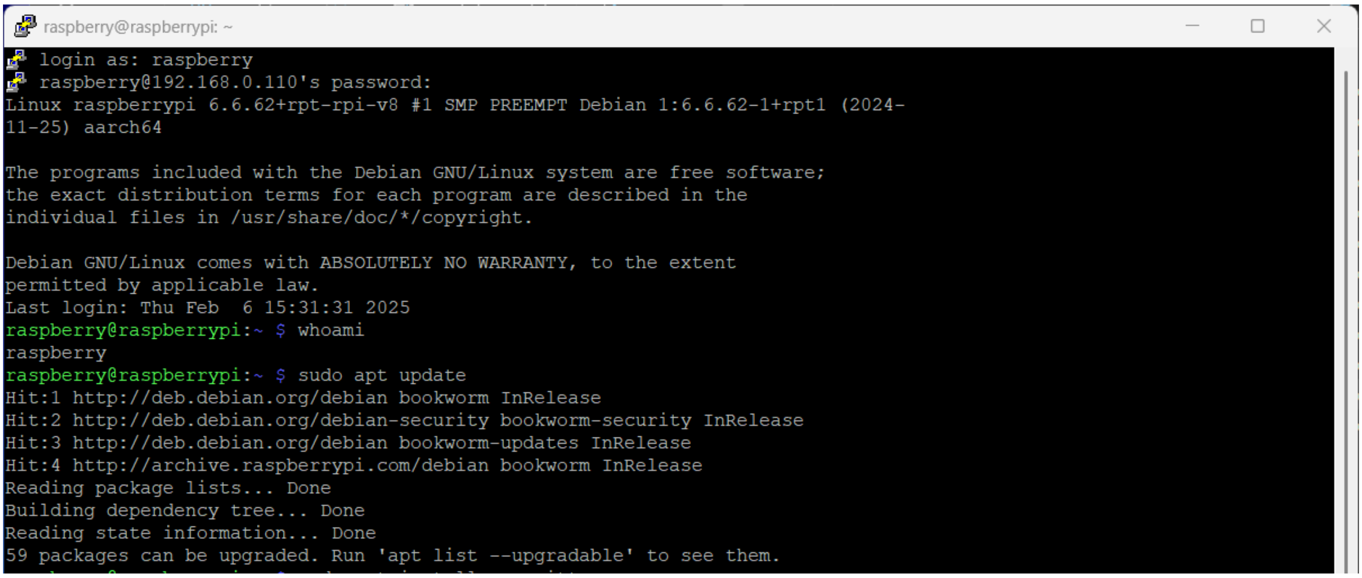
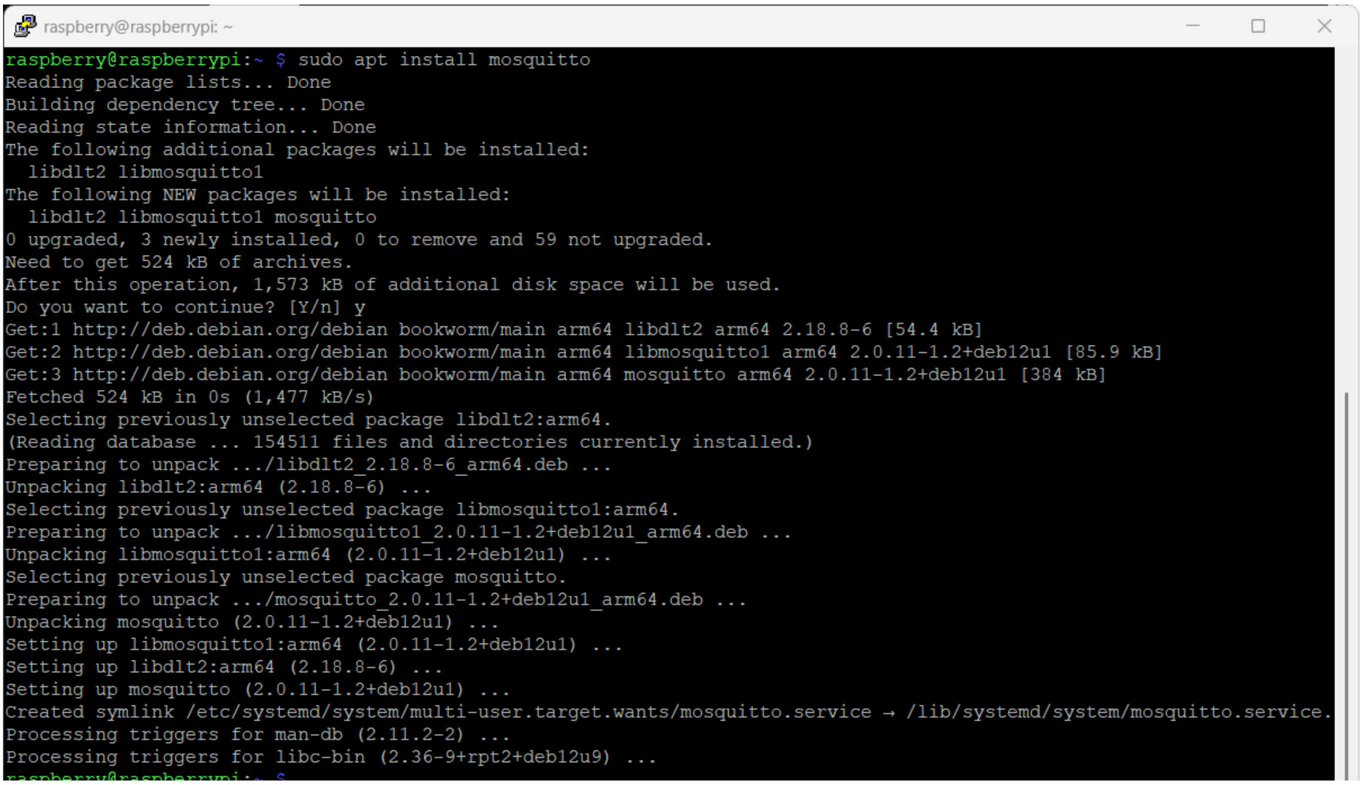
Lab05 (MQTT) Screenshots

1. Install and Configure the MQTT Broker on a Raspberry Pi 3:



```
login as: raspberry
raspberry@192.168.0.110's password:
Linux raspberrypi 6.6.62+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.62-1+rpt1 (2024-
11-25) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Feb  6 15:31:31 2025
raspberry@raspberrypi:~ $ whoami
raspberry
raspberry@raspberrypi:~ $ sudo apt update
Hit:1 http://deb.debian.org/debian bookworm InRelease
Hit:2 http://deb.debian.org/debian-security bookworm-security InRelease
Hit:3 http://deb.debian.org/debian bookworm-updates InRelease
Hit:4 http://archive.raspberrypi.com/debian bookworm InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
59 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

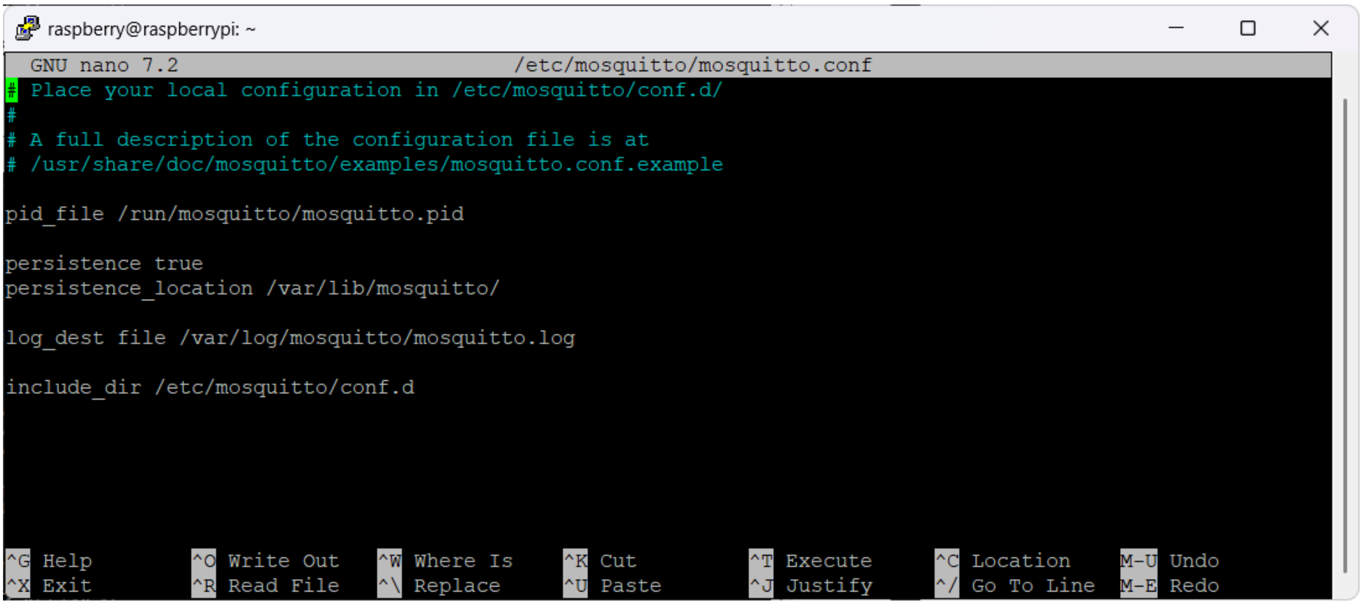
Fig. 1. Screenshot of updating the Raspberry Pi's package list



