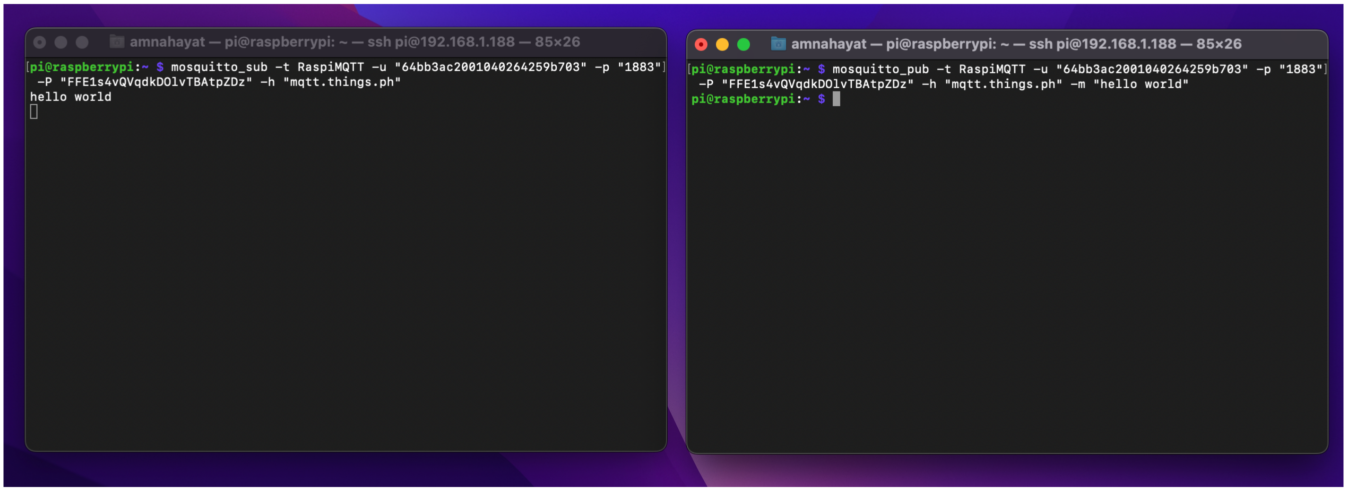
   mosquitto_pub -- help

   We will be using the same commands used for subscribing, with the addition of **-m,** which allows us to send a message.

   Enter the following command into the terminal window:

   mosquitto_sub -t RaspiMQTT -u "**YOUR USERNAME**" -p "1883" -P "**YOUR PASSWORD**" -h "mqtt.things.ph" -m “hello world”

Once you enter this command, you should see it pop up in the terminal window where you subscribed. This means you have successfully connected to the broker!



amnahayat — pi@raspberrypi: ~ — ssh pi@192.168.1.188 — 85×26

```
[pi@raspberrypi:~ $ mosquitto_sub -t RaspiMQTT -u "64bb3ac2001040264259b703" -p "1883"]
 -P "FFE1s4vQVqdkDOlvTBAtpZDz" -h "mqtt.things.ph"
hello world
```

amnahayat — pi@raspberrypi: ~ — ssh pi@192.168.1.188 — 85×26

```
[pi@raspberrypi:~ $ mosquitto_pub -t RaspiMQTT -u "64bb3ac2001040264259b703" -p "1883"]
 -P "FFE1s4vQVqdkDOlvTBAtpZDz" -h "mqtt.things.ph" -m "hello world"
pi@raspberrypi:~ $
```