



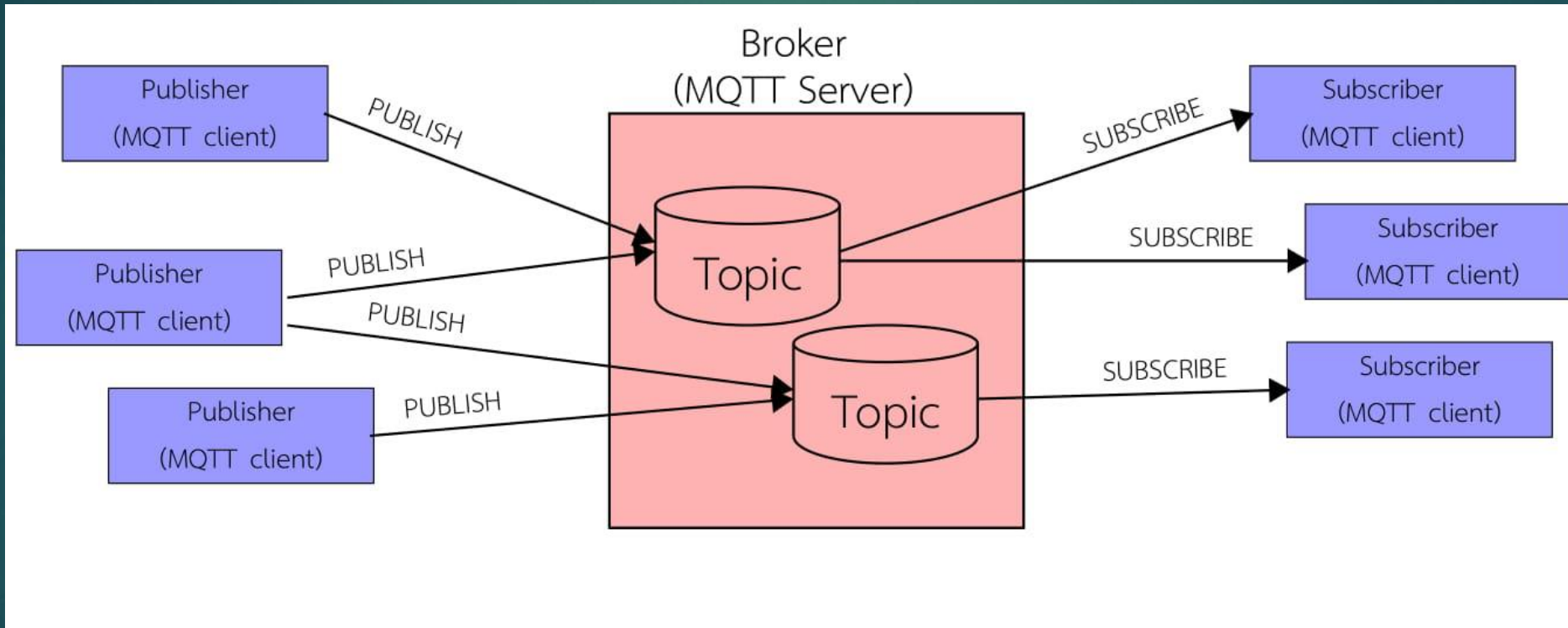
Publish/Subscribe

# Overview

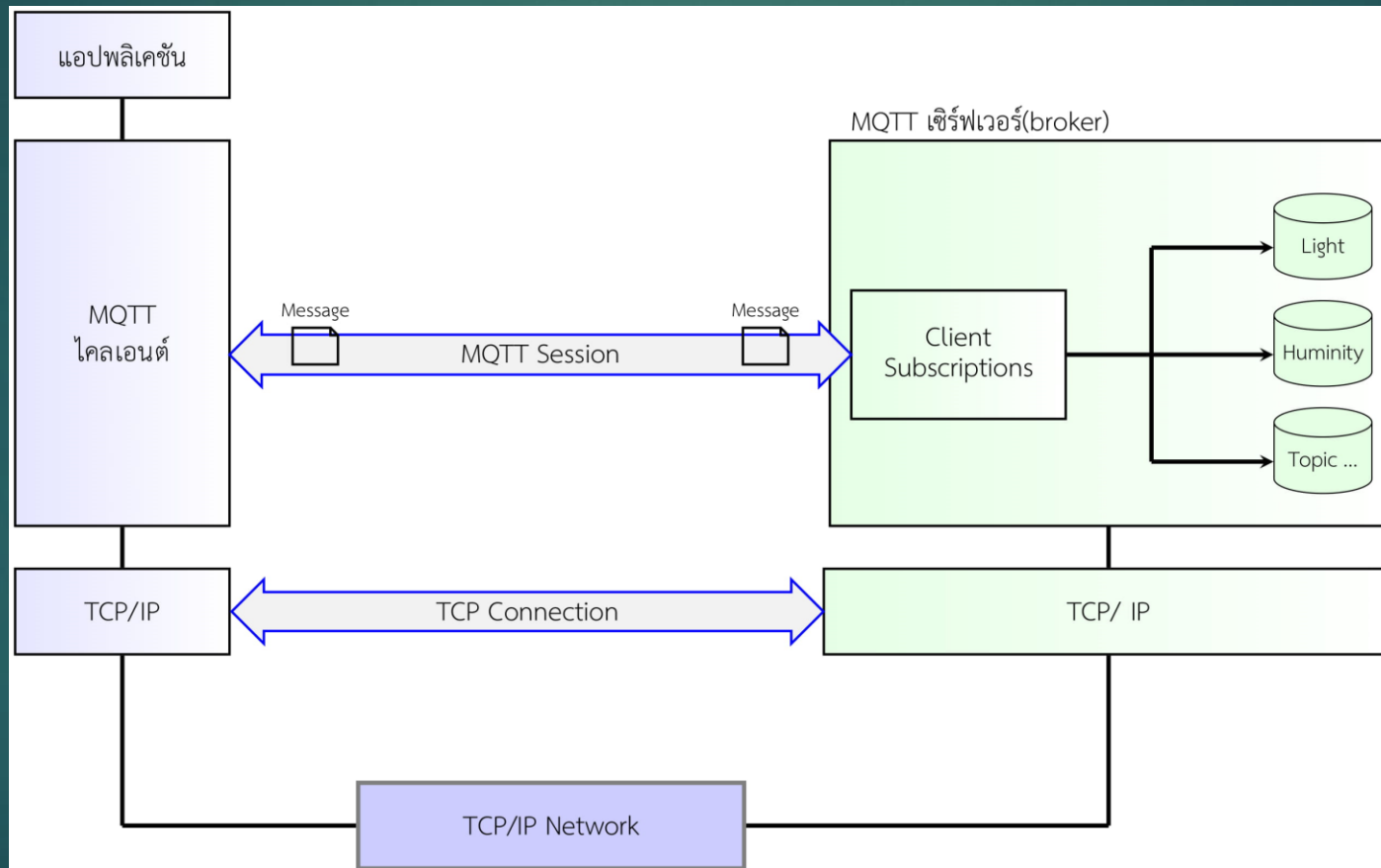
## ▶ ข้อดี

- ▶ ผู้ส่งและผู้รับไม่จำเป็นต้องมีการสื่อสารกันโดยตรง
- ▶ ทั้งนี้ผู้ส่งในที่นี้ก็คือพับบลิชเซอร์สามารถส่งเมสเสจ (Message) เข้าสู่ระบบด้วยการอ้างถึงทอปิก (Topic) ที่ต้องการ
- ▶ การสื่อสารที่เป็นแบบมัลติคาสต์แทนที่จะเป็นแบบหนึ่งต่อหนึ่งเท่านั้น

