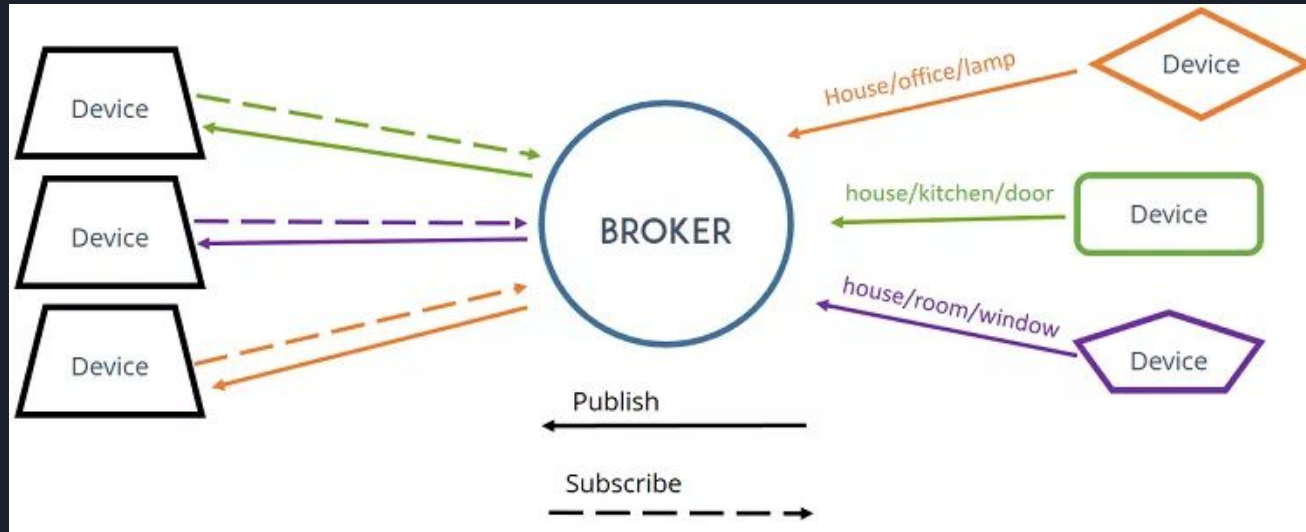




IOT Lab Assignment 3

mqtt

What is MQTT?



MQTT is a simple messaging protocol, designed for constrained devices with low-bandwidth.

- Broker
- Topics
- Publish / Subscribe

Wifi you can use in the lab

Ssid: dan

Pw: supersecretpassword





Setting up the broker on your pi

`sudo dnf install mosquitto` (install mqtt software)

`sudo firewall-cmd --permanent --add-port=1883/tcp` (let through firewall)

`service firewalld restart` (restart firewall)

`systemctl enable mosquitto` (enable service on reboots)

`systemctl start mosquitto` (start mqtt)

`systemctl status mosquitto` (make sure it's running)



Test the broker

HOST = your ip address (use the command 'ip a' to get this)

Run this command in one terminal `mosquitto_sub -h HOST -t /test`

Run this command in another terminal `mosquitto_pub -h HOST -t /test -m 'im a message'`