```
raspberry@raspberrypi:~ $ sudo apt install mosquitto
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libdlt2 libmosquitto1
The following NEW packages will be installed:
  libdlt2 libmosquitto1 mosquitto
0 upgraded, 3 newly installed, 0 to remove and 59 not upgraded.
Need to get 524 kB of archives.
After this operation, 1,573 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://deb.debian.org/debian bookworm/main arm64 libdlt2 arm64 2.18.8-6 [54.4 kB]
Get:2 http://deb.debian.org/debian bookworm/main arm64 libmosquitto1 arm64 2.0.11-1.2+deb12u1 [85.9 kB]
Get:3 http://deb.debian.org/debian bookworm/main arm64 mosquitto arm64 2.0.11-1.2+deb12u1 [384 kB]
Fetched 524 kB in 0s (1,477 kB/s)
Selecting previously unselected package libdlt2:arm64.
(Reading database ... 154511 files and directories currently installed.)
Preparing to unpack .../libdlt2_2.18.8-6_arm64.deb ...
Unpacking libdlt2:arm64 (2.18.8-6) ...
Selecting previously unselected package libmosquitto1:arm64.
Preparing to unpack .../libmosquitto1_2.0.11-1.2+deb12u1_arm64.deb ...
Unpacking libmosquitto1:arm64 (2.0.11-1.2+deb12u1) ...
Selecting previously unselected package mosquitto.
Preparing to unpack .../mosquitto_2.0.11-1.2+deb12u1_arm64.deb ...
Unpacking mosquitto (2.0.11-1.2+deb12u1) ...
Setting up libmosquitto1:arm64 (2.0.11-1.2+deb12u1) ...
Setting up libdlt2:arm64 (2.18.8-6) ...
Setting up mosquitto (2.0.11-1.2+deb12u1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/mosquitto.service → /lib/systemd/system/mosquitto.service.
Processing triggers for man-db (2.11.2-2) ...
Processing triggers for libc-bin (2.36-9+rpt2+deb12u9) ...
```

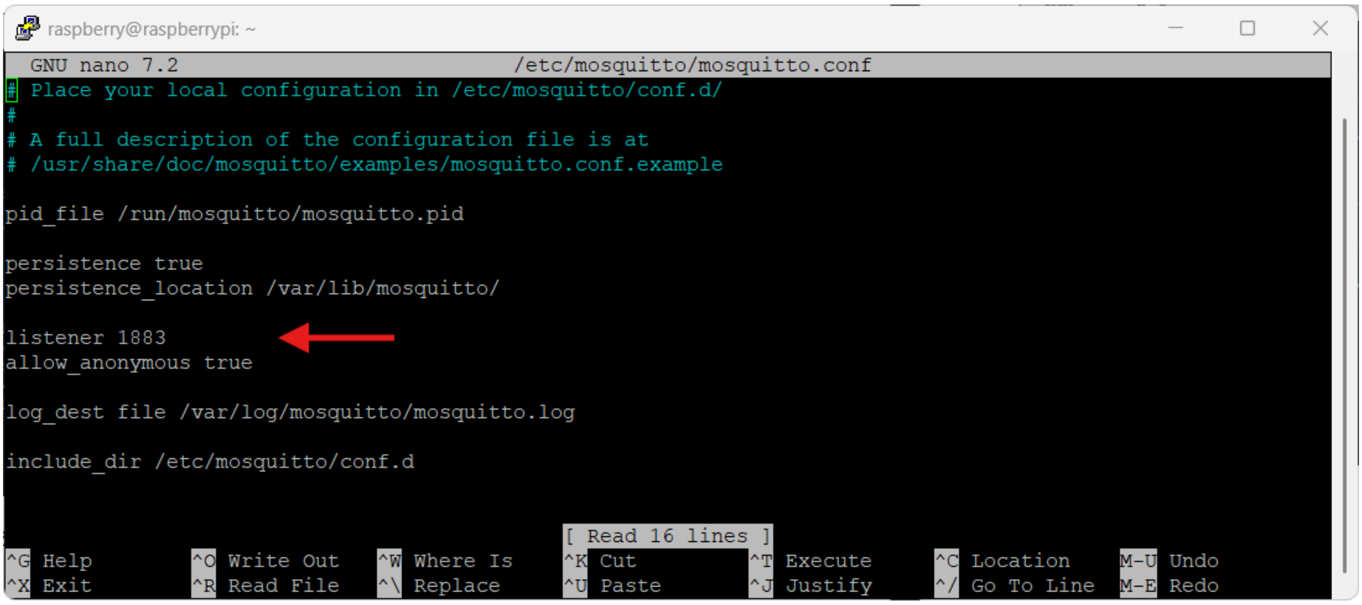
Fig. 2. Screenshot of installing the Mosquitto MQTT broker

Lab05 (MQTT) Screenshots



```
  GNU nano 7.2                        /etc/mosquitto/mosquitto.conf
# Place your local configuration in /etc/mosquitto/conf.d/
#
# A full description of the configuration file is at
# /usr/share/doc/mosquitto/examples/mosquitto.conf.example

pid_file /run/mosquitto/mosquitto.pid

persistence true
persistence_location /var/lib/mosquitto/

log_dest file /var/log/mosquitto/mosquitto.log

include_dir /etc/mosquitto/conf.d




^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo
```

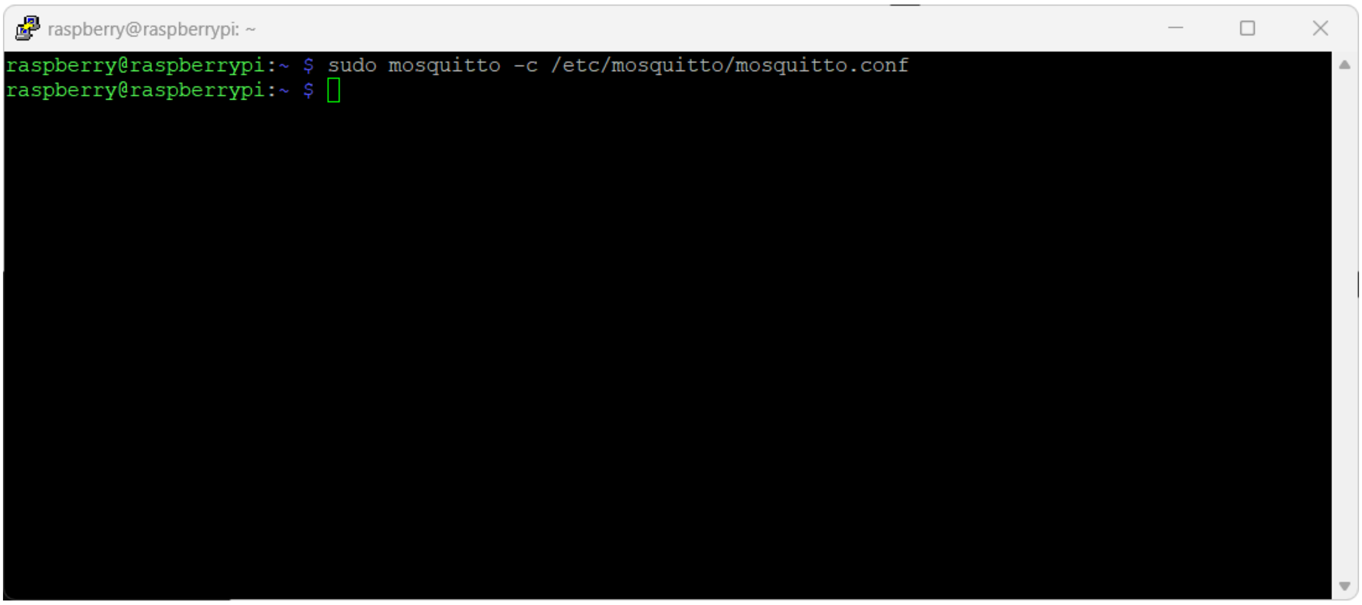
Fig. 3. Screenshot of Mosquitto Configuration File