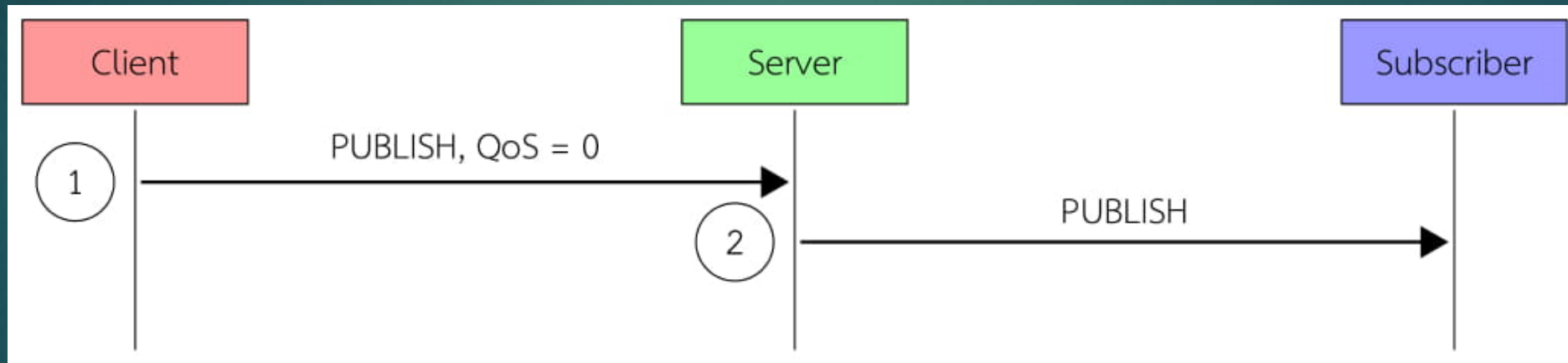
# System Architecture



# องค์ประกอบหลัก

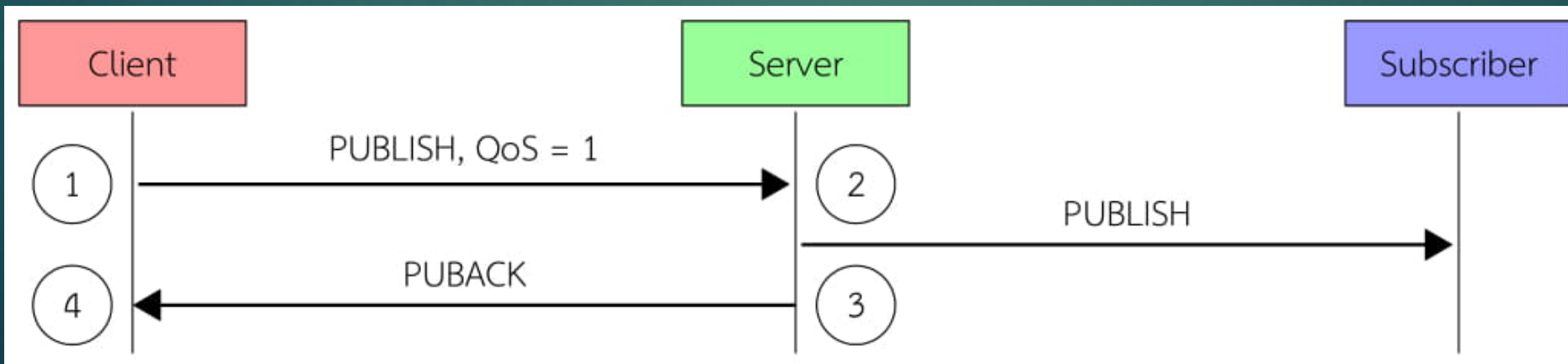


# QoS 0 (At-most-once, No Guarantee)



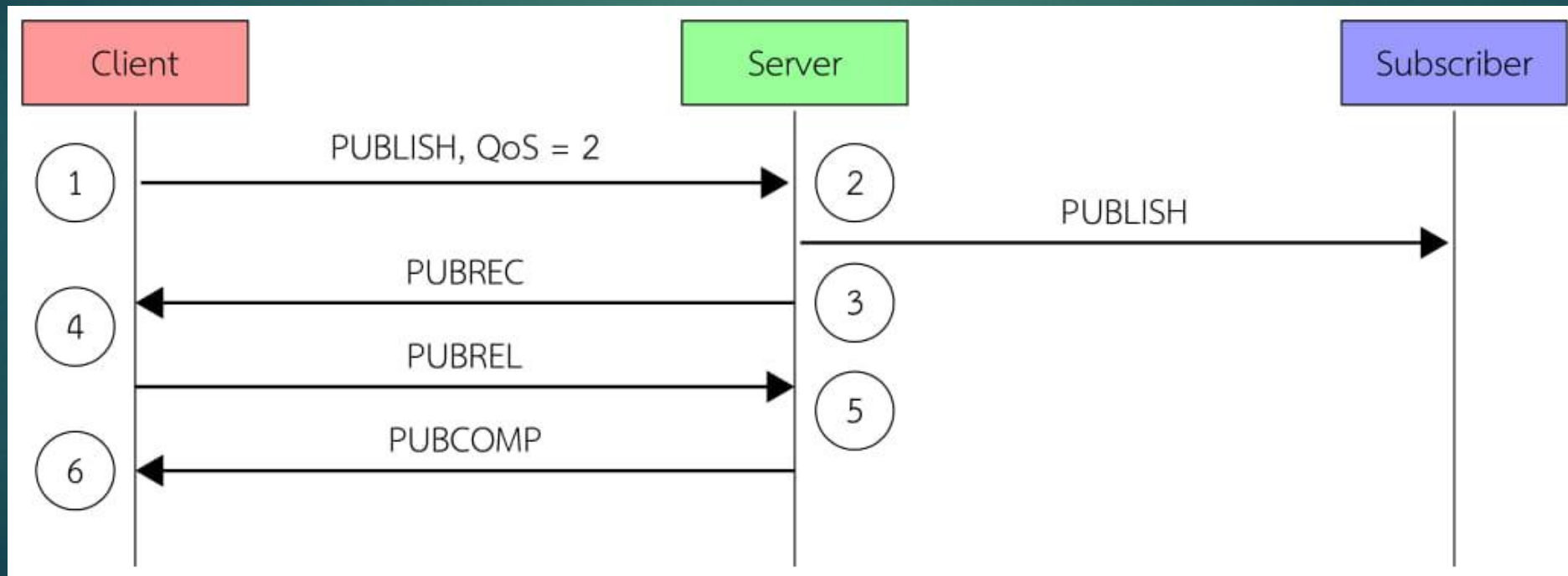
# QoS 1 (At-least-once, Acknowledgement)