You should see 'im a message' show up in the first terminal



Install Python & paho-mqtt library on pi

Sudo dnf install python

Sudo pip install paho-mqtt

Docs: <https://github.com/eclipse/paho.mqtt.python>

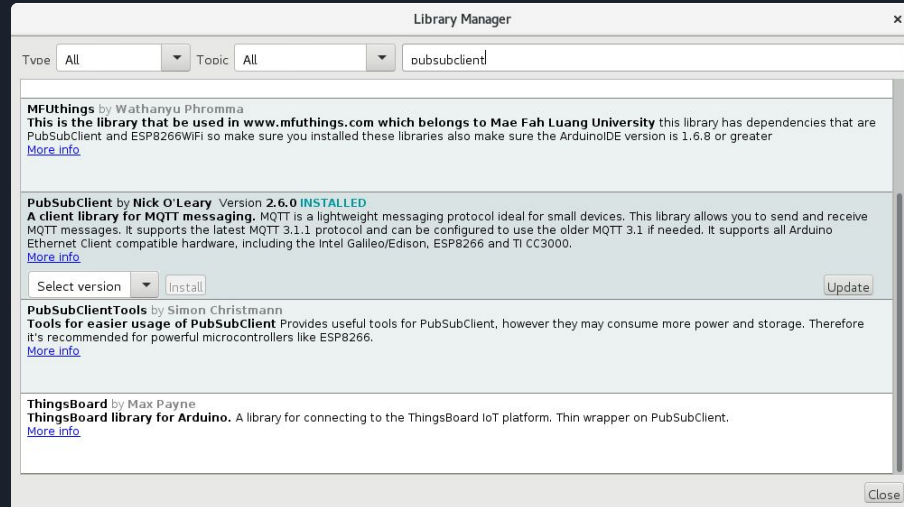


example_paho.py

```
1  import paho.mqtt.client as mqtt #import the library
2
3
4  def on_message(client, userdata, message):
5      print(message.topic + " " + str(message.payload)) #print incoming messages
6
7
8
9
10 broker_address="192.168.1.3"    #broker address (your pis ip address)
11
12 client = mqtt.Client("python client") #create new client instance
13 client.connect(broker_address) #connect to broker
14
15 client.on_message=on_message #set the on message function
16
17 client.subscribe("/test") #subscribe to topic
18
19 client.loop_start() #start client
20
21 #do something
22 if True:
23     client.publish("/test2","0") #publish
24
25
26 client.loop_stop() #stop client
27
```

Install PubSubClient in arduino ide

Sketch -> Include Library -> Manage Libraries -> PubSubClient



Docs: <https://pubsubclient.knolleary.net/api.html>

Example_pubsub

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>

// WiFi/MQTT parameters
#define WLAN_SSID       "dan"
#define WLAN_PASS       "supersecretpassword"
#define BROKER_IP       "192.168.1.3"

WiFiClient client;
PubSubClient mqttclient(client);

void callback(char* topic, byte* payload, unsigned int length) {
    Serial.println(topic);
    Serial.write(payload, length); //print incoming messages
    Serial.println("");
}

void setup() {
    Serial.begin(115200);

    // connect to wifi
    WiFi.mode(WIFI_STA);
    WiFi.begin(WLAN_SSID, WLAN_PASS);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(F("."));
    }

    Serial.println(F("WiFi connected"));
    Serial.println(F("IP address: "));
    Serial.println(WiFi.localIP());

    // connect to mqtt server
    mqttclient.setServer(BROKER_IP, 1883);
    mqttclient.setCallback(callback);
    connect();
}
```

```
void loop() {
    if (!mqttclient.connected()) {
        connect();
    }

    if (true){
        mqttclient.publish("/test1", "test message", false); //send message
    }

    mqttclient.loop();
}

void connect() {
    while (WiFi.status() != WL_CONNECTED) {
        Serial.println(F("Wifi issue"));
        delay(3000);
    }
    Serial.print(F("Connecting to MQTT server... "));
    while(!mqttclient.connected()) {
        if (mqttclient.connect(WiFi.macAddress().c_str())) {
            Serial.println(F("MQTT server Connected!"));

            mqttclient.subscribe("/test"); //subscribe to topic

        } else {
            Serial.print(F("MQTT server connection failed! rc="));
            Serial.print(mqttclient.state());
            Serial.println("try again in 10 seconds");
            // Wait 5 seconds before retrying
            delay(20000);
        }
    }
}
```

Assignment

- Set up mqtt broker on pi
- Create python program to send messages to toggle led and receive button messages
- Create a arduino program to receive led message and turn the led on/off and send message when button is pressed
- Take a video of yours and send it to me before next class dpivonka@redhat.com

