

Code for Practical 4

A screenshot of a Windows PowerShell window titled "Windows PowerShell". The command `mosquitto_sub` is run three times to subscribe to the topic "topic/test". The command `mosquitto_pub` is run twice to publish messages to the same topic. The published messages are "Hello I am publisher from cmd" and "Chickens". The PowerShell window has a light blue background and a white foreground. The text output is in black font. The window title bar shows "Windows PowerShell" and the taskbar shows icons for OneDrive, Redmi 12C, This PC, OS (C), and Google Drive.

```
PS C:\Program Files\mosquitto> ./mosquitto_sub -h localhost -t topic/test
PS C:\Program Files\mosquitto> ./mosquitto_sub -h localhost -t topic/test
PS C:\Program Files\mosquitto> ./mosquitto_sub -h localhost -t topic/test
Hello I am publisher from cmd
Chickens

PS C:\Program Files\mosquitto> ./mosquitto_pub -h localhost -t topic/test -m "Hello I am publisher from cmd"
PS C:\Program Files\mosquitto> ./mosquitto_pub -h localhost -t topic/test -m "Hello I am publisher from cmd"
PS C:\Program Files\mosquitto> ./mosquitto_pub -h localhost -t topic/test -m "Hello I am publisher from cmd"
PS C:\Program Files\mosquitto> ./mosquitto_pub -h localhost -t topic/test -m "Chickens"
```

Figure 1: Publisher and Subscriber - Part 1

```
[Administrator: Windows PowerShell]
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements

PS C:\WINDOWS\system32> C:\Program Files\mosquitto
C:\Program : The term 'C:\Program' is not recognized as the name
program. Check the spelling of the name, or if a path was inclu
At Line:1 char:1
+ C:\Program Files\mosquitto
+ CategoryInfo          : ObjectNotFound: (C:\Program:String)
+ FullyQualifiedErrorId : CommandNotFoundException

PS C:\WINDOWS\system32> cd C:\Program Files\mosquitto
Set-Location : A positional parameter cannot be found that accept
At Line:1 char:1
+ cd C:\Program Files\mosquitto
+ CategoryInfo          : InvalidArgument: () [Set-Location]
+ FullyQualifiedErrorId : PositionalParameterNotFound,Micr

Ln 1, Col 1    483 characters      Plain text      100%      Unix (LF)

.edmi 12C      Unwind.dll.exe      30/10/2025 09:18 AM  A
his PC          uv.dll            12/07/2025 03:14 AM  A
OS (C)          websockets.dll    12/07/2025 08:15 AM  A
                      zlib1.dll        12/07/2025 03:11 AM  A
```

Figure 2: Creating and Hashing the Passwords

The image shows two separate Windows PowerShell windows. The top window displays a subscriber session where a connection attempt to a local host on topic/test2 fails due to unauthorized access. The bottom window shows a publisher session where messages are sent to the same topic with payloads like 'This is a message', 'hello', and 'hiii'.

Figure 3: Publisher and Subscriber with passwords

```

import paho.mqtt.client as mqtt
import RPi.GPIO as GPIO
import time

LEDin=18

GPIO.setmode(GPIO.BCM)
GPIO.setup(LEDin,GPIO.OUT)

def on_message(client, userdata, message):
    if (str(message.payload.decode("utf-8"))== '1'):
        GPIO.output(LEDin,GPIO.HIGH)
        print("Received Message: ", str(message.payload.decode("utf-8")))
        time.sleep(2)
    else:
        GPIO.output(LEDin,GPIO.LOW)
        print("Received Message: ", str(message.payload.decode("utf-8")))
        time.sleep(2)

mqttBroker = "192.168.8.115"
client = mqtt.Client(client_id="subscriberclient", callback_api_version=mqtt.CallbackAPIVersion.VERSION2)
client.username_pw_set("marc", "potato")
client.connect(mqttBroker)
client.subscribe("topic/led")
client.on_message = on_message
client.loop_forever()

```

Figure 4: Subscriber Code for Raspberry Pi

```

import paho.mqtt.client as mqtt
import time

mqttBroker = "localhost"
client = mqtt.Client(client_id="publisherclient", callback_api_version=mqtt.CallbackAPIVersion.VERSION2)
client.username_pw_set("marc", "potato")
client.connect(mqttBroker)

while True:
    val = input("Enter LED value (1=on, 0=off, q=quit): ")
    if val.lower() == 'q':
        break
    if val in ['0', '1']:
        client.publish("topic/led", int(val))
        print(f"Just published {val}")
    time.sleep(1)

```

Figure 5: Publisher Code for Laptop

The screenshot shows the Geany IDE interface with multiple tabs open. The current tab is 'prac4.py'. The code in the editor is a MQTT publisher. A terminal window is also visible, showing the output of running the script, which includes a warning about a channel being already in use and the received messages from the subscriber.

```

import paho.mqtt.client as mqtt
import RPi.GPIO as GPIO
import time

LEDin=18

GPIO.setmode(GPIO.BCM)
GPIO.setup(LEDin,GPIO.OUT)

def on_message(client, userdata, message):
    if (str(message.payload.decode("utf-8"))=="'1')":
        GPIO.output(LEDin,GPIO.HIGH)
        print("Received Message: ", str(message.payload.decode("utf-8")))
        time.sleep(2)
    else:
        GPIO.output(LEDin,GPIO.LOW)
        print("Received Message: ", str(message.payload.decode("utf-8")))
        time.sleep(2)

mqttBroker = "192.168.8.115"
client = mqtt.Client(client_id="subscriberclient", callback_api_version=mqtt.CallbackAPIVersion.VERSION2)
client.username_pw_set("marc", "potato")
client.connect(mqttBroker)
client.subscribe("topic/led")
client.on_message = on_message
client.loop_forever()

```

Figure 6: Case Study receiving the signal