- เมสเสจจะถูกส่งอย่างน้อยหนึ่งครั้ง
- หากภาคส่งไม่ได้รับแอคโนเลจเมนต์ เมสเสจจะถูกส่งออกไปใหม่ และมีการกำหนดบิต DUP เป็น 1
- เมสเสจใน QoS 1 จะถูกจัดเก็บไว้จนกระทั่งได้รับการตอบแอคโนเลจเมนต์ จึงจะถูกปล่อยออกไป
- QoS 1 ถูกกำหนดเป็นดีฟอลต์ QoS ของการสื่อสารแบบ MQTT



# QoS 2 (Exactly-once)

- QoS 2 เป็นการส่งเพียงครั้งเดียวเท่านั้น (Exactly-once) โดยমেসেজที่ส่งจะถูกจัดเก็บไว้ที่ภาคส่งและภาครับจนกระทั่งได้รับการประมวลผล



# Installation

- ▶ Install MQTT
- ▶ Ubuntu

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

- ▶ Others
- ▶ <https://mosquitto.org/download/>
- ▶ and example
  - ▶ <https://delightnet.nl/index.php/mqtt/12-mqtt-broker-installation>



# Eclipse Mosquitto

- ▶ โบรกเกอร์ที่รองรับโปรโตคอล MQTT
- ▶ **Download: <https://mosquitto.org/download/>**

# เริ่มต้นใช้งาน

## ▶ โบรกเกอร์

```
mosquitto [-c config file] [ -d | --daemon ] [-p port number] [-v]
```

## ▶ แพ็บลิชเชอร์

```
mosquitto_pub [-h hostname] [-p port-number] -t message-topic -m message [-q message-QoS]
```

## ▶ ซับสไครเบอร์

```
mosquitto_sub { [-h hostname] [-p port-number] -t message-topic [-q message-QoS] [-U unsub-topic]
```

