Code for Practical 4

Figure 1: Publisher and Subscriber - Part 1

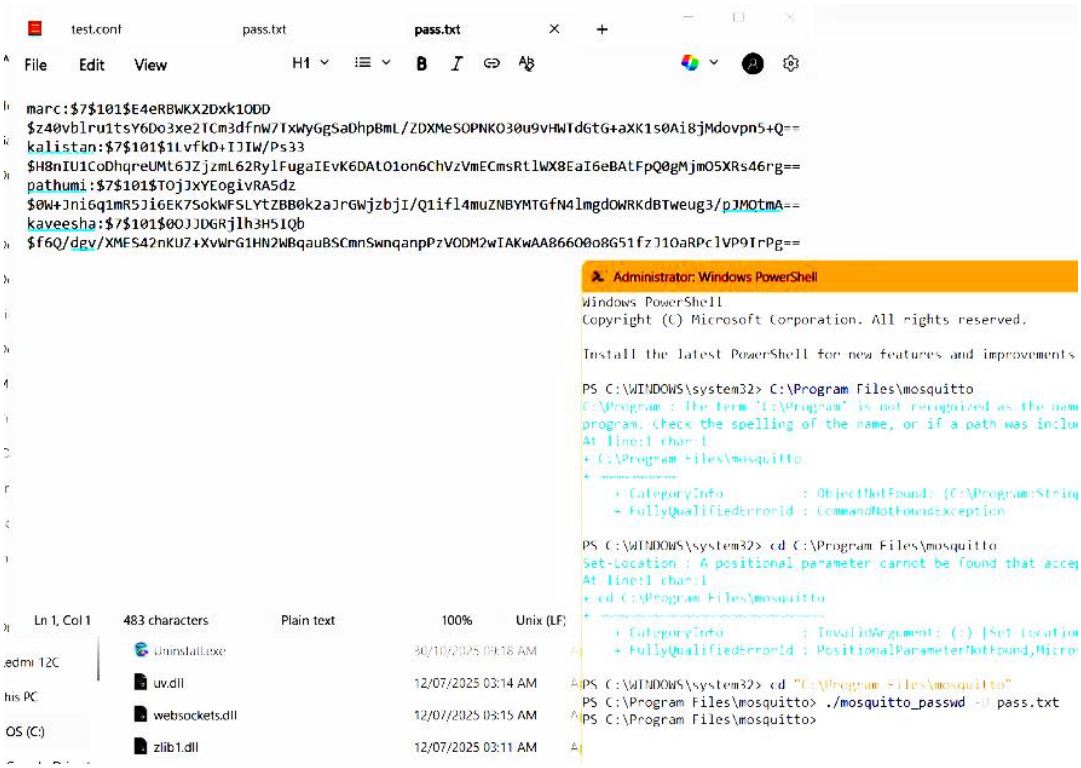

Figure 2: Creating and Hashing the Passwords

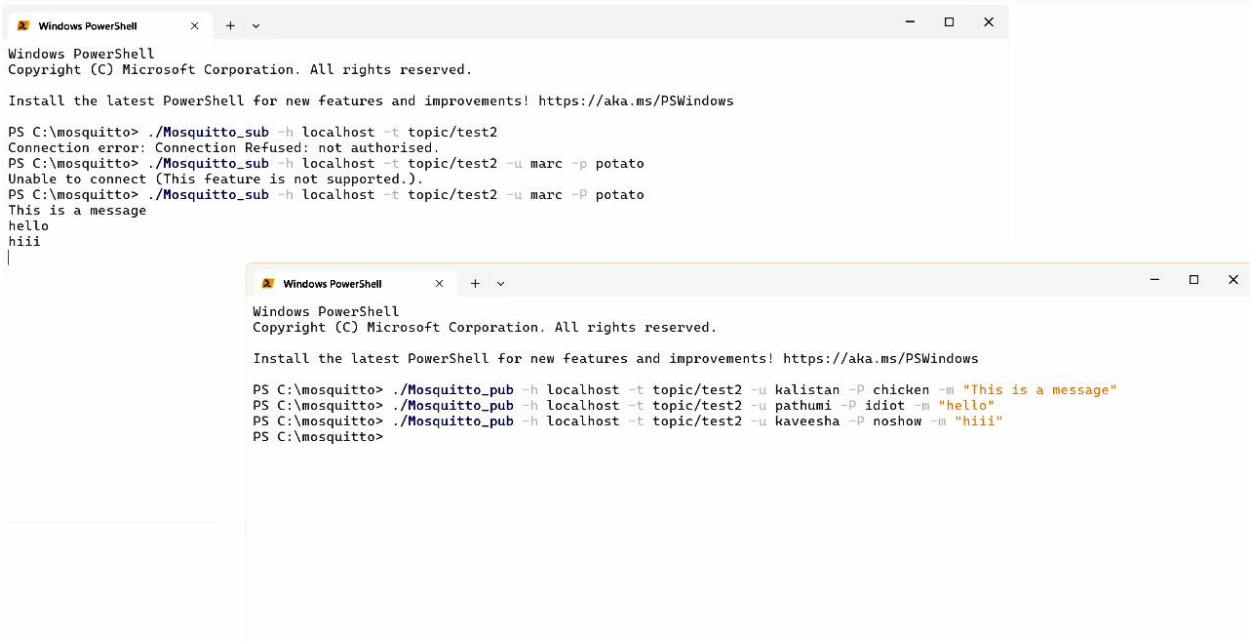

Figure 3: Publisher and Subscriber with passwords

```python
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

```python
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

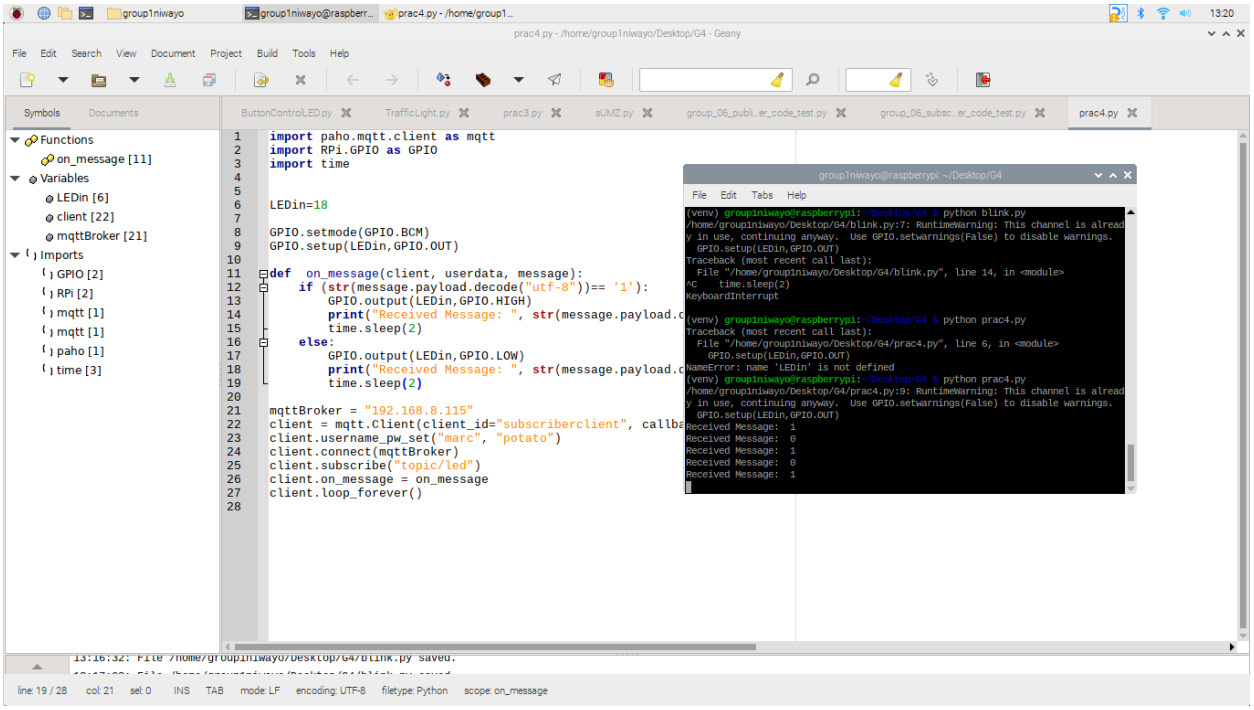

Figure 6: Case Study receiving the signal