```
  GNU nano 7.2                        /etc/mosquitto/mosquitto.conf
# Place your local configuration in /etc/mosquitto/conf.d/
#
# A full description of the configuration file is at
# /usr/share/doc/mosquitto/examples/mosquitto.conf.example

pid_file /run/mosquitto/mosquitto.pid

persistence true
persistence_location /var/lib/mosquitto/

listener 1883
allow_anonymous true

log_dest file /var/log/mosquitto/mosquitto.log

include_dir /etc/mosquitto/conf.d


                                                  [ Read 16 lines ]
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo
```

Fig. 4. Screenshot of Mosquitto Configuration File (after allowing broker to listen to port 1883 and allow anonymous clients to connect and use the MQTT broker)

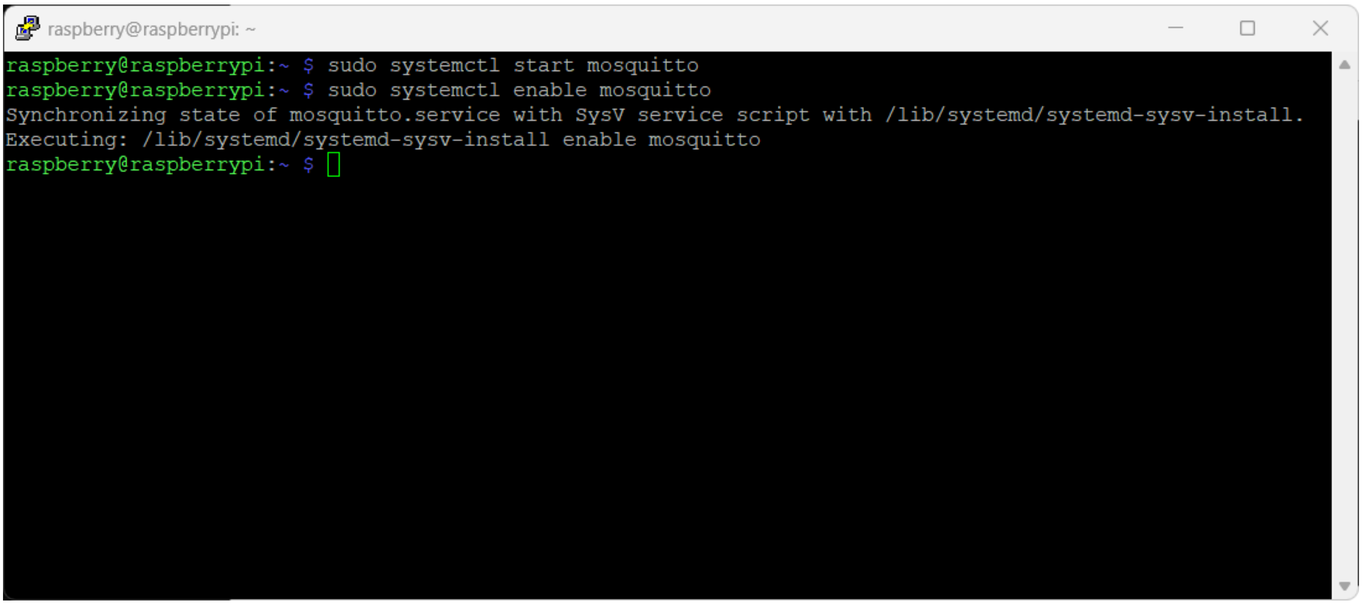
Lab05 (MQTT) Screenshots



```
raspberry@raspberrypi:~ $ sudo mosquitto -c /etc/mosquitto/mosquitto.conf
raspberry@raspberrypi:~ $ 
```

Fig. 4. Screenshot of starting mosquito broker manually via command line

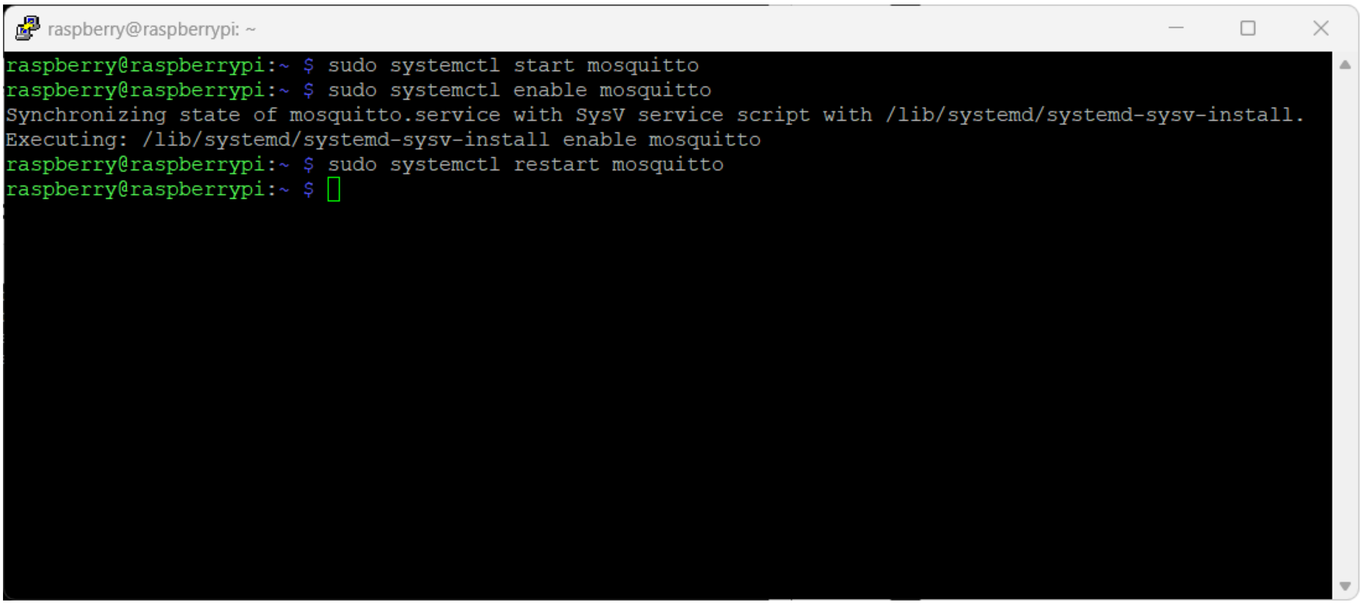
2. Enable Mosquitto Broker to run on boot



```
raspberry@raspberrypi:~ $ sudo systemctl start mosquitto
raspberry@raspberrypi:~ $ sudo systemctl enable mosquitto
Synchronizing state of mosquitto.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable mosquitto
raspberry@raspberrypi:~ $ 
```

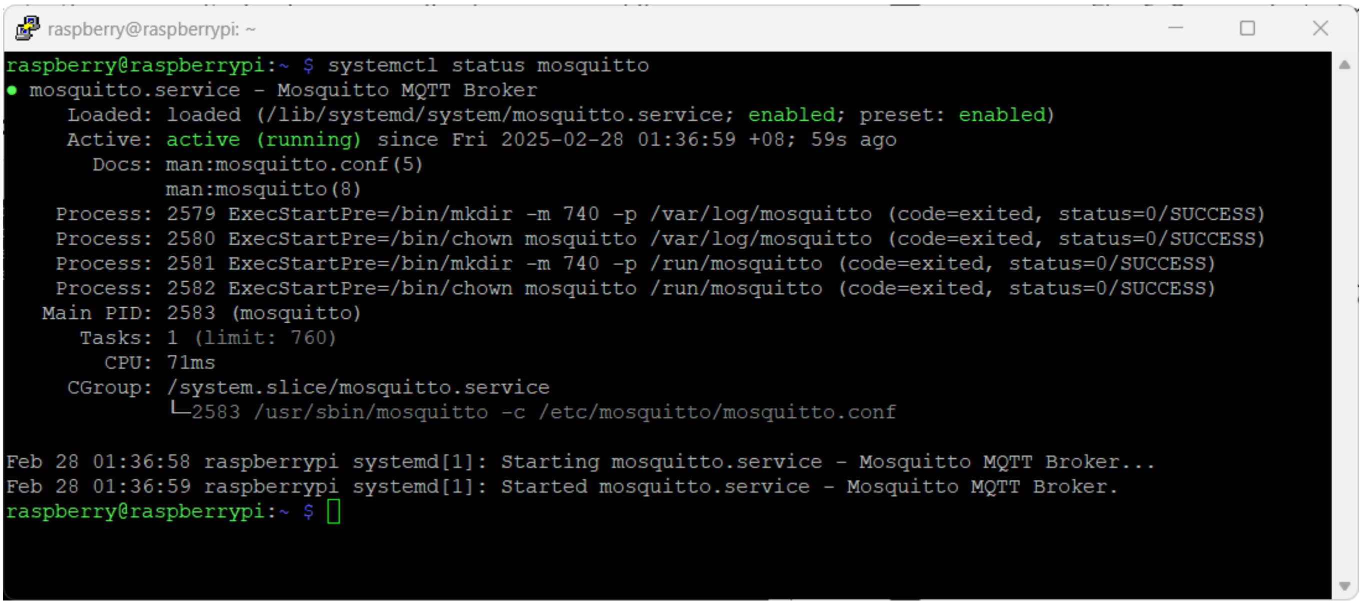
Fig. 5. Screenshot of starting and enabling Mosquitto to run on boot

Lab05 (MQTT) Screenshots



```
raspberry@raspberrypi:~ $ sudo systemctl start mosquitto
raspberry@raspberrypi:~ $ sudo systemctl enable mosquitto
Synchronizing state of mosquitto.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable mosquitto
raspberry@raspberrypi:~ $ sudo systemctl restart mosquitto
raspberry@raspberrypi:~ $ 
```

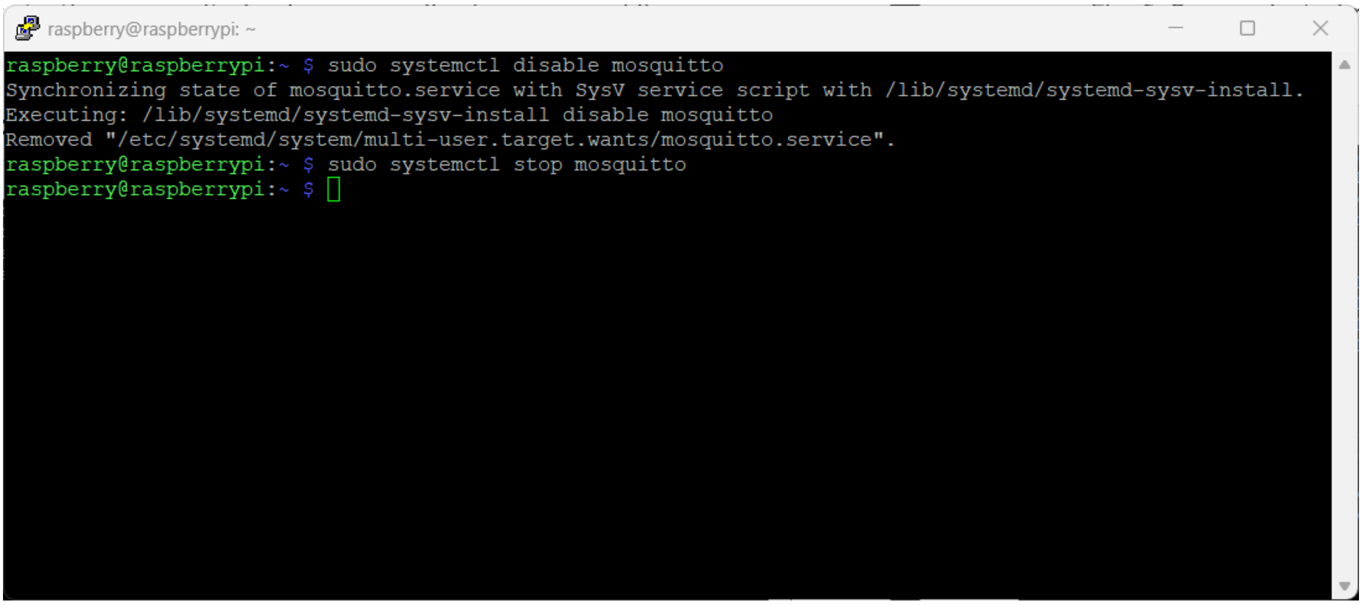
Fig. 6. Screenshot of restarting Mosquitto Broker to apply the new configuration



```
raspberry@raspberrypi:~ $ systemctl status mosquitto
● mosquitto.service - Mosquitto MQTT Broker
     Loaded: loaded (/lib/systemd/system/mosquitto.service; enabled; preset: enabled)
     Active: active (running) since Fri 2025-02-28 01:36:59 +08; 59s ago
       Docs: man:mosquitto.conf(5)
             man:mosquitto(8)
    Process: 2579 ExecStartPre=/bin/mkdir -m 740 -p /var/log/mosquitto (code=exited, status=0/SUCCESS)
    Process: 2580 ExecStartPre=/bin/chown mosquitto /var/log/mosquitto (code=exited, status=0/SUCCESS)
    Process: 2581 ExecStartPre=/bin/mkdir -m 740 -p /run/mosquitto (code=exited, status=0/SUCCESS)
    Process: 2582 ExecStartPre=/bin/chown mosquitto /run/mosquitto (code=exited, status=0/SUCCESS)
   Main PID: 2583 (mosquitto)
      Tasks: 1 (limit: 760)
        CPU: 71ms
     CGroup: /system.slice/mosquitto.service
             └─2583 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

Feb 28 01:36:58 raspberrypi systemd[1]: Starting mosquitto.service - Mosquitto MQTT Broker...
Feb 28 01:36:59 raspberrypi systemd[1]: Started mosquitto.service - Mosquitto MQTT Broker.
raspberry@raspberrypi:~ $ 
```

Fig. 7. Screenshot of Mosquitto running status

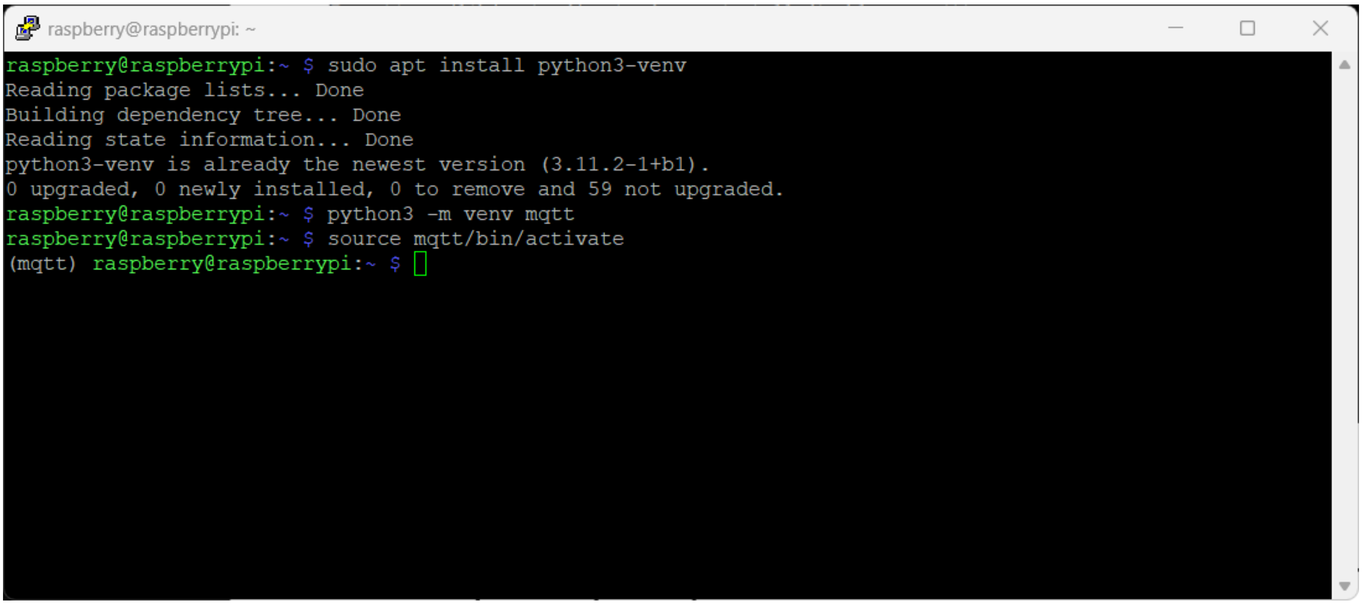
Lab05 (MQTT) Screenshots



```
raspberry@raspberrypi:~ $ sudo systemctl disable mosquitto
Synchronizing state of mosquitto.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable mosquitto
Removed "/etc/systemd/system/multi-user.target.wants/mosquitto.service".
raspberry@raspberrypi:~ $ sudo systemctl stop mosquitto
raspberry@raspberrypi:~ $ 
```

Fig. 8. Screenshot of disabling and stopping Mosquitto MQTT broker

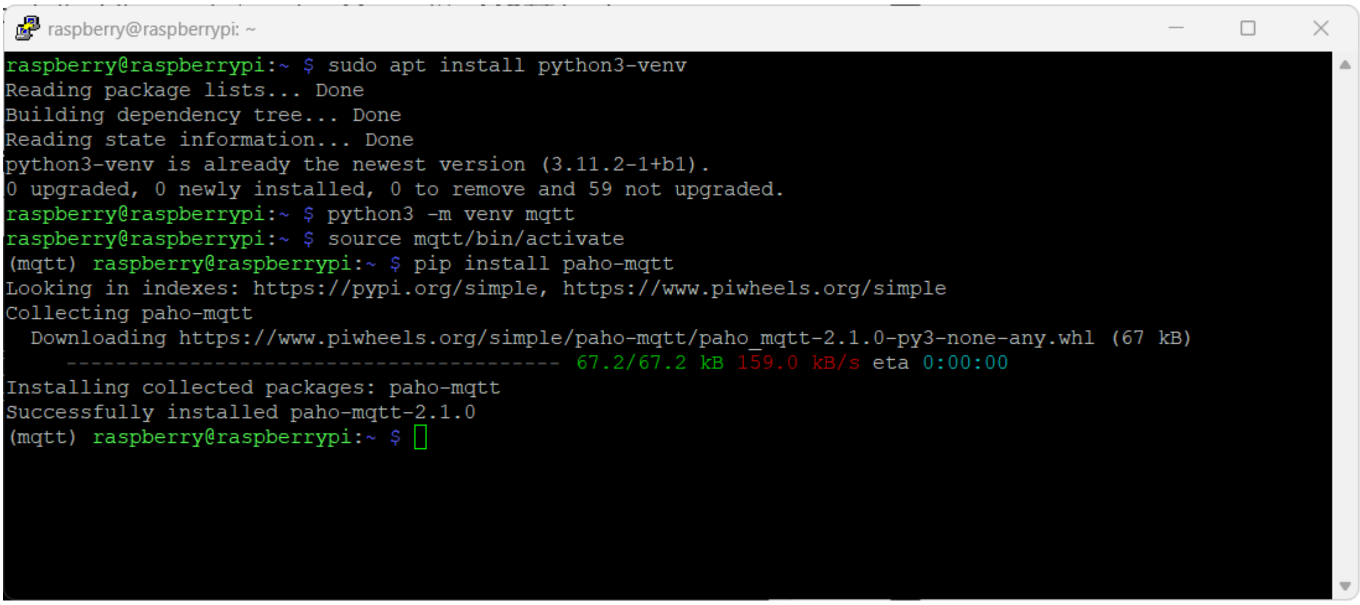
3. Install and Configure the MQTT Client (Publisher and/or Subscriber) on another Raspberry Pi 3



```
raspberry@raspberrypi:~ $ sudo apt install python3-venv
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3-venv is already the newest version (3.11.2-1+b1).
0 upgraded, 0 newly installed, 0 to remove and 59 not upgraded.
raspberry@raspberrypi:~ $ python3 -m venv mqtt
raspberry@raspberrypi:~ $ source mqtt/bin/activate
(mqtt) raspberry@raspberrypi:~ $ 
```

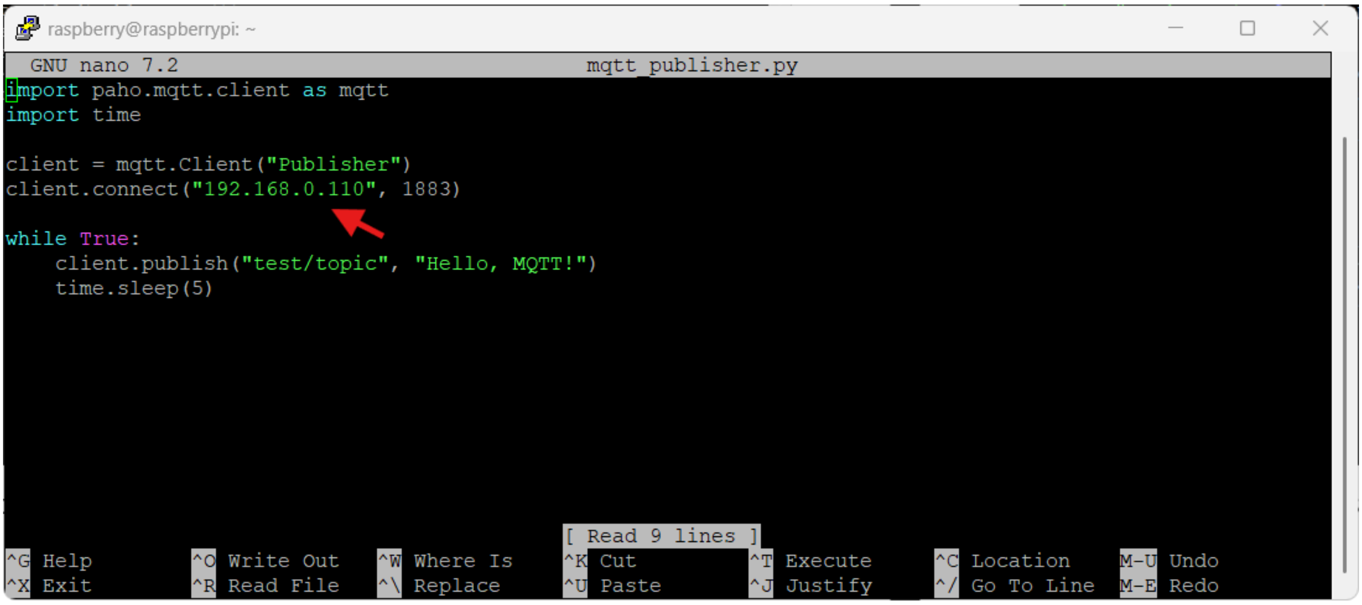
Fig. 9. Screenshot of activating the Virtual Environment `mqtt`

Lab05 (MQTT) Screenshots

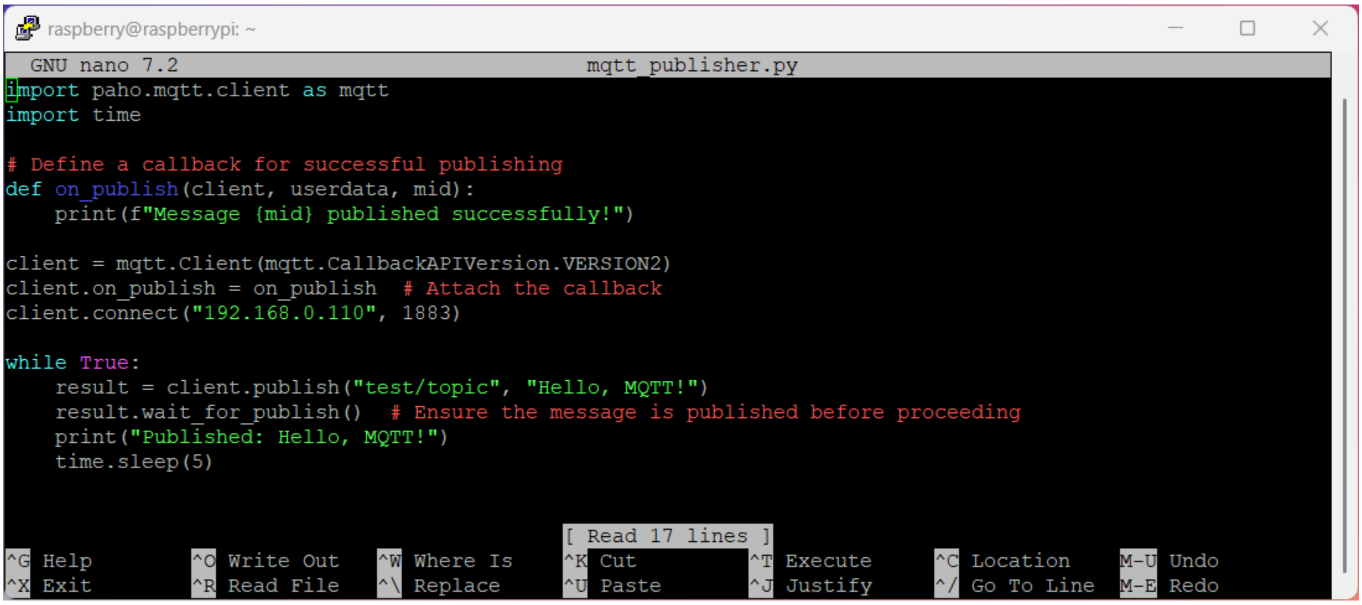


```
raspberry@raspberrypi:~ $ sudo apt install python3-venv
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3-venv is already the newest version (3.11.2-1+b1).
0 upgraded, 0 newly installed, 0 to remove and 59 not upgraded.
raspberry@raspberrypi:~ $ python3 -m venv mqtt
raspberry@raspberrypi:~ $ source mqtt/bin/activate
(mqtt) raspberry@raspberrypi:~ $ pip install paho-mqtt
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting paho-mqtt
  Downloading https://www.piwheels.org/simple/paho-mqtt/paho_mqtt-2.1.0-py3-none-any.whl (67 kB)
     ---------------------------------------- 67.2/67.2 kB 159.0 kB/s eta 0:00:00
Installing collected packages: paho-mqtt
Successfully installed paho-mqtt-2.1.0
(mqtt) raspberry@raspberrypi:~ $ 
```

Fig. 10. Screenshot of installation of Python Paho MQTT library for both publisher and subscriber

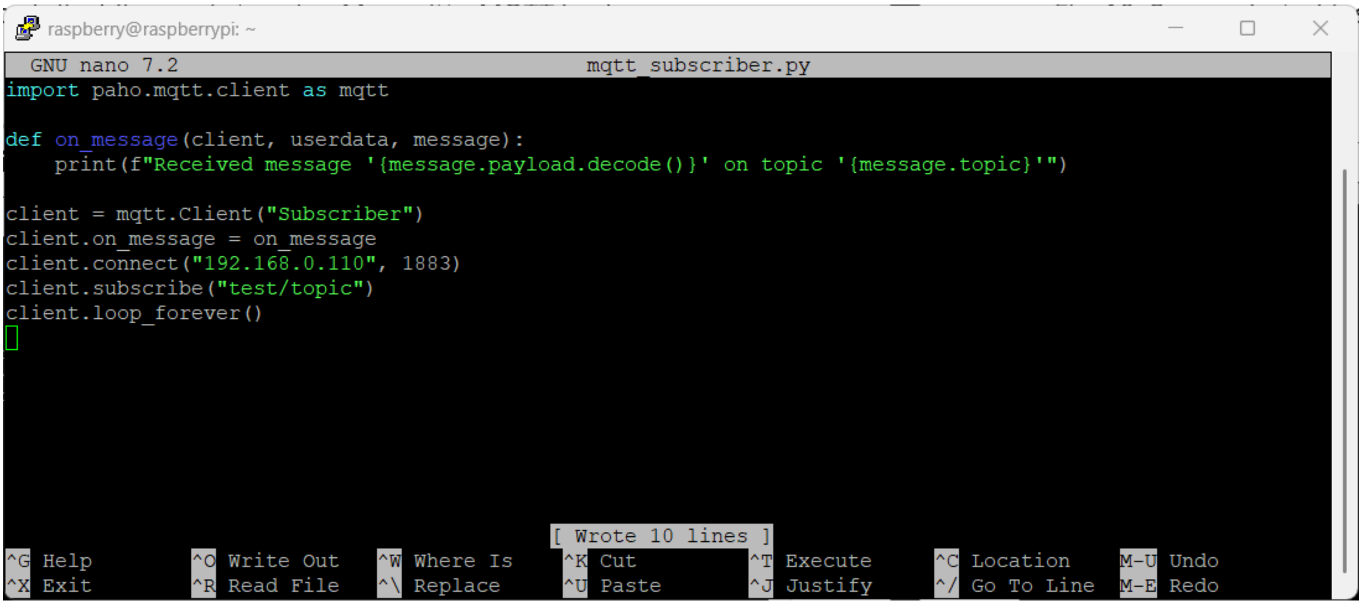


```
  GNU nano 7.2                                   mqtt_publisher.py
import paho.mqtt.client as mqtt
import time

client = mqtt.Client("Publisher")
client.connect("192.168.0.110", 1883)

while True:
    client.publish("test/topic", "Hello, MQTT!")
    time.sleep(5)








                                             [ Read 9 lines ]
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo
```



```
  GNU nano 7.2                              mqtt_publisher.py
import paho.mqtt.client as mqtt
import time

# Define a callback for successful publishing
def on_publish(client, userdata, mid):
    print(f"Message {mid} published successfully!")

client = mqtt.Client(mqtt.CallbackAPIVersion.VERSION2)
client.on_publish = on_publish  # Attach the callback
client.connect("192.168.0.110", 1883)

while True:
    result = client.publish("test/topic", "Hello, MQTT!")
    result.wait_for_publish()  # Ensure the message is published before proceeding
    print("Published: Hello, MQTT!")
    time.sleep(5)

                                             [ Read 17 lines ]
^G Help         ^O Write Out    ^W Where Is     ^K Cut          ^T Execute      ^C Location     M-U Undo
^X Exit         ^R Read File    ^\ Replace      ^U Paste        ^J Justify      ^/ Go To Line   M-E Redo
```

Fig. 11. Screenshot of the Python Script for the MQTT Publisher `mqtt_publisher.py` with the address changed to the broker's IP address.

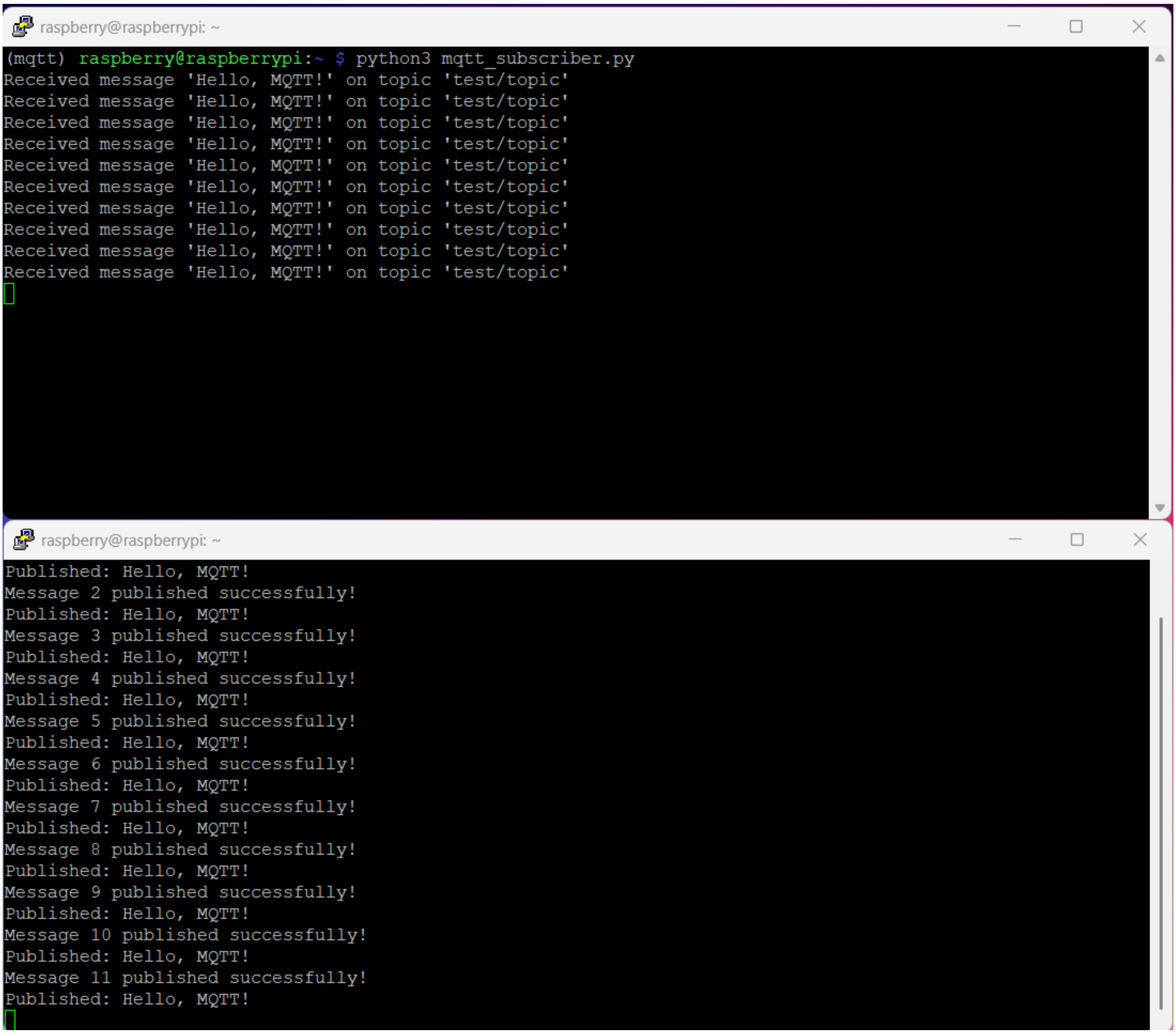


```
  GNU nano 7.2                              mqtt_subscriber.py
import paho.mqtt.client as mqtt

def on_message(client, userdata, message):
    print(f"Received message '{message.payload.decode()}' on topic '{message.topic}'")

client = mqtt.Client("Subscriber")
client.on_message = on_message
client.connect("192.168.0.110", 1883)
client.subscribe("test/topic")
client.loop_forever()


                                            [ Wrote 10 lines ]
^G Help         ^O Write Out    ^W Where Is     ^K Cut          ^T Execute      ^C Location     M-U Undo
^X Exit         ^R Read File    ^\ Replace      ^U Paste        ^J Justify      ^/ Go To Line   M-E Redo
```

Fig. 12. Screenshot of the Python Script for the MQTT Scriber `mqtt_subscriber.py` with the address changed to the broker's IP address.

4. Testing your MQTT Communication

Lab05 (MQTT) Screenshots



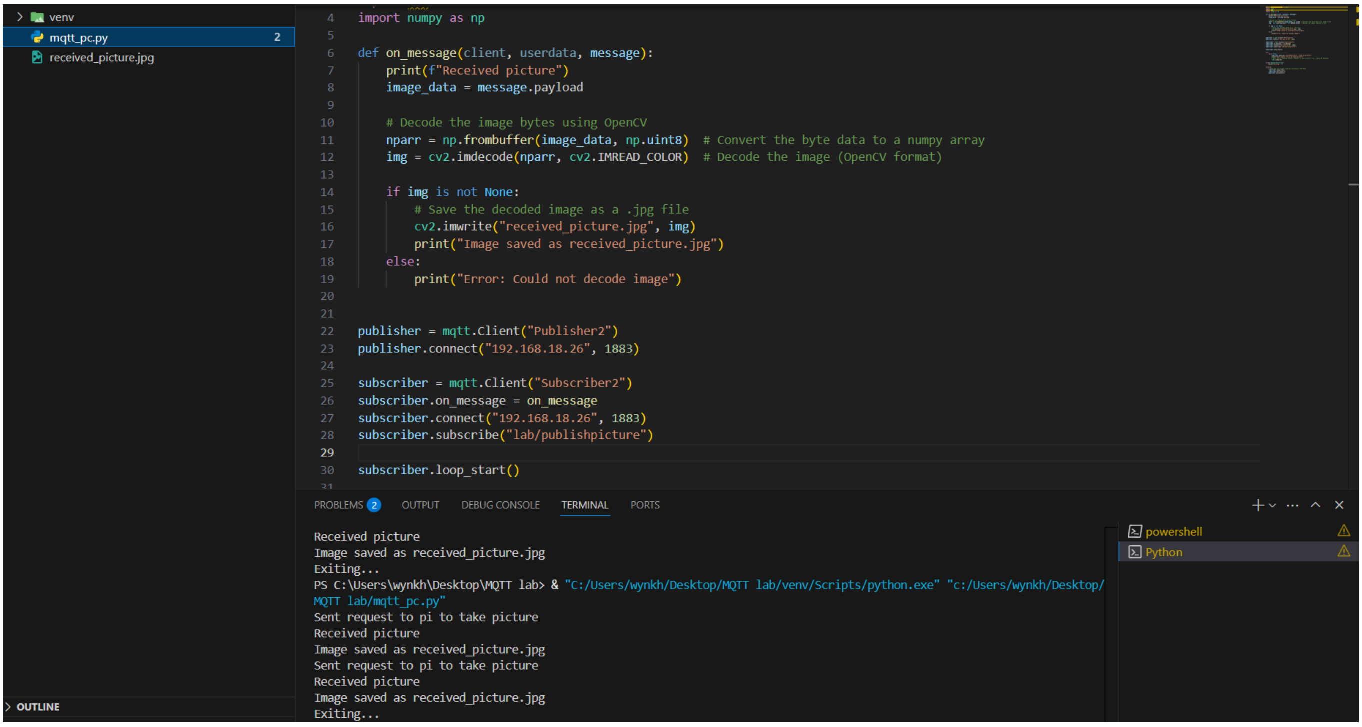
```
(mqtt) raspberry@raspberrypi:~ $ python3 mqtt_subscriber.py
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
Received message 'Hello, MQTT!' on topic 'test/topic'
```

```
Published: Hello, MQTT!
Message 2 published successfully!
Published: Hello, MQTT!
Message 3 published successfully!
Published: Hello, MQTT!
Message 4 published successfully!
Published: Hello, MQTT!
Message 5 published successfully!
Published: Hello, MQTT!
Message 6 published successfully!
Published: Hello, MQTT!
Message 7 published successfully!
Published: Hello, MQTT!
Message 8 published successfully!
Published: Hello, MQTT!
Message 9 published successfully!
Published: Hello, MQTT!
Message 10 published successfully!
Published: Hello, MQTT!
Message 11 published successfully!
Published: Hello, MQTT!
```

Fig. 13. Screenshot of the messages being published and received in each terminal

Lab05 (MQTT) Screenshots

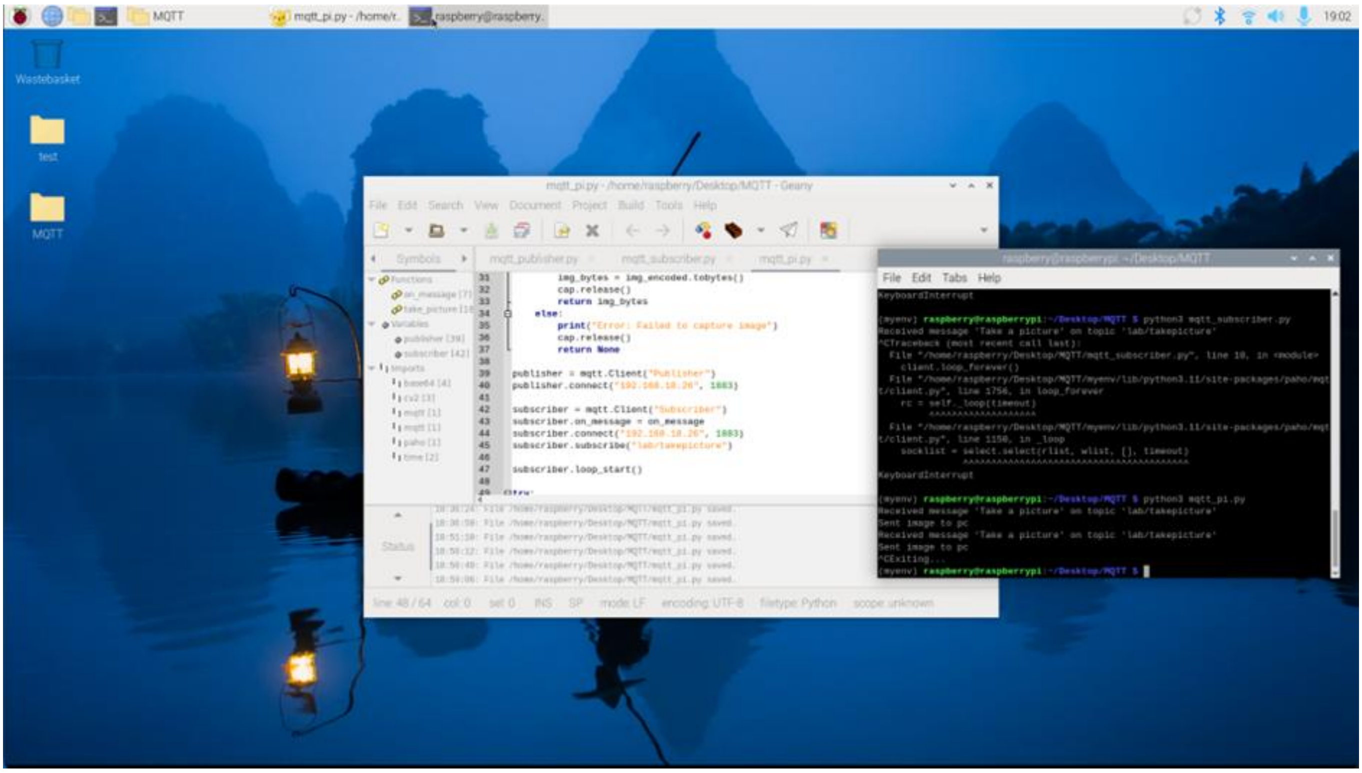
5. Lab Assignment (Did together with Wyvern Khiang 2200577)



```python
import numpy as np

def on_message(client, userdata, message):
    print(f"Received picture")
    image_data = message.payload

    # Decode the image bytes using OpenCV
    nparr = np.frombuffer(image_data, np.uint8)  # Convert the byte data to a numpy array
    img = cv2.imdecode(nparr, cv2.IMREAD_COLOR)  # Decode the image (OpenCV format)

    if img is not None:
        # Save the decoded image as a .jpg file
        cv2.imwrite("received_picture.jpg", img)
        print("Image saved as received_picture.jpg")
    else:
        print("Error: Could not decode image")


publisher = mqtt.Client("Publisher2")
publisher.connect("192.168.18.26", 1883)

subscriber = mqtt.Client("Subscriber2")
subscriber.on_message = on_message
subscriber.connect("192.168.18.26", 1883)
subscriber.subscribe("lab/publishpicture")

subscriber.loop_start()
```

```
Received picture
Image saved as received_picture.jpg
Exiting...
PS C:\Users\wynkh\Desktop\MQTT lab> & "C:/Users/wynkh/Desktop/MQTT lab/venv/Scripts/python.exe" "c:/Users/wynkh/Desktop/
MQTT lab/mqtt_pc.py"
Sent request to pi to take picture
Received picture
Image saved as received_picture.jpg
Sent request to pi to take picture
Received picture
Image saved as received_picture.jpg
Exiting...
```

Fig. 14. Screenshot of the code for MQTT



```python
            img_bytes = img_encoded.tobytes()
            cap.release()
            return img_bytes
        else:
            print("Error: Failed to capture image")
            cap.release()
            return None

publisher = mqtt.Client("Publisher")
publisher.connect("192.168.18.26", 1883)

subscriber = mqtt.Client("Subscriber")
subscriber.on_message = on_message
subscriber.connect("192.168.18.26", 1883)
subscriber.subscribe("lab/takepicture")

subscriber.loop_start()
```

```
18:36:24: File /home/raspberry/Desktop/MQTT/mqtt_pi.py saved.
18:36:56: File /home/raspberry/Desktop/MQTT/mqtt_pi.py saved.
18:51:58: File /home/raspberry/Desktop/MQTT/mqtt_pi.py saved.
18:58:12: File /home/raspberry/Desktop/MQTT/mqtt_pi.py saved.
18:58:40: File /home/raspberry/Desktop/MQTT/mqtt_pi.py saved.
18:59:06: File /home/raspberry/Desktop/MQTT/mqtt_pi.py saved.
```

```
KeyboardInterrupt

(myenv) raspberry@raspberrypi:~/Desktop/MQTT $ python3 mqtt_subscriber.py
Received message 'Take a picture' on topic 'lab/takepicture'
^CTraceback (most recent call last):
  File "/home/raspberry/Desktop/MQTT/mqtt_subscriber.py", line 10, in <module>
    client.loop_forever()
  File "/home/raspberry/Desktop/MQTT/myenv/lib/python3.11/site-packages/paho/mqt
t/client.py", line 1756, in loop_forever
    rc = self._loop(timeout)
         ^^^^^^^^^^^^^^^^^^^
  File "/home/raspberry/Desktop/MQTT/myenv/lib/python3.11/site-packages/paho/mqt
t/client.py", line 1150, in _loop
    socklist = select.select(rlist, wlist, [], timeout)
               ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
KeyboardInterrupt

(myenv) raspberry@raspberrypi:~/Desktop/MQTT $ python3 mqtt_pi.py
Received message 'Take a picture' on topic 'lab/takepicture'
Sent image to pc
Received message 'Take a picture' on topic 'lab/takepicture'
Sent image to pc
^CExiting...
(myenv) raspberry@raspberrypi:~/Desktop/MQTT $
```

Fig. 15. Screenshot of publisher taking webcam screenshot in the background and sending to the MQTT broker

Lab05 (MQTT) Screenshots

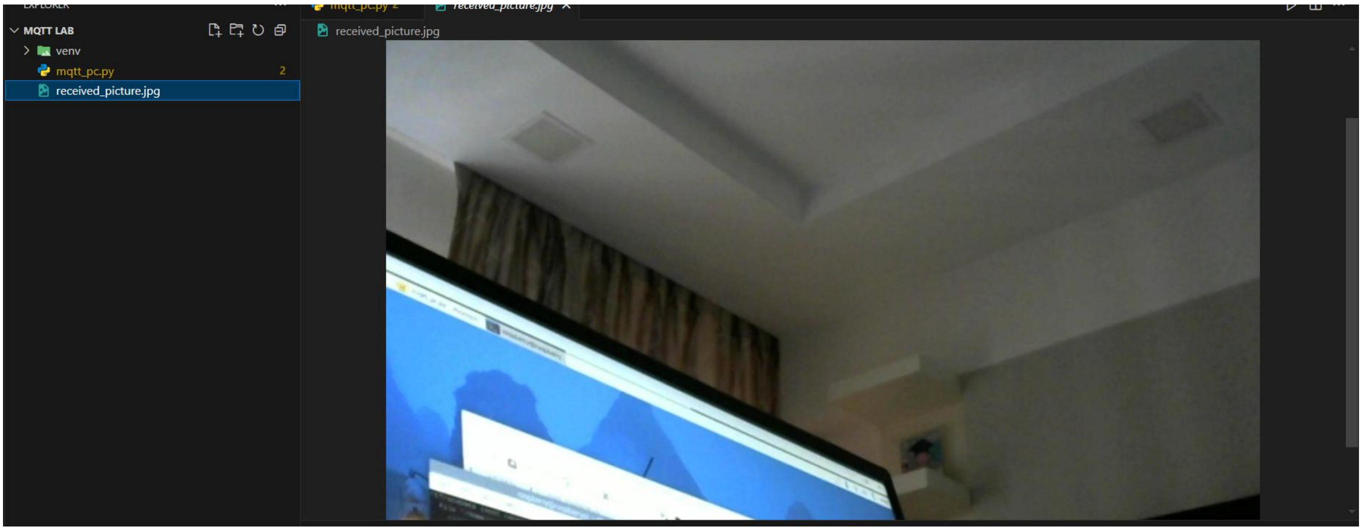


Fig. 15. Screenshot of image received by the subscriber from the MQTT broker