# ทดสอบ

▶ สร้าง โบรกเกอร์ ซับสไครเบอร์ พับบลิชเชอร์ ภายในเครื่อง

▶ 1. ติดตั้ง

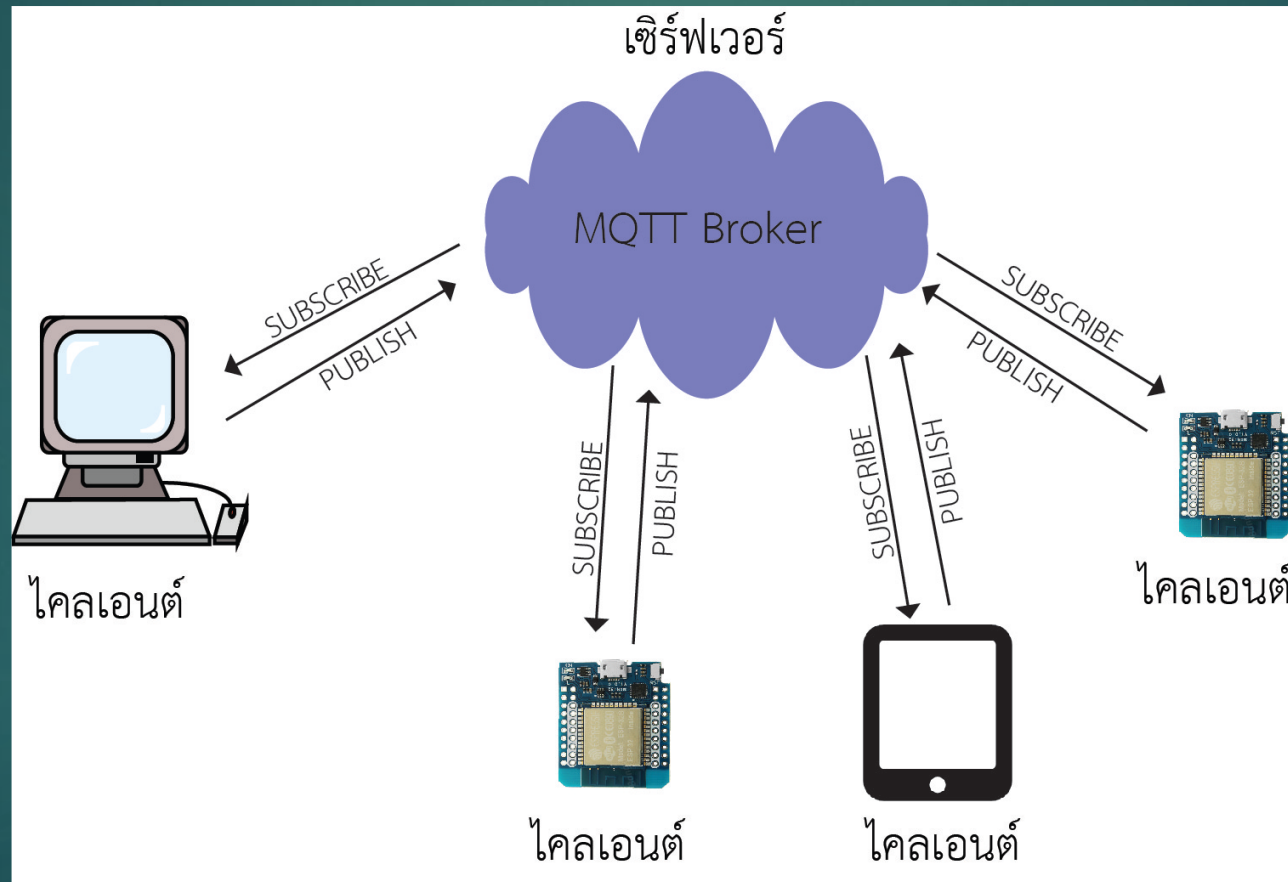
▶ 2. สร้างซับสไครเบอร์

```
mosquitto_sub -t Test -v -h localhost
```

▶ 3. สร้างพับบลิชเชอร์

```
mosquitto_pub -m "list of speed" -t Test -d -h localhost
```

# MQTT Topology



# ติดต่อกับ ESP32 ติดตั้ง Micropython: umqtt2

- ▶ Remove the version in micropython (umqtt.simple)

```
import sys
sys.path.reverse()
```

- ▶ Install the umqtt.simple2 (เชื่อมต่อไวไฟก่อน)

```
import upip
upip.install("micropython-umqtt.simple2")
```

# การสร้างไคลเอนต์

- ▶ สิ่งที่ต้องมี
  - ▶ Client ID
  - ▶ Server ( Broker location)

```
class MQTTClient:  
    def __init__(self, client_id, server, port=0, user=None,  
                  password=None, keepalive=0,  
                  ssl=False, ssl_params=None, socket_timeout=5,  
                  message_timeout=10):
```

# การสร้างไคลเอนต์แบบพับบลิชเซอร์

```
from umqtt.simple2 import MQTTClient
c = MQTTClient("umqtt_pub", Server)
c.connect()
c.publish(b"theTopic", b"mymessage")
```

# การสร้างไคลเอนต์แบบซึบสไครเบอร์

```
from umqtt.simple2 import MQTTClient
c = MQTTClient("umqtt_sub", Server)
c.set_callback(callback_fn)
c.connect()
c.subscribe(b"theTopic")
```



# จับสไครบ์ทุกทอปิก

```
import time
from umqtt.simple2 import MQTTClient

# Received messages from subscriptions will be delivered to this
# callback
def sub_callback(topic, msg, retain, dup):
    print("t:{} m:{}".format(topic, msg))

client = MQTTClient("esp32_sub", "192.168.0.52", port=1883,
                    user=None, password=None, keepalive=30,
                    ssl=False, ssl_params={})

client.set_callback(sub_callback)
client.connect()
client.subscribe(b"#")
count = 0
for count in range(10):
    client.check_msg()
    time.sleep(3)
client.disconnect